

ISSN 1816-0301 (Print)  
ISSN 2617-6963 (Online)

УДК 004.932  
<https://doi.org/10.37661/1816-0301-2020-17-2-25-35>

Поступила в редакцию 26.01.2020  
Received 26.01.2020

Принята к публикации 24.03.2020  
Accepted 24.03.2020

## Скелетизация изображений на основе комбинации одно- и двухподытерационных моделей

Ц. Ма, В. Ю. Цветков<sup>✉</sup>, В. К. Конопелько

Белорусский государственный университет  
информатики и радиоэлектроники, Минск, Беларусь  
<sup>✉</sup>E-mail: [vtsvet@bsuir.by](mailto:vtsvet@bsuir.by)

**Аннотация.** Рассматривается задача скелетизации бинарных изображений. Скелетизация дает возможность представить бинарное изображение в виде множества тонких линий, взаимное расположение, размеры и форма которых адекватно описывают размеры, форму и ориентацию в пространстве соответствующих областей изображения. Высокое качество скелетов обеспечивают итерационные параллельные алгоритмы. Они могут реализовываться с использованием одной или нескольких подытераций. На каждой из них происходит удаление избыточных элементов, окрестности которых удовлетворяют определенным условиям. Для многих одноподытерационных алгоритмов характерно нарушение связности и формирование избыточных фрагментов скелета. Наиболее качественные скелеты формирует известный одноподытерационный алгоритм ОПТА (One-Pass Thinning Algorithm), основанный на 18 бинарных масках, который, однако, чувствителен к контурному шуму и имеет высокую вычислительную сложность. Благодаря относительной простоте широко известность получил двухподытерационный алгоритм Zhang – Suen (ZS), основанный на шести логических условиях, но он размывает диагональные линии толщиной 2 пиксела и удаляет области размером 2×2 пиксела. Оба алгоритма не обеспечивают достижение минимальной толщины линий скелета (многие неузловые элементы имеют более двух соседей). Для построения предельно тонких связанных скелетов бинарных изображений с низкой вычислительной сложностью предлагаются математическая модель и алгоритм ОПСА (One-Pass Combination Algorithm) одноподытерационной скелетизации на основе комбинации и упрощения моделей одно- и двухподытерационной скелетизации. Данная модель и алгоритм позволяют повысить скорость скелетизации, восстановить исходное изображение по скелету, снизить избыточность связей элементов скелета.

**Ключевые слова:** скелетизация изображений, алгоритм ОПТА, алгоритм Zhang – Suen, одноподытерационная скелетизация, двухподытерационная скелетизация

**Для цитирования.** Ма, Ц. Скелетизация изображений на основе комбинации одно- и двухподытерационных моделей / Ц. Ма, В. Ю. Цветков, В. К. Конопелько // Информатика. – 2020. – Т. 17, № 2. – С. 25–35. <https://doi.org/10.37661/1816-0301-2020-17-2-25-35>

---

## Image skeletonization based on combination of one- and two-sub-iterations models

Jun Ma, Viktor Yu. Tsviatkou<sup>✉</sup>, Valery K. Kanapelka

Belarusian State University of Informatics and Radioelectronics, Minsk, Belarus  
<sup>✉</sup>E-mail: [vtsvet@bsuir.by](mailto:vtsvet@bsuir.by)

**Abstract.** This paper is focused on the field of the skeletonization of the binary image. Skeletonization makes it possible to represent a binary image in the form of many thin lines, the relative position, sizes and shape of which adequately describe the size, shape and orientation in space of the corresponding image areas. Skeletonization has many variety methods. Iterative parallel algorithms provide high quality skeletons. They can be implemented using one or more sub-iterations. In each iteration, redundant pixels, the neighborhoods of which meet certain conditions,

are removed layer by layer along the contour and finally they leave only the skeleton. Many one-sub-iterations algorithms are characterized by a breakdown in connectivity and the formation of excess skeleton fragments. The highest-quality skeletons are formed by the well-known single-iteration OPTA algorithm, which based on 18 binary masks, but it is sensitive to contour noise and has a high computational complexity. The Zhang and Suen two-iteration algorithm (ZS), which is based on 6 logical conditions, is widely used due to its relative simplicity. But it suffers from the problem of the blurs of the diagonal lines with a thickness of 2 pixels and the lost of the square which size is  $2 \times 2$  pixels. Besides, both algorithms mentioned above do not achieve the unit pixel thickness of the skeleton lines (many non-node pixels have more than two neighbors). Mathematical model and OPCA (One-Pass Combination Algorithm) algorithm which is based on a combination and simplification of single-iterative OPTA and two-iterative ZS are proposed for constructing extremely thin bound skeletons of binary images with low computational complexity. These model and algorithm also made it possible to accelerate the speed of skeletonization, to enhance recoverability of the original image on the skeleton and to reduce the redundancy of the bonds of the skeleton elements.

**Keywords:** image skeletonization, algorithm OPTA, algorithm Zhang – Suen, single-iteration skeletonization, double-iteration skeletonization

**For citation.** Ma J., Tsviatkou V. Yu., Kanapelka V. K. Image skeletonization based on combination of one- and two-sub-iterations models. *Informatics*, 2020, vol. 17, no. 2, pp. 25–35 (in Russian). <https://doi.org/10.37661/1816-0301-2020-17-2-25-35>

**Введение.** В задачах параметризации объектов изображений часто используется скелетизация (утонышение) – преобразование однородной области, соответствующей объекту, в множество тонких линий, взаимное расположение, размеры и форма которых передают информацию о размере, форме и ориентации в пространстве соответствующей области. Впервые задача утонышения областей бинарных изображений была рассмотрена в работе [1], где для распознавания образов предложен алгоритм на основе оконного усреднения и взвешивания значений пикселей. Затем в статье [2] был представлен алгоритм утонышения единичной области на основе постепенного вложения ее границ. Согласно предложенной в [3] классификации эти работы послужили началом развития двух основных групп алгоритмов скелетизации: неитерационных и итерационных. Неитерационные алгоритмы реализуют скелетизацию за один проход, что обеспечивает высокое быстродействие, но приводит к низкому качеству скелетов. Итерационные алгоритмы постепенно удаляют слои пикселей на границе области до получения тонких линий, что улучшает качество скелетов, но связано с высокой вычислительной сложностью. В свою очередь, итерационные алгоритмы разделяются на последовательные и параллельные. В последовательных алгоритмах результат скелетизации на каждой итерации зависит от порядка выборки пикселей, что приводит к неинвариантности скелета к повороту изображения. В параллельных алгоритмах порядок обработки пикселей на каждой итерации не влияет на результат, что обеспечивает стабильность скелета при повороте изображения и позволяет повысить скорость скелетизации за счет распараллеливания вычислений.

Параллельные алгоритмы разделяются по числу подытераций. В работе [4] предложен один из первых алгоритмов с одной подытерацией, использующий логические условия для определения удаляемых пикселей в окрестности  $3 \times 3$ ; линий скелета, возникающих из-за незначительных изломов границ областей, и избыточных пикселей на диагональных линиях. Алгоритмы с одной подытерацией на основе логических условий и окрестности  $3 \times 3$  пиксела предложены также в статьях [5, 6], где эти условия модифицируются для ускорения обработки. Известна также группа одноподытерационных алгоритмов [7–9] на основе масок различных размеров и ориентаций, берущих начало от работ [10, 11]. Алгоритмы данной группы за счет роста вычислительной сложности в некоторых случаях улучшают качество скелетов.

Проблемами многих одноподытерационных алгоритмов являются нарушение связности и формирование избыточных фрагментов скелета. Для их решения в некоторых алгоритмах каждая итерация разбивается на две и более подытерации. Благодаря своей относительной простоте согласно анализу [12] широкую известность получил двухподытерационный алгоритм ZS [13], основанный на шести логических условиях. Как установлено в работе [14], основными недостатками алгоритма ZS являются размытие диагональных линий толщиной 2 пиксела и удаление областей размером  $2 \times 2$  пиксела. Для устранения этих недостатков и раз-

вития алгоритма ZS предложены следующие его модификации: изменение порогового значения в первом логическом условии [14], использование горизонтальных и вертикальных прямоугольных окон [15, 16] и расширенных окон из 11 пикселей [17–19], расширение алгоритма на полутоновые изображения [20], расширение условий удаления пикселей на второй подытерации для устранения чрезмерной эрозии [21]. Повышение качества скелетов в данных модификациях достигается за счет роста вычислительной сложности. Многие двухподытерационные алгоритмы, как и одноподытерационные, ориентированы на скелетизацию символов сканированных документов.

Целью настоящей работы является формирование связанных скелетов минимальной толщины для объектов произвольной формы на бинарных изображениях с низкой вычислительной сложностью.

**Постановка задачи.** Для бинарного изображения  $I = \|i(y, x)\|_{(y=0, Y-1, x=0, X-1)}$  размером  $Y \times X$ , пиксели которого имеют значения 1 или 0 в зависимости от принадлежности площадному объекту или фону соответственно, алгоритмы скелетизации формируют матрицу  $S = \|s(y, x)\|_{(y=0, Y-1, x=0, X-1)}$  скелетизации, значения элементов которой 1 или 0 указывают на фрагменты скелета или фона соответственно.

В алгоритме ОРТА на каждой итерации окрестность единичного элемента  $s_{ОРТА}(y, x)$  матрицы скелетизации  $S_{ОРТА}$  проходит две проверки (перед первой итерацией значения пикселей бинарного изображения  $I$  переносятся в матрицу  $S_{ОРТА}$  скелетизации). Если окрестность элемента  $s_{ОРТА}(y, x)$  соответствует маске на рис. 1, а, то  $s_{ОРТА}(y, x) \leftarrow 0$  (шаг 1), а если маске на рис. 1, б, то  $s_{ОРТА}(y, x) \leftarrow 1$  (шаг 2). Затем полученная матрица  $S_{ОРТА}$  скелетизации проходит еще одну проверку. В результате этой проверки удаляются единичные элементы  $s_{ОРТА}(y, x)$ , окрестность которых совпадает с одной из масок на рис. 1, в (шаг 3). Обобщенная маска для алгоритма ОРТА показана на рис. 1, г, где  $p(1) = s_{ОРТА}(y, x)$ .

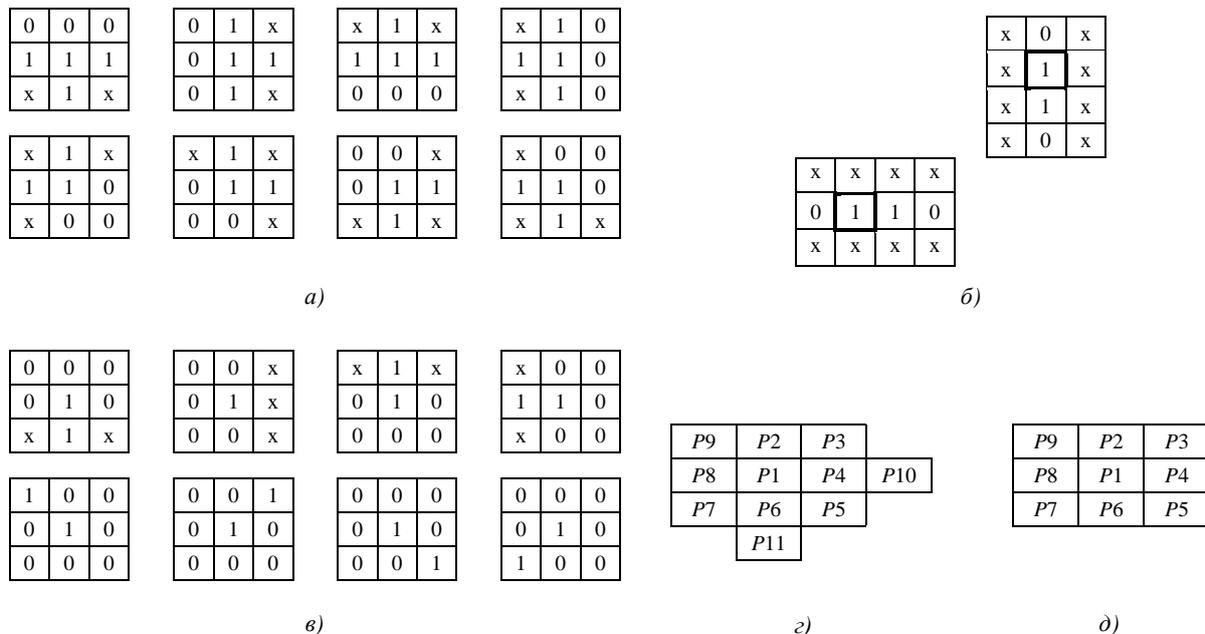


Рис. 1. Бинарные маски алгоритмов ОРТА и ZS: а) на шаге 1 ОРТА (x – безразличное состояние); б) на шаге 2 ОРТА; в) на шаге 3 ОРТА; г) общая по шагам 1–3 ОРТА; д) для алгоритма ZS

В алгоритме ZS каждая итерация состоит из двух подытераций и приводит к обнулению части единичных элементов  $s_{zs}(y, x)$  матрицы  $S_{zs}$  скелетизации (перед первой итерацией значения пикселей бинарного изображения  $I$  переносятся в матрицу  $S_{zs}$  скелетизации), смежные элементы в окрестности которых (рис. 1,  $\delta$ ,  $p(1) = s_{zs}(y, x)$ ) удовлетворяют определенным условиям.

На первой подытерации обнуляются элементы  $s_{zs}(y, x)$ , их смежные элементы удовлетворяют условиям

$$2 \leq \sum_{k=2}^9 p(k) \leq 6; \quad (1)$$

$$A(p(1)) = 1; \quad (2)$$

$$p(2) p(4) p(6) = 0; \quad (3)$$

$$p(4) p(6) p(8) = 0, \quad (4)$$

где  $A(p(1))$  – число комбинаций 01, встречающихся среди элементов  $p(2)–p(9)$ .

На второй подытерации обнуляются элементы  $s_{zs}(y, x)$ , их смежные элементы удовлетворяют условиям (1), (2) и условиям

$$p(2) p(4) p(8) = 0; \quad (5)$$

$$p(2) p(6) p(8) = 0. \quad (6)$$

Итерации продолжаются до тех пор, пока элементы  $s_{zs}(y, x)$  не обнулятся.

В качестве примера на рис. 2 показано бинарное изображение  $I$  размером  $15 \times 15$  пикселей, содержащее несколько объектов (рис. 2, *a*), и бинарные изображения матриц скелетизации  $S$ , соответствующие бинарному изображению  $I$  и сформированные с помощью одноподытерационного ОРТА [11] (рис. 2, *б*) и двухподытерационного ZS [13] (рис. 2, *в*) алгоритмов скелетизации.

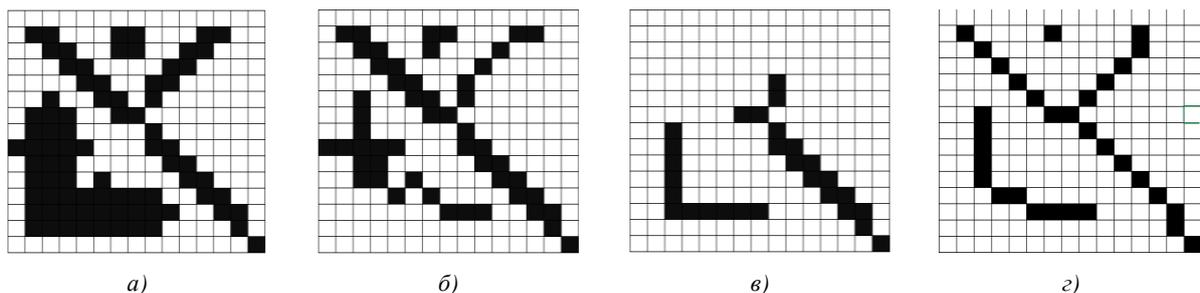


Рис. 2. Бинарное изображение  $I$  (*a*) и результаты его скелетизации, полученные с помощью алгоритмов ОРТА (*б*), ZS (*в*), OPCA (*г*)

На рис. 2 видны основные недостатки алгоритмов ОРТА и ZS. В обоих алгоритмах не достигается минимальная толщина линий скелета (многие узловые элементы имеют более двух соседей). Алгоритм ZS теряет некоторые диагональные линии и области  $2 \times 2$ . Скелеты, формируемые алгоритмом ОРТА, лишены этих недостатков, но подвержены контурному шуму (незначительные искривления контурной линии отражаются на форме скелета).

Имея превосходство в качестве скелетов, алгоритм ОРТА обладает большей вычислительной сложностью в сравнении с алгоритмом ZS: время скелетизации изображения на рис. 2, *a* алгоритмом ОРТА приблизительно в 1,9 раза больше, чем время скелетизации алгоритмом ZS. Это подтверждается теоретической оценкой сложности алгоритмов. Из рис. 1, *a–в*

следует, что в алгоритме ОРТА на обработку каждого единичного элемента  $s_{ОРТА}(y, x)$  матрицы  $S_{ОРТА}$  на каждой итерации приходится до 50 операций обработки смежных элементов (44 операции на шаге 1 и 6 операций на шаге 2) и еще до 56 операций на каждый единичный элемент  $s_{ОРТА}(y, x)$  необходимо для дополнительной проверки на шаге 3. Согласно выражениям (1)–(6) на каждой итерации элементы  $p(2)$ ,  $p(4)$ ,  $p(6)$ ,  $p(8)$  выбираются для обработки шесть раз, остальные элементы – три раза. Таким образом, на все единичные элементы  $s_{ZS}(y, x)$  матрицы  $S_{ZS}$  на каждой итерации приходится 36 операций обработки смежных элементов. Следовательно, вычислительная сложность алгоритма ОРТА приблизительно в 1,4 раза выше вычислительной сложности алгоритма ZS (при большом числе итераций вычислительной сложностью шага 3 алгоритма ОРТА можно пренебречь).

Качество скелетов, сформированных алгоритмом ОРТА, делает его эффективной базой для создания нового алгоритма скелетизации, скорость которого может быть увеличена за счет использования для удаления избыточных элементов скелета вместо бинарных масок некоторых логических условий из алгоритма ZS.

**Предлагаемые математическая модель и алгоритм скелетизации.** Для построения предельно тонких связанных скелетов (неузловые элементы скелета имеют не более двух соседей) бинарных изображений с низкой вычислительной сложностью предлагается математическая модель ОРСА одноподытерационной скелетизации на основе комбинации и упрощения моделей ОРТА и ZS. Согласно предлагаемой модели на каждой итерации единичный элемент  $s_{ОРСА}(y, x)$  матрицы  $S_{ОРСА}$  скелетизации обнуляется (перед первой итерацией значения пикселей бинарного изображения  $I$  переносятся в матрицу  $S_{ОРСА}$  скелетизации), если значения элементов выборки  $P = \|p(k)\|_{(k=2,11)}$ , формируемой, как показано на рис. 1,  $\varepsilon$  ( $p(1) = s_{ОРСА}(y, x)$ ), удовлетворяют условию (1) и условиям

$$\sum_{k=2}^9 |p(k) - p(\text{mod}_8(k-2+1)+2)| = 2; \quad (7)$$

$$\neg((p(4)=1) \wedge (p(8)=0) \wedge (p(10)=0) \vee (p(6)=1) \wedge (p(2)=0) \wedge (p(11)=0)). \quad (8)$$

После выполнения всех итераций, когда не удаляется ни один элемент скелета на основании условий (1), (7) и (8), полученная матрица  $S_{ОРСА}$  скелетизации проходит еще одну проверку, в результате которой удаляются единичные элементы  $s_{ОРСА}(y, x)$ , удовлетворяющие условию

$$(p(k)=0) \wedge (p(\text{mod}_8(k-2+3)+2)=1) \wedge (p(\text{mod}_8(k-2+5)+2)=1) \text{ при } k = \{3, 5, 7, 9\}. \quad (9)$$

Условие (7) эквивалентно условию (2) в модели ZS, условие (8) эквивалентно использованию масок  $3 \times 4$  и  $4 \times 3$  в модели ОРТА (см. рис. 1, б). Математическая модель ОРСА отличается от модели ОРТА исключением четырех масок  $3 \times 3$ , предназначенных для удаления избыточных пикселей на горизонтальных и вертикальных прямых линиях скелета (верхний ряд на рис. 1, а); использованием упрощенного условия (9) вместо четырех масок  $3 \times 3$  для удаления пикселей в точках изломов линий скелета (нижний ряд на рис. 1, а); исключением восьми масок  $3 \times 3$ , предназначенных для удаления избыточных концевых пикселей скелета (см. рис. 1, в), и от модели ZS – исключением всех условий удаления пикселей, кроме условий (1) и (2).

Из условий (1) и (7)–(9) следует, что на каждой итерации элементы  $p(2)$ ,  $p(4)$ ,  $p(6)$ ,  $p(8)$  выбираются для обработки два раза, элементы  $p(3)$ ,  $p(5)$ ,  $p(7)$ ,  $p(9)$ ,  $p(10)$ ,  $p(11)$  – один раз. Таким образом, на каждый единичный элемент  $s_{ОРСА}(y, x)$  матрицы  $S_{ОРСА}$  на каждой итерации приходится 14 операций обработки смежных элементов. Следовательно, вычислительная сложность алгоритма ОРСА ниже вычислительной сложности алгоритмов ОРТА и ZS

приблизительно в 3,6 и 2,6 раза соответственно (без учета вычислительной сложности дополнительной обработки матриц скелетизации в алгоритмах ОПТА и ОРСА, которой можно пренебречь при большом числе итераций).

Исходя из рассмотренной модели, в работе предлагается алгоритм ОРСА скелетизации на основе комбинации и упрощения алгоритмов ОПТА и ZS. Сущность алгоритма ОРСА состоит в формировании матрицы  $S_{\text{ОРСА}}$  скелетизации путем переноса в нее значений пикселей обрабатываемого бинарного изображения; итеративной обработке матрицы  $S_{\text{ОРСА}}$  скелетизации для удаления ее избыточных единичных элементов с использованием условий (1), (7), (8); прекращении итеративной обработки при отсутствии избыточных единичных элементов матрицы  $S_{\text{ОРСА}}$  скелетизации, для которых выполняются условия (1), (7), (8), и удалении избыточных единичных элементов матрицы  $S_{\text{ОРСА}}$  скелетизации с помощью условия (9).

Алгоритм ОРСА состоит из следующих шагов:

Вход: бинарное изображение  $I$ .

Инициализация матрицы  $S_{\text{ОРСА}}$  скелетизации:  $S_{\text{ОРСА}} \leftarrow I$ .

Цикл 1 (итеративная обработка матрицы  $S_{\text{ОРСА}}$ ).

Инициализация счетчика удаленных элементов:  $N_{DE} \leftarrow 0$ .

Цикл 2 по  $(y, x)$  (удаление избыточных элементов матрицы  $S_{\text{ОРСА}}$ ).

Если для  $s_{\text{ОРСА}}(y, x) = 1$  выполняются условия (1), (7), (8), то  $s_{\text{ОРСА}}(y, x) \leftarrow 0$ ;

$N_{DE} \leftarrow N_{DE} + 1$ .

Конец цикла 2.

Если  $N_{DE} = 0$ , то выход из цикла 1.

Конец цикла 1.

Цикл 3 по  $(y, x)$  (удаление избыточных элементов матрицы  $S_{\text{ОРСА}}$ ).

Если для  $s_{\text{ОРСА}}(y, x) = 1$  выполняется условие (9), то  $s_{\text{ОРСА}}(y, x) \leftarrow 0$ .

Конец цикла 3.

Выход: матрица  $S_{\text{ОРСА}}$  скелетизации.

Таким образом, в результате выполнения алгоритма ОРСА формируется матрица  $S_{\text{ОРСА}}$  скелетизации, единичные элементы которой указывают на линии скелетов, соответствующих бинарным объектам на исходном изображении  $I$ .

**Оценка эффективности алгоритма скелетизации.** Проведено сравнение предложенного алгоритма ОРСА с известными алгоритмами ОПТА [11] и ZS [13] (алгоритмы реализованы на языке программирования C++ и протестированы на компьютере с ОС Windows 8 64-бит, CPU i7 2,6 GHz, RAM 8 GB). При этом использованы предложенные в [22, 23] оценки: толщина скелета  $ST$  ( $ST \in [0,1]$ ; чем ближе к единице значение  $ST$ , тем скелет тоньше) [22]; скорость скелетизации  $TS$  (число удаляемых элементов в секунду; чем больше значение  $TS$ , тем скорость утоньшения выше) [23]; средняя ось  $MA$ , определяющая возможность восстановления исходного изображения по скелету (чем ближе к единице значение  $MA$ , тем восстанавливаемость выше [22]). Данные оценки определяются с помощью следующих выражений:

$$ST = 1 - \frac{\sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} \bigcup_{k=1}^4 s(y, x) Q(k)}{\sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} s(y, x)}, \quad (10)$$

где  $Q(k)$  – бинарная маска (рис. 3), позволяющая выявить избыточные элементы скелета;

$$TS = \frac{\sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} i(y, x) - \sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} s(y, x)}{ET}, \quad (11)$$

где  $ET$  – время скелетизации, с;

$$MA = \frac{\sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} i'(y, x)}{\sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} i(y, x)}, \quad (12)$$

где  $i'(y, x)$  – пиксели восстановленного (например, с помощью морфологического наращивания) по скелету  $S$  бинарного изображения  $I' = \left\| i'(y, x) \right\|_{(y=0, Y-1, x=0, X-1)}$ .

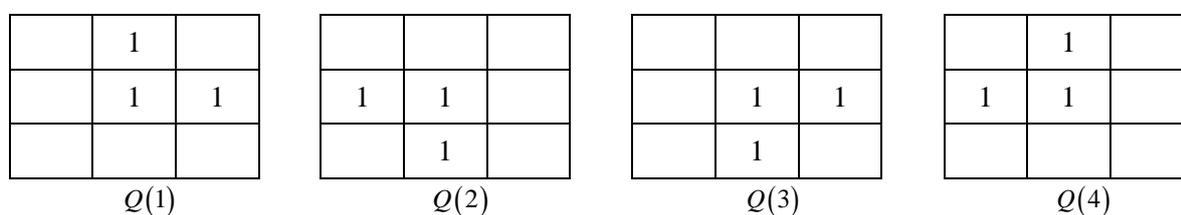


Рис. 3. Бинарные маски

Значения оценок  $ST$ ,  $TS$ ,  $MA$  для шести тестовых изображений  $I(1)$ – $I(6)$  (рис. 4) и алгоритмов ОРТА, ZS и ОРСА приведены в табл. 1. Результаты скелетизации  $S(1)$ – $S(6)$  тестовых изображений с помощью алгоритмов ОРТА, ZS и ОРСА показаны на рис. 5–7.

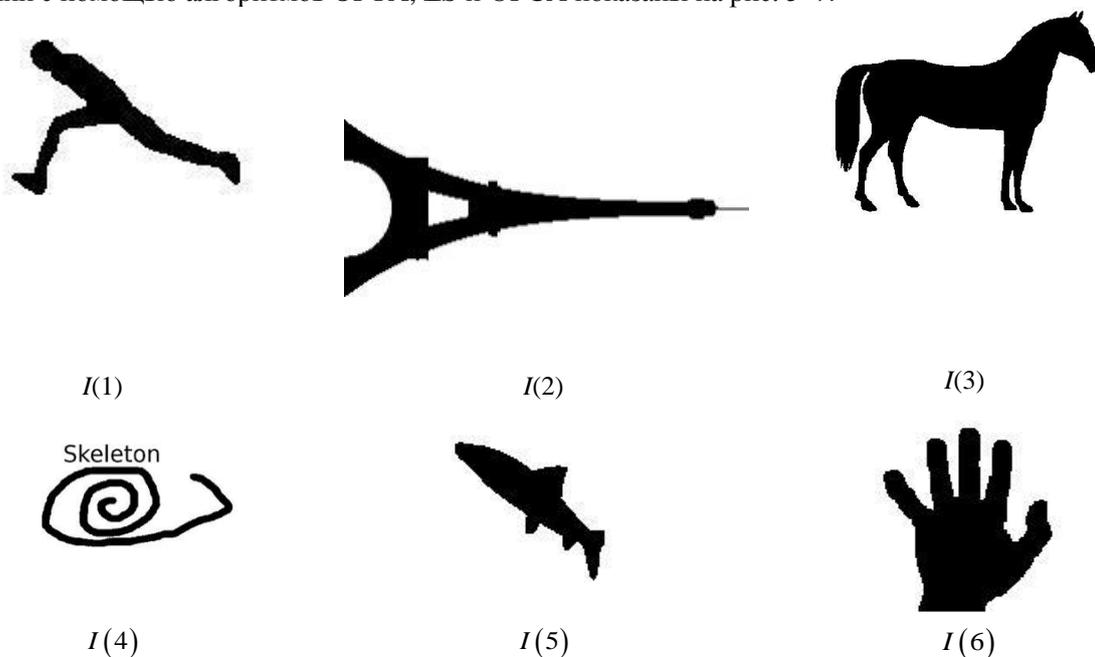


Рис. 4. Тестовые бинарные изображения

В табл. 1 приведены также значения избыточности связей между пикселями скелета  $SC$  ( $SC \geq 0$ ). Чем ближе к нулю значение  $SC$ , тем меньше избыточных связей и пикселей в скелете.

Неконцевые и узловые пиксели предельно тонкого скелета должны иметь связи только с двумя смежными пикселями. Поэтому для определения избыточности  $SC$  предлагается использовать выражение

$$SC = \frac{\sum_{y=1}^{Y-2} \sum_{x=1}^{X-2} \left( s(y, x) \left( \sum_{j=-1}^1 \sum_{i=-1}^1 s(y+j, x+i) - 1 \right) \right)}{\sum_{y=1}^{Y-2} \sum_{x=1}^{X-2} s(y, x)} - \left( 2 \sum_{y=1}^{Y-2} \sum_{x=1}^{X-2} s(y, x) - N_{EP} \right), \quad (13)$$

где  $N_{EP}$  – число концевых элементов скелета.

Таблица 1

Оценки результатов скелетизации тестовых изображений  $I(1)$ – $I(6)$ 

Изображение	Алгоритм	$ST$	$TS$	$MA$	$SC$
$I(1)$	OPTA	0,8631	645	0,8521	0,1237
	ZS	0,7833	854	0,9208	0,1849
	OPCA	0,9877	1318	0,9263	0,0183
$I(2)$	OPTA	0,8965	195	0,9115	0,0952
	ZS	0,8784	462	0,9602	0,1233
	OPCA	0,9993	575	0,9606	0,0032
$I(3)$	OPTA	0,9796	204	0,8941	0,0303
	ZS	0,8350	516	0,9502	0,1500
	OPCA	0,9991	705	0,9544	0,0058
$I(4)$	OPTA	0,9365	793	0,8986	0,0650
	ZS	0,8031	1352	0,9482	0,1700
	OPCA	0,9995	2549	0,9487	0,0028
$I(5)$	OPTA	0,7042	428	0,8540	0,2350
	ZS	0,6566	766	0,9279	0,2756
	OPCA	0,9931	1092	0,9436	0,0172
$I(6)$	OPTA	0,9618	351	0,7204	0,0419
	ZS	0,7275	633	0,9370	0,2220
	OPCA	0,9941	916	0,9493	0,0118

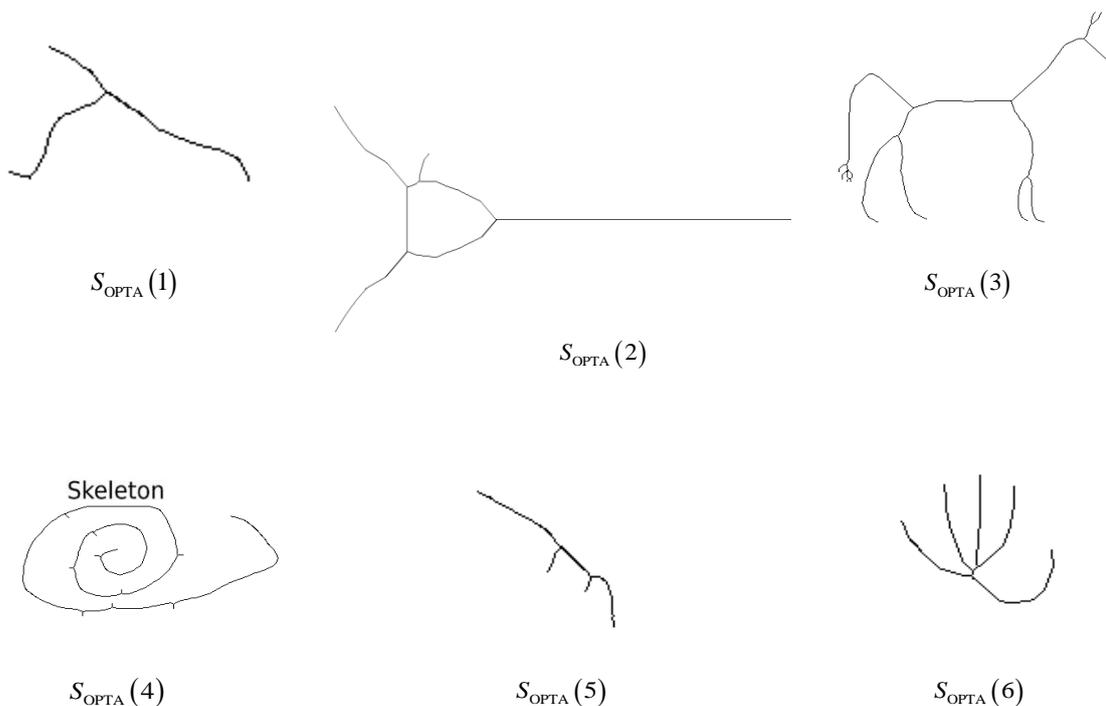


Рис. 5. Результаты скелетизации тестовых изображений с помощью алгоритма OPTA

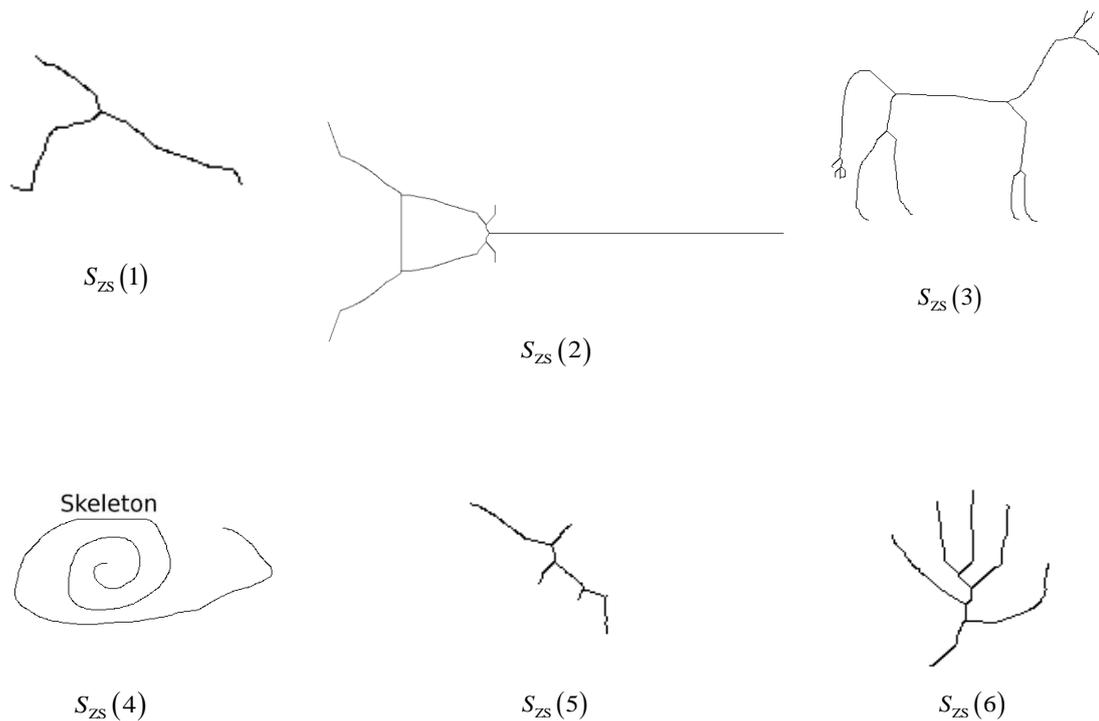


Рис. 6. Результаты скелетизации тестовых изображений с помощью алгоритма ZS

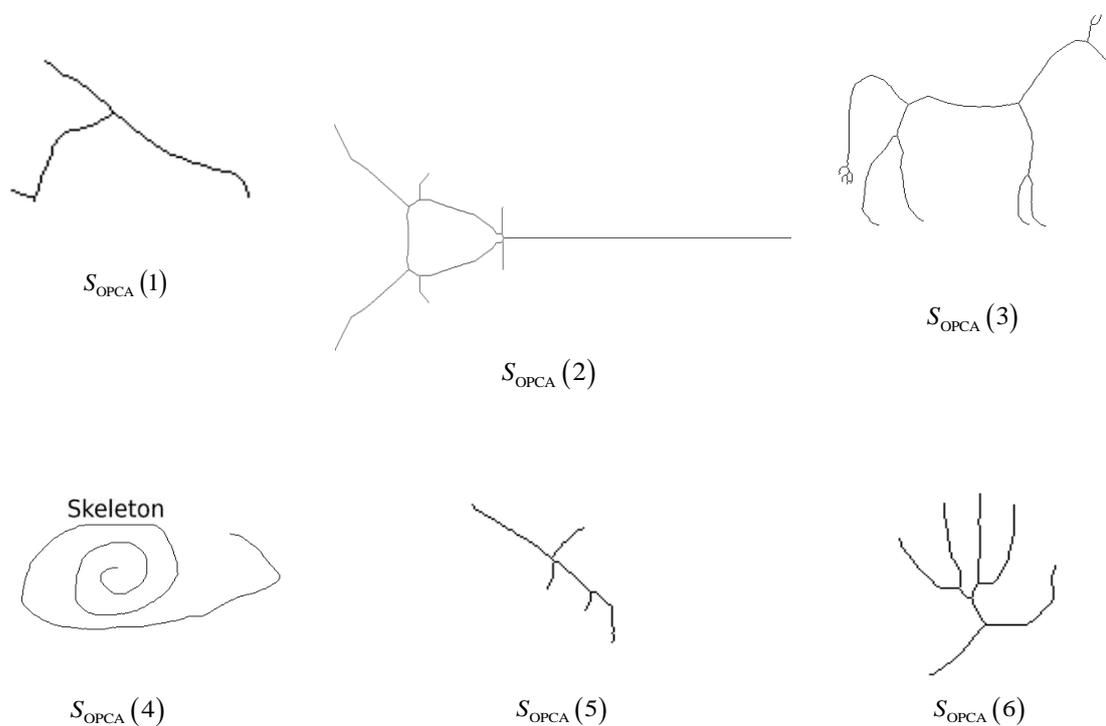


Рис. 7. Результаты скелетизации тестовых изображений с помощью алгоритма OPKA

Все оценки, приведенные в табл. 1, свидетельствуют о более высокой эффективности алгоритма OPKA в сравнении с алгоритмами OPTA и ZS.

В табл. 2 приведены значения оценок результатов скелетизации (10)–(13), усредненные по 100 бинарным изображениям, которые содержатся в работах [13, 14, 23] и тестовой базе Kimia's 99 [24].

Таблица 2

Средние значения оценок результатов скелетизации для 100 изображений, полученные с помощью алгоритмов OPTA, ZS и OPCA

Алгоритм	<i>ST</i>	<i>TS</i>	<i>MA</i>	<i>SC</i>
OPTA	0,9146	305	0,8588	0,0985
ZS	0,7799	640	0,9308	0,1876
OPCA	0,9956	860	0,9460	0,0097

Из табл. 2 следует, что алгоритм OPCA превосходит алгоритмы OPTA и ZS соответственно на 8,1 и 21,6 % по оценке *ST*, в 2,8 и 1,3 раза по оценке *TS*, на 8,7 и 1,5 % по оценке *MA*, в 10,2 и 19,3 раза по оценке *SC*.

Предложенная оценка *SC* избыточности связей между пикселями скелета, так же как и оценка *ST*, характеризует толщину линий скелета, но обеспечивает большую точность. Из табл. 1 следует, что для изображений *I*(3) и *I*(6) алгоритмы OPCA и OPTA практически не отличаются по оценке *ST* (отличия по этим изображениям составляют 2 и 3 % соответственно), однако существенно отличаются по оценке *SC* (в 5,2 и 3,6 раза соответственно).

**Заключение.** В работе предложены математическая модель и алгоритм OPCA одноподытерационной скелетизации на основе комбинации и упрощения моделей OPTA и ZS. Алгоритм OPCA отличается от OPTA исключением масок, предназначенных для удаления избыточных элементов на горизонтальных и вертикальных прямых линиях скелета; использованием упрощенного условия для удаления пикселей в точках изломов линий скелета; исключением масок, предназначенных для удаления избыточных концевых элементов скелета, а от ZS – исключением всех условий удаления пикселей, кроме двух основных. По сравнению с OPTA алгоритм OPCA обеспечивает уменьшение толщины скелета на 8,1 %, повышение скорости скелетизации в 2,8 раза, повышение восстанавливаемости исходного изображения по скелету на 8,7 %, уменьшение избыточности связей между пикселями скелета в 10,2 раза. По сравнению с ZS алгоритм OPCA обеспечивает уменьшение толщины скелета на 21,6 %, повышение скорости скелетизации в 1,3 раза, повышение восстанавливаемости исходного изображения по скелету на 1,5 %, уменьшение избыточности связей между пикселями скелета в 19,3 раза.

## References

1. Dinneen G. P. Programming pattern recognition. *Western Joint Computer Conference, New York, 1–3 March 1955*. New York, 1955, vol. 1, pp. 94–100.
2. Kirsch R. A., Cahn L., Ray C., Urban G. H. Experiments in processing pictorial information with a digital computer. *Eastern Computer Conference, Washington, 9–13 December 1957*. Washington, 1957, vol. 12, pp. 221–229.
3. Lam L., Lee S.-W., Suen C. Y. Thinning methodologies – a comprehensive survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1992, vol. 14, no. 9, pp. 869–885.
4. Holt C., Stewart A., Clint C., Perrott R. An improved parallel thinning algorithm. *Communication ACM*, 1987, vol. 30, pp. 156–160.
5. Manzanera A., Bernard T., Preteux F., Longuet B. Ultra-fast skeleton based on an isotropic fully parallel algorithm. *Discrete Geometry for Computer Imagery*, 1999, vol. 1568, pp. 313–324.
6. Bernard T. M., Manzanera A. Improved low complexity fully parallel thinning algorithm. *Proceedings 10th International Conference on Image Analysis and Processing, Venice, Italy, 27–29 September 1999*. Venice, 1999, pp. 215–220.
7. Chen C.-S., Tsai W.-H. A new fast one-pass thinning algorithm and its parallel hardware implementation. *Pattern Recognition Letters*, 1990, vol. 11, pp. 471–477.
8. Wu R.-Y., Tsai W.-H. A new one-pass parallel algorithm for binary images. *Pattern Recognition Letters*, 1992, vol. 13, pp. 715–723.

9. Deng W., Lyengar S. S., Brener N. E. A fast parallel thinning algorithm for the binary image skeletonization. *International Journal of High Performance Computing Applications*, 2000, vol. 14, pp. 65–81.
10. Stefanelli R., Rosenfeld A. Some parallel thinning algorithms for digital pictures. *Journal of the ACM*, 1971, vol. 18, pp. 255–264.
11. Chin R. T., Wan H. K., Stover D. L., Iverson R. D. A one-pass thinning algorithm and its parallel implementation. *Computer Vision, Graphics, and Image Processing*, 1987, vol. 40, pp. 30–40.
12. Harous, S., Elnagar A. Handwritten character-based parallel thinning algorithms: a comparative study. *University of Sharjah Journal of Pure & Applied Sciences*, 2009, vol. 6, no. 1, pp. 81–101.
13. Zhang T. Y., Suen C. Y. A fast parallel algorithm for thinning digital patterns. *Communications of the ACM*, 1984, vol. 27, no. 3, pp. 236–239.
14. Lu H. E., Wang P. S. P. A comment on "a fast parallel algorithm for thinning digital patterns". *Communications of the ACM*, 1986, vol. 29, no. 3, pp. 239–242.
15. Abdulla W. H., Saleh A. O. M., Morad A. H. A preprocessing algorithm for handwritten character recognition. *Pattern Recognition Letters*, 1988, vol. 7, no. 1, pp. 13–18.
16. Sossa J. H. An improved parallel algorithm for thinning digital patterns. *Pattern Recognition Letters*, 1989, vol. 10, pp. 77–80.
17. Guo Z., Hall R. W. Parallel thinning with two-subiteration algorithms. *Communications of the ACM*, 1989, vol. 32, no. 3, pp. 359–373.
18. Guo Z., Hall R. W. Fast fully parallel thinning algorithms. *CVGIP: Image Understanding*, 1992, vol. 55, no. 3, pp. 317–328.
19. Zhang Y. Y., Wang P. P. A parallel thinning algorithm with two-subiteration that generates one-pixel-wide skeletons. *International Conference on Pattern Recognition, Vienna, Austria, 25–29 August 1996*. Vienna, 1996, vol. 4, pp. 457–461.
20. Kundu M. K., Chaudhuri B. B., Majumder D. D. A parallel graytone thinning algorithm (PGTA). *Pattern Recognition Letters*, 1991, vol. 12, no. 8, pp. 491–494.
21. Kwon J.-S., Gi J.-W., Kang E.-K. An enhanced thinning algorithm using parallel processing. *Proceeding of the International Conference on Image Processing, Thessaloniki, Greece, 7–10 October 2001*. Thessaloniki, 2001, pp. 752–755.
22. Jang B. K., Chin R. T. One-pass parallel: analysis, properties, and quantitative evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1992, vol. 14, pp. 1129–1140.
23. Boudaoud L. B., Sider A., Tari A. A new thinning algorithm for binary images. *3rd International Conference on Control, Engineering & Information Technology, Tlemcen, Algeria, 25–27 May 2015*. Tlemcen, 2015, pp. 1–6.
24. Sebastian T. B., Klein P. N., Kimia B. B. Recognition of shapes by editing their shock graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004, vol. 26, no. 5, pp. 550–571.

### Информация об авторах

Ма Цзюнь, аспирант кафедры инфокоммуникационных технологий, Белорусский государственный университет информатики и радиоэлектроники, Минск, Беларусь.

E-mail: majun1313@hotmail.com

Цветков Виктор Юрьевич, доктор технических наук, доцент, заведующий кафедрой инфокоммуникационных технологий, Белорусский государственный университет информатики и радиоэлектроники, Минск, Беларусь.

E-mail: vtsvet@bsuir.by

Конопелько Валерий Константинович, доктор технических наук, профессор, профессор кафедры инфокоммуникационных технологий, Белорусский государственный университет информатики и радиоэлектроники, Минск, Беларусь.

E-mail: kafikt@bsuir.by

### Information about the authors

Jun Ma, Postgraduate Student of Department of Infocommunication Technologies, Belarusian State University of Informatics and Radioelectronics, Minsk, Belarus.

E-mail: majun1313@hotmail.com

Viktar Yu. Tsviatkou, Dr. Sci (Eng.), Associate Professor, Head of Department of Infocommunication Technologies, Belarusian State University of Informatics and Radioelectronics, Minsk, Belarus.

E-mail: vtsvet@bsuir.by

Valery K. Kanapelka, Dr. Sci (Eng.), Professor, Professor of Department of Infocommunication Technologies, Belarusian State University of Informatics and Radioelectronics, Minsk, Belarus.

E-mail: kafikt@bsuir.by