

МЕТОДЫ СУПЕРПИКСЕЛЬНОЙ СЕГМЕНТАЦИИ И ИХ ПРИМЕНЕНИЕ ДЛЯ АНАЛИЗА ИЗОБРАЖЕНИЯ С РАЗНОРОДНОЙ ТЕКСТУРОЙ

С. В. Саввин, А. А. Сирота

Воронежский государственный университет

Поступила в редакцию 27.09.2016 г.

Аннотация. Описаны методы и принципы обработки информации, лежащие в основе суперпиксельной сегментации изображений, а также преимущества суперпиксельной сегментации по сравнению с обычной. Перечислены основные показатели оценки качества алгоритмов суперпиксельной сегментации и проведено сравнение современных алгоритмов с использованием этих показателей. На примере алгоритма SLIC показана возможность модификации существующих алгоритмов для анализа изображений с разнородной структурой и приведено сравнение модифицированного алгоритма с классическим.

Ключевые слова: сегментация изображений, суперпиксели, алгоритмы NCut, SLIC, ERS.
Annotation. The advantages of the superpixel segmentation of images and its basic principles are described. The main quality metrics and the comparison of several state-of-the-art algorithms using these metrics are described. The possibility of modification of the existing algorithms is demonstrated by the example of images with heterogeneous texture and SLIC algorithm and the comparison of the modified and the classic SLIC algorithms is described.

Keywords: image segmentation, superpixels, NCut, SLIC, ERS.

ВВЕДЕНИЕ

Одной из наиболее часто решаемых задач при обработке изображений является сегментация – разбиение изображения на непересекающиеся области, покрывающие все изображение и однородные по некоторым признакам: цвет, яркость, текстура и т. д. Как известно, большинство стандартных алгоритмов обработки изображений используют их представление в виде регулярной пиксельной сетки, что обусловлено, в большей степени, исходным способом хранения изображений в цифровой форме. Однако это не всегда оптимально с многих точек зрения и, прежде всего, для организации последующей обработки. В этом плане относительно новым и активно развиваемым, прежде всего, за рубежом, является подход, основанный на так называемой суперпиксельной сегментации (СС). СС реализует разбиение изображения на множество мелких фрагментов (супер-

пикселей), представляющих из себя относительно однородные группы расположенных рядом пикселей [1, 2, 3, 4, 5, 6]. Каждый суперпиксель потенциально является атомарным регионом (фрагментом) изображения, т. е. все входящие в него пиксели рассматриваются при дальнейшей обработке как единое целое. При этом суперпиксели не обязательно должны иметь правильную форму (рис. 1) и, естественно, всегда имеется определенное число ошибок, допускаемых при стремлении разбить изображение на однородные фрагменты.

Как уже отмечалось, суперпиксельная карта изображения, полученная после проведения суперпиксельной сегментации, обладает рядом преимуществ по сравнению с обычной регулярной сеткой пикселей [1, 2, 3, 4, 5, 6]. Во-первых, это вычислительная эффективность: десятки и сотни тысяч пикселей заменяются всего лишь на десятки или сотни суперпикселей, которые выступают в алгоритмах как единое целое, что может значительно снизить время обработки в ал-



Рис. 1. Пример суперпиксельной сегментации изображения

горитмах распознавания. Во-вторых, при суперпиксельном представлении изображения можно говорить о взаимосвязях между отделёнными друг от друга пикселями, в то время как в пиксельном – только о связях между рядом лежащими пикселями. В-третьих, суперпиксели имеют большее смысловое значение. Каждый суперпиксель является согласованной единицей, так как принадлежащие ему пиксели имеют схожие цвета, яркость и прочие свойства. И, наконец, в отличие от окна прямоугольной формы, суперпиксель с большей вероятностью принадлежит только одному объекту изображения, не затрагивая остальные. Данные свойства суперпикселей обуславливают их использование при решении задач сегментации объектов как с известными, так и с неизвестными свойствами.

ИЗВЕСТНЫЕ МЕТОДЫ И АЛГОРИТМЫ СУПЕРПИКСЕЛЬНОЙ СЕГМЕНТАЦИИ

Основное требование к суперпиксельной сегментации состоит в следующем: пиксели внутри каждого суперпикселя должны быть максимально похожи, а пиксели, находящиеся в разных суперпикселях, должны в известной степени отличаться. Данная задача может решаться принципиально разными способами. В этом плане авторами рассмотрено множество известных алгоритмов СС.

Их, по нашему мнению, можно разделить на три большие группы [2, 3, 4, 5, 6]:

- Разделяющие (дивизимные);
- Объединяющие (агломеративные);
- Дискриминационные.

Разделяющие и объединяющие алгоритмы суперпиксельной сегментации представляют изображение в виде ненаправленного взвешенного графа $G = (V, E)$, в котором вершинами V служат пиксели, а грани E соединяют смежные пиксели. Весом грани $w(i, j)$ служит мера сходства вершин i и j , которые она соединяет.

В разделяющих алгоритмах всё изображение изначально считается одним большим суперпикселем, а затем осуществляется его рекурсивное разбиение на более мелкие компоненты. К разделяющим алгоритмам относится, например, первый из опубликованных алгоритмов суперпиксельной сегментации – алгоритм нормализованного разреза (Ncut) [3]. Его идея заключается в поиске на графе минимального нормализованного разреза между двумя половинами A и B графа $G = (V, E)$ (1). Под разрезом понимается суммарный вес всех граней, соединяющих две половины графа (2). Учёт размера кластера осуществляется через сумму весов всех его граней (3).

$$Ncut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(B, V)}{assoc(B, V)}, \quad (1)$$

$$cut(A, B) = \sum_{u \in A, v \in B} w(u, v), \quad (2)$$

$$assoc(A, V) = \sum_{u \in A, t \in V} w(u, t). \quad (3)$$

Объединяющие алгоритмы, напротив, принимают каждый пиксель изображения за отдельный суперпиксель, осуществляя в дальнейшем их жадное слияние. Примером объединяющего алгоритма является алгоритм ERS (Entropy Rate Superpixel) [4], в котором осуществляется максимизация целевой функции $F(A)$ (4).

$$F(A) = B(A) + \lambda \cdot B(A), \quad (4)$$

где $A \subset E$ – подмножество рёбер, образующих разбиение на суперпиксели.

Целевая функция состоит из двух основных компонентов, вес которых настраивается при помощи коэффициента λ :

- $H(A)$ – энтропия случайного пути на графе $G = (V, A)$, отвечающая за однородность пикселей внутри суперпикселя;

- $B(A)$ – функция балансировки, отвечающая за получение суперпикселей примерно одинакового размера.

В дискриминационных алгоритмах осуществляется предварительное разбиение пикселей на кластеры с последующим итеративным уточнением. Так, например, алгоритм SLIC (Simple Linear Iterative Clustering) [5] представляет из себя модифицированный алгоритм кластеризации k -средних, в котором осуществляется минимизация функции ошибки кластеризации. Основное отличие SLIC от классического алгоритма k -средних заключается в ограничении области поиска: пиксели для каждого сегмента ищутся не во всём изображении, а в небольшой области, пропорциональной среднему размеру сегмента. В качестве меры близости используется взвешенная сумма евклидовых расстояний по координатам и трём цветовым компонентам.

Таким образом, задача суперпиксельной сегментации сводится к задаче оптимизации значения той или иной целевой функции. При этом результат суперпиксельной сегментации может существенно отличаться в зависимости от используемого подхода. Выбор того или иного алгоритма зависит от вида решаемой задачи и, соответственно, выдвигаемых к нему требований.

ПОКАЗАТЕЛИ КАЧЕСТВА СУПЕРПИКСЕЛЬНОЙ СЕГМЕНТАЦИИ

Так как существует множество алгоритмов, отличающихся по получаемым результатам, крайне важной является задача их сравнения. Не все показатели качества обычной сегментации подходят для оценки суперпиксельной сегментации. Поэтому часть из них модифицируется с учетом специфики задачи, а, кроме того, добавляются принципиально новые показатели (например, для оценки компактности).

Можно выделить известные показатели для оценки качества суперпиксельной сегментации, которые группируются следующим образом:

- Показатели точности:
 - доля (процент) обнаруженных границ (относительное количество реальных границ объектов, находящихся рядом с границами суперпикселей (чем больше, тем лучше);
 - избыточность сегментации (ошибка, связанная с прохождением реальных границ объектов внутри суперпикселей (чем меньше, тем лучше);
 - максимально достижимая точность результирующей сегментации (точность обычной сегментации, выполненной на основе суперпиксельной, чем больше, тем лучше);
- Показатели компактности:
 - среднеквадратичное отклонение размеров суперпикселей (чем меньше, тем лучше);
 - изодиаметрический и изопериметрический коэффициенты (отношение площади суперпикселя к площади круга, имеющего такой же диаметр и периметр соответственно, чем больше, тем лучше);
- Показатель устойчивости к преобразованиям и шумам (F-мера, рассчитываемая для границ суперпикселей, выделенных на искажённом изображении, чем больше, тем устойчивее алгоритм).
- Показатель быстродействия – время работы алгоритма в стандартной реализации (чем меньше, тем лучше).

При проведении сравнительных исследований различных алгоритмов как правило строятся графики зависимостей данных показателей от числа суперпикселей. Однако при тестировании устойчивости к искажениям имеет смысл рассматривать зависимость от доли искажённых пикселей изображения.

СРАВНЕНИЕ ИЗВЕСТНЫХ АЛГОРИТМОВ

В ходе выполненного анализа для сравнения различных подходов к суперпиксельной сегментации были выбраны три современных и эффективных алгоритма из каждой группы (разделяющие, объединяющие, дискриминационные):

- Алгоритм Калинина [2];
- Алгоритм ERS;
- Алгоритм SLIC.

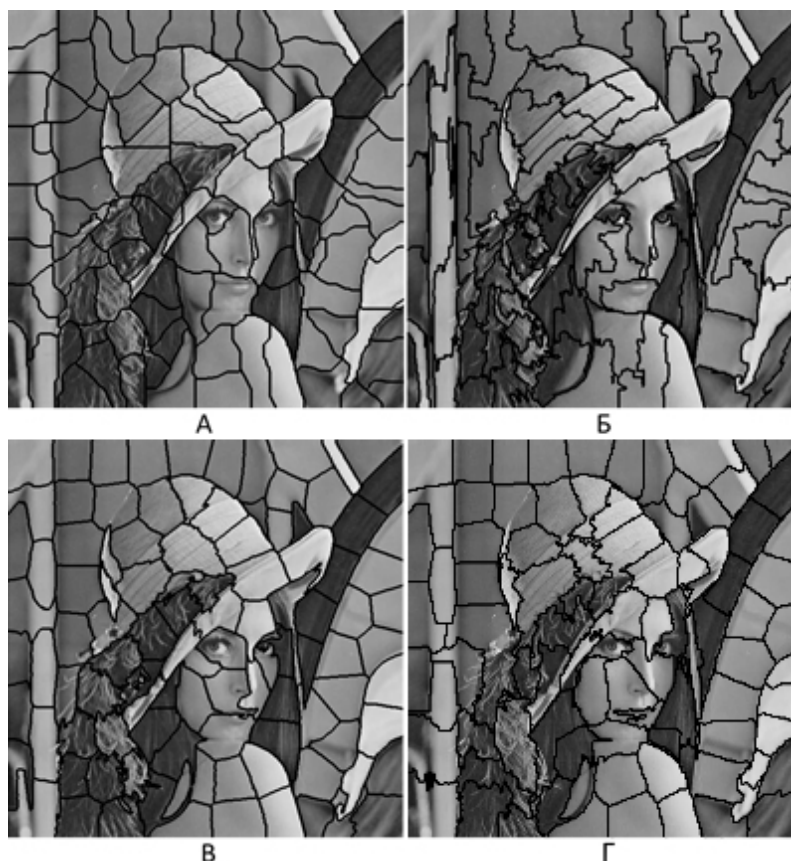


Рис. 2. Примеры работы алгоритмов (А – алгоритм Калинина, Б – алгоритм ERS, В – алгоритм SLIC, Г – модифицированный алгоритм SLIC)

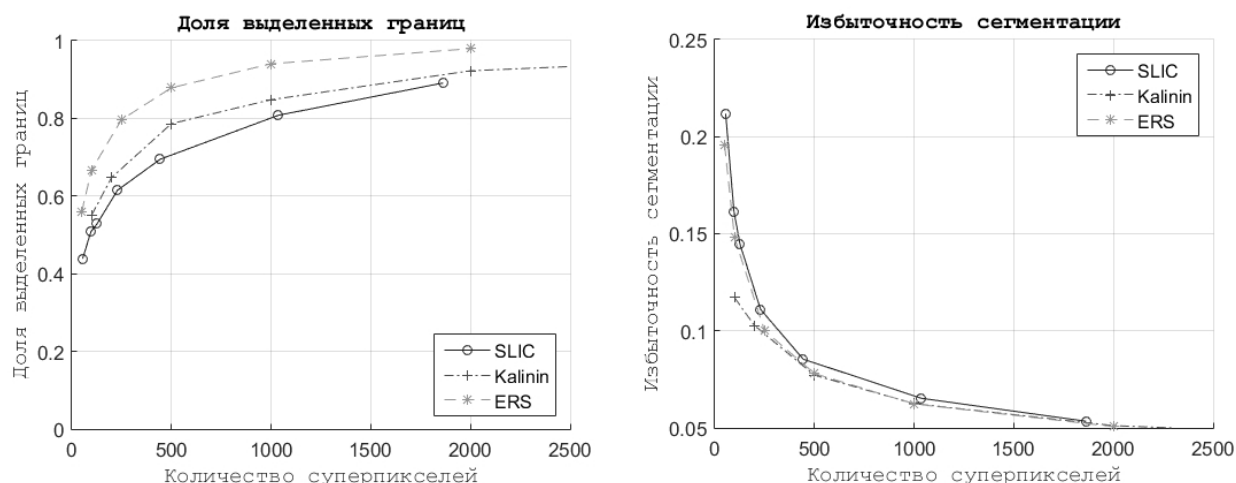


Рис. 3. Сравнение точности алгоритмов (слева – процент обнаруженных границ, справа – избыточность сегментации)

Иллюстративные примеры работы данных алгоритмов представлены на рис. 2.

Инструментарий для исследования и сравнения алгоритмов реализован на языке Matlab [6]. Анализ проводился на основе изображений из тестового набора Калифорнийского университета в Беркли [7]. Все алгоритмы реализованы на языке програм-

мирования С++ с соответствующим подключением к среде Matlab.

На рис. 3 приведено сравнение точности алгоритмов. Наибольший процент выделенных границ имеет алгоритм ERS, в то время как алгоритм Калинина имеет наименьшую избыточность сегментации.

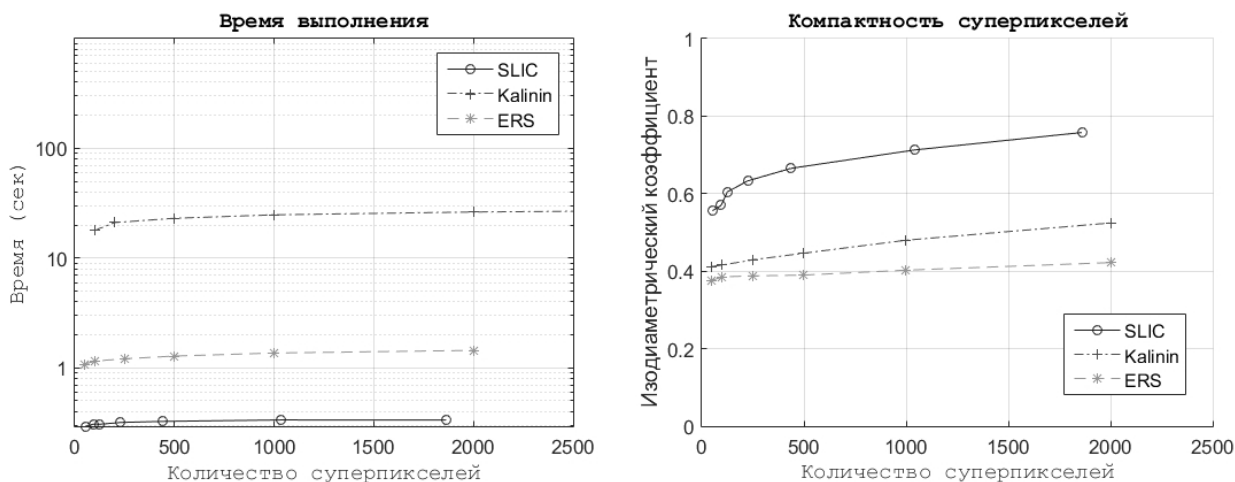


Рис. 4. Сравнение времени работы и компактности алгоритмов (слева – время работы, справа – изодиаметрический коэффициент)

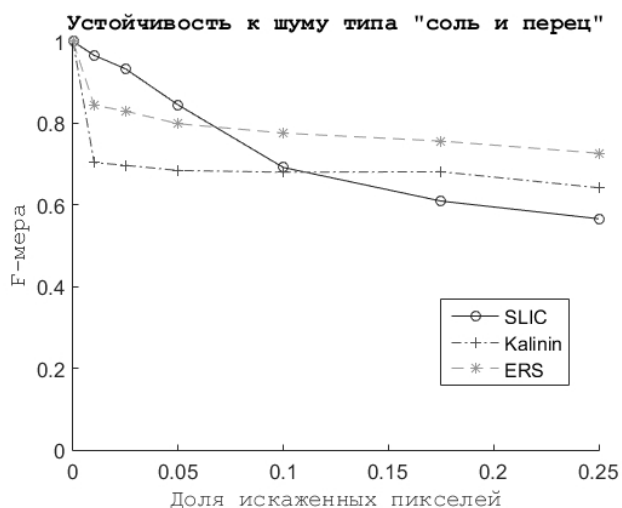


Рис. 5. Сравнение устойчивости алгоритмов к шуму типа «соль и перец»

Из рис. 4 видно, что SLIC превосходит другие алгоритмы по времени работы и компактности получаемых суперпикселей.

Увеличение числа изменённых пикселей при шумах типа «соль и перец» мало сказывается на результатах алгоритмов Калинина и ERS, но сильно ухудшает результаты алгоритма SLIC (рис. 5).

Таким образом, ни один из трёх рассмотренных алгоритмов не превосходит остальные по всем критериям, каждый из них имеет свои достоинства и недостатки. Поэтому далее для исследования возможностей развития методов суперпиксельной сегментации, направленного на обеспечение сегментации

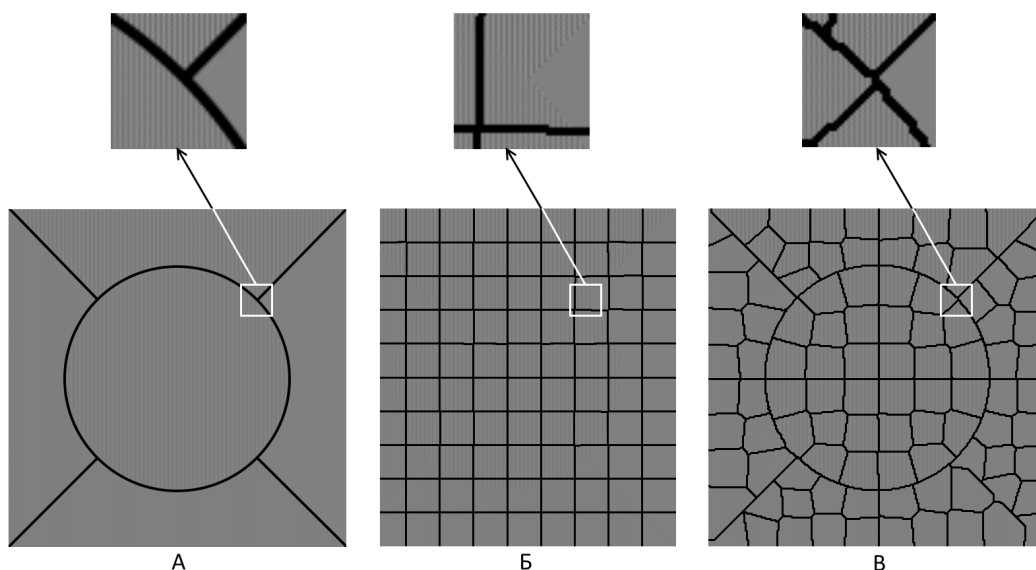


Рис. 6. Суперпиксельная сегментация изображения с областями примерно одинакового цвета, но с разной текстурой (А – реальные границы областей, Б – классический алгоритм SLIC, В – модифицированный алгоритм SLIC)

изображений, содержащих фрагменты с одинаковой яркостью и неоднородной текстурой был выбран алгоритм SLIC. Это связано с тем, что алгоритм SLIC достаточно прост в реализации и, в тоже время, эффективен, а его высокая скорость работы упрощает проведение экспериментов.

МОДИФИКАЦИЯ АЛГОРИТМА SLIC ДЛЯ АНАЛИЗА ИЗОБРАЖЕНИЙ С РАЗНОРОДНОЙ ТЕКСТУРОЙ

Алгоритмы суперпиксельной сегментации изображений обладают большими возможностями для развития и модификации. Одним из возможных подходов к их совершенствованию является привлечение дополнительных признаков к сегментации с целью улучшения общего качества СС. Рассмотрим классический алгоритм SLIC. Как уже отмечалось, алгоритм является пространственно ориентированной реализацией алгоритма кластеризации k -средних. Однако при его применении в стандартном варианте в процессе кластеризации отдельных пикселей не учитывается текстура отдельных областей изображения.

Данное обстоятельство хорошо проявляется на изображениях, у которых потенциальные суперпиксели имеют примерно одинаковую яркость и цветность, но отличаются по текстуре (рис. 6А). Стандартная реализация SLIC не способна распознать на таких изображениях границы между областями с разной текстурой и просто делит изображение на примерно одинаковые по размеру области (рис. 6Б).

Очевидное решение для данной проблемы – пытаться рассматривать сразу несколько соседних пикселей одновременно, например, блоки 2×2 , чтобы учитывать взаимосвязи и текстурные особенности для групп соседних пикселей. При этом, расстоянием между некоторым блоком и центром кластера будет взвешенная сумма евклидовых расстояний между средними значениями координат блока, средними значениями цветовых компонент блока и дисперсиями цветовых компонент блока.

Предлагаемая реализация алгоритма описывается следующими соотношениями:

1. Изображение представляется в цветовом пространстве LAB. Таким, образом каждый пиксель характеризуется набором из 5 признаков (3 цветовых компоненты и 2 координаты): $(l, a, b, x, y)^T$, а каждый блок 2×2 состоит из 4 соседних пикселей. N – общее количество пикселей в изображении.

2. Каждый блок 2×2 B_i , $i = \overline{1, N/4}$ характеризуется набором из 8 признаков: $B_i = (l_M^{(i)}, a_M^{(i)}, b_M^{(i)}, l_D^{(i)}, a_D^{(i)}, b_D^{(i)}, x_M^{(i)}, y_M^{(i)})^T$, где l_M , a_M и b_M – средние значения цветовых компонент, l_D , a_D и b_D – дисперсии цветовых компонент, а x_M и y_M – средние значения координат.

3. Изображение разбивается на сегменты размером $S \times S$ пикселей, где $S = \sqrt{N/K}$, K – заданное количество суперпикселей. Блок размером 2×2 , находящийся в центре каждого сегмента, становится центром кластера.

4. Для всех блоков B_i , $i = \overline{1, N/4}$ размером 2×2 вычисляется расстояние до центра кластера C_j , $j = \overline{1, K}$, которому блок первоначально принадлежит:

$$D_i^{(j)} = \sqrt{\left(\frac{d_M^{(i,j)}}{M_M}\right)^2 + \left(\frac{d_D^{(i,j)}}{M_D}\right)^2 + \left(\frac{d_S^{(i,j)}}{M_S}\right)^2}, \quad (5)$$

$$d_M^{(i,j)} = \sqrt{(l_M^{(i)} - l_M^{(j)})^2 + (a_M^{(i)} - a_M^{(j)})^2 + (b_M^{(i)} - b_M^{(j)})^2}, \quad (6)$$

$$d_D^{(i,j)} = \sqrt{(l_D^{(i)} - l_D^{(j)})^2 + (a_D^{(i)} - a_D^{(j)})^2 + (b_D^{(i)} - b_D^{(j)})^2}, \quad (7)$$

$$d_S^{(i,j)} = \sqrt{(x_M^{(i)} - x_M^{(j)})^2 + (y_M^{(i)} - y_M^{(j)})^2}, \quad (8)$$

где M_M , M_D , M_S – максимальные значения соответствующих расстояний в кластере. Начальный размер всех кластеров устанавливался равным:

$$N_j = \frac{S^2}{4}, \quad j = \overline{1, K}.$$

Область поиска для каждого кластера ограничена блоком размером $2S \times 2S$ пикселей относительно начального положения центра кластера.

Далее, в цикле, охватывающем следующие пункты, выполняется итеративное уточнение результатов сегментации.

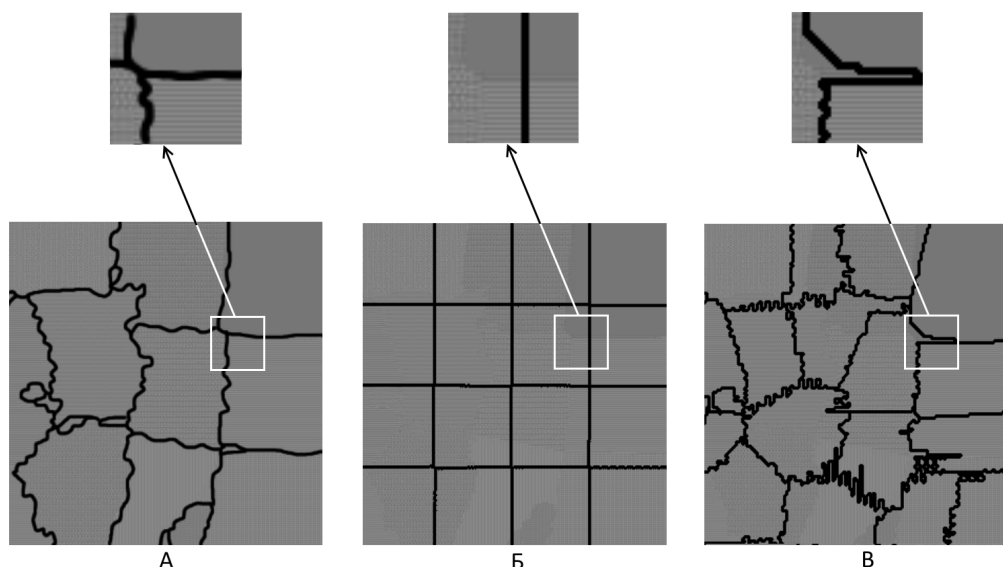


Рис. 7. Суперпиксельная сегментация фрагмента изображения с областями примерно одинакового цвета, но с разной текстурой, сгенерированного использованием тестового набора Беркли (А – реальные границы областей, Б – классический алгоритм SLIC, В – модифицированный алгоритм SLIC)

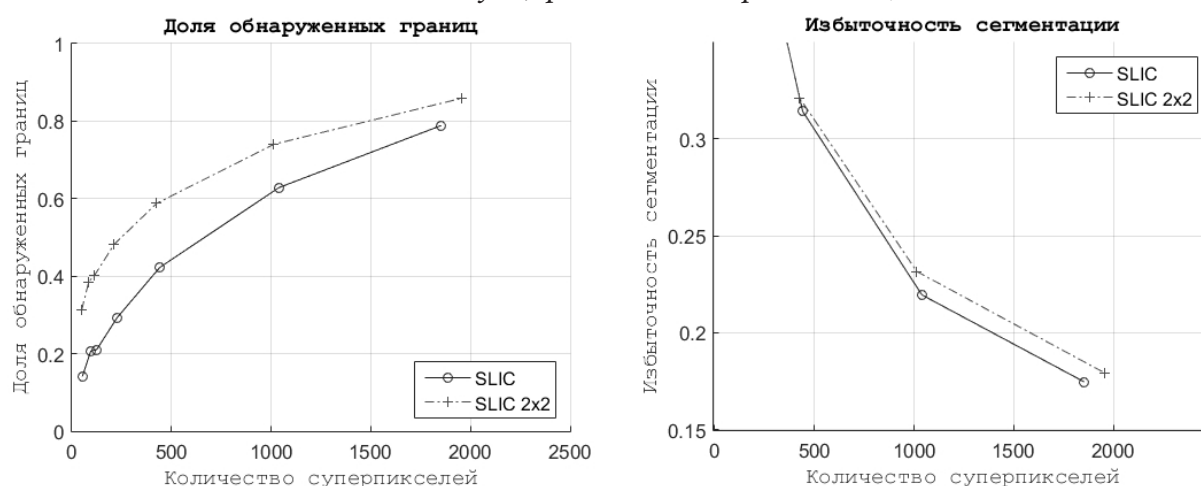


Рис. 8. Сравнение модифицированного и классического алгоритмов SLIC на изображениях с областями, но с разной текстурой (слева – процент обнаруженных границ, справа – избыточность сегментации)

5.1. Для центра каждого кластера вычисляется расстояние до всех блоков 2×2 в его области поиска и затем сравнивается с расстоянием до центра кластера, которому в данный момент принадлежит блок.

5.2. Если расстояние $D_i^{(j)}$ от некоторого блока B_i до центра j -го кластера C_j меньше, чем сохраненное расстояние $D_i^{(k)}$ до центра k -го кластера C_k , то пиксель переносится в j -ый кластер и центры кластеров пересчитываются (9, 10).

$$N'_j = N_j + 1, C'_j = \frac{C_j + B_i}{N_j + 1}, \quad (9)$$

$$N'_k = N_k - 1, C'_k = \frac{C_k - B_i}{N_k - 1}, \quad (10)$$

5.3. Если достигнута требуемая точность, например, разница центров кластеров на соседних шагах меньше заданного порогового значения или выполнено заданное число итераций осуществляется останов, в противном случае – переход на следующий шаг цикла.

На рис. 6В видно, что данный алгоритм уже способен обнаруживать границы на изображении, где стандартный алгоритм оказался бессильным.

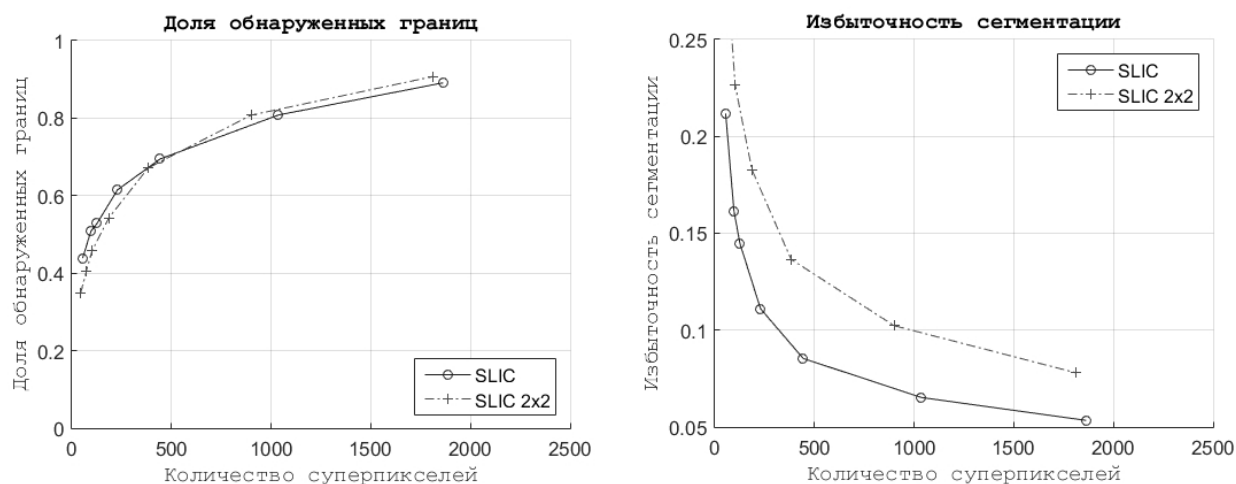


Рис. 9. Сравнение модифицированного и классического алгоритмов SLIC на изображениях из тестового набора Беркли (слева – процент обнаруженных границ, справа – избыточность сегментации)

Для проверки данной гипотезы необходимо было сгенерировать большое число подобных изображений. Это можно сделать на основе тестового набора Беркли, который включает в себя не только изображения, но и ручную размеченные границы объектов. При закрашивании каждого объекта случайно сгенерированной текстурой получаются изображения необходимые для проведения теста. Результат работы модифицированного алгоритма SLIC вновь оказывается лучше по сравнению с классическим алгоритмом (рис. 7).

Сравнение на большем количестве подобных изображений, подтверждает, что модифицированный алгоритм лучше выделяет границы объектов на таких изображениях (до 20 % больше), несколько уступая в избыточности сегментации, что объясняется более извилистыми границами суперпикселей. Соответствующие зависимости приведены на рис. 8.

При использовании модифицированного алгоритма на произвольных изображениях (рис. 2Г), он также показывает неплохие результаты. Так, при тестировании на «обычных» изображениях из тестового набора Беркли, он обнаруживает приблизительно такое же количество границ как классический алгоритм, несколько уступая в избыточности сегментации (рис. 9).

Таким образом, модифицированный алгоритм SLIC превосходит в точности ориги-

нальный алгоритм на тех изображениях, где оригинальный SLIC практически не способен выделять границы между суперпикселями, но в среднем имеет несколько худшие значения процента обнаруженных границ и ошибок сегментации.

ЗАКЛЮЧЕНИЕ

Данная статья посвящена алгоритмам суперпиксельной сегментации. Приведен анализ преимуществ суперпиксельного представления изображений по сравнению с традиционной матрицей пикселей, которые обуславливают использование суперпиксельной сегментации на предварительных этапах обработки изображений. Рассмотрены различные подходы к суперпиксельной сегментации изображений, которая является задачей оптимизации. В ходе сравнения современных алгоритмов суперпиксельной сегментации установлено, что ни один из них не обладает явным преимуществом над другими, имея как сильные, так и слабые стороны. Предложен модифицированный алгоритм суперпиксельной сегментации, обеспечивающий, в отличие известных, возможность выделения фрагментов с разнородной текстурой.

СПИСОК ЛИТЕРАТУРЫ

1. Superpixel: Empirical Studies and Applications / The University of Chicago. – Режим доступа: <http://ttic.uchicago.edu/~xren/research/superpixel/>
 2. Kalinin P. V. A graph based approach to hierarchical image over-segmentation / P. V. Kalinin, A. A. Sirota // Computer Vision and Image Understanding. – 2015. – Т.130. – С. 80–86.
 3. Malik J. Normalized Cuts and Image Segmentation / J. Shi, J. Malik // IEEE transactions on pattern analysis and machine intelligence. – 2000. – №8, Т. 22. – С. 888–905.
 4. Liu M. Entropy Rate Superpixel Segmentation / M. Liu [и др.] // сб. тр. участников IEEE Conference on Computer Vision and Pattern Recognition (Колорадо-Спрингс, 21-23 июня 2011 г.). – Колорадо-Спрингс, 2011. – С. 2097–2104.
 5. Achanta R. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods / R. Achanta [и др.] // IEEE transactions on pattern analysis and machine intelligence. – 2012. – №11, Т.34. – С. 2247–2281.
 6. Computer Vision | Superpixel / Technische Universität Chemnitz. – Режим доступа: <https://www.tu-chemnitz.de/etit/proaut/forschung/cv/segmentation.html.en>
 7. The Berkeley Segmentation Dataset and Benchmark / University of California, Berkeley. – Режим доступа: <https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/>
- Саввин С. В.** – магистр кафедры технологий обработки и защиты информации, факультет компьютерных наук, Воронежский государственный университет.
E-mail: savvin_s_v@sc.vsu.ru
- Сирота А. А.** – д-р техн. наук, профессор, заведующий кафедрой технологий обработки и защиты информации, факультет компьютерных наук, Воронежский государственный университет.
E-mail: sir@cs.vsu.ru
- Savvin S. V.** – Magister, Department of Processing Technology and Information Security, Computer Sciences Faculty, Voronezh State University.
E-mail: savvin_s_v@sc.vsu.ru
- Sirota A. A.** – Doctor of Technical Sciences, Professor, Department of Processing Technology and Information Security, Computer Sciences Faculty, Voronezh State University.
E-mail: sir@cs.vsu.ru