

## АЛГОРИТМ СЕГМЕНТАЦИИ ИЗОБРАЖЕНИЙ, ОСНОВАННЫЙ НА ПОИСКЕ СООБЩЕСТВ НА ГРАФАХ

С.В. Белим<sup>1</sup>, С.Б. Ларионов<sup>1</sup>

<sup>1</sup> Омский государственный университет им. Ф.М. Достоевского (ОмГУ), Омск, Россия

### Аннотация

В статье предложен алгоритм сегментации изображений, основанный на поиске сообществ на графах. Изображение представляется в виде неориентированного взвешенного графа, на котором осуществляется поиск сообществ. Каждому пикселю сопоставляется вершина графа. Ребрами соединяются только соседние пиксели. Вес ребер определяется разностью интенсивности трех цветовых составляющих пикселей. Для проверки качества разбиения графа на подграфы используется функция модульности Ньюмана. Предложен жадный алгоритм поиска сообществ на графе применительно к задаче сегментации изображения. Каждому сообществу соответствует сегмент на изображении. Проведен компьютерный эксперимент. Выявлено влияние параметра алгоритма на результаты сегментации. Показано, что предложенный алгоритм не чувствителен к импульсному случайному шуму.

**Ключевые слова:** выделение сообществ на графах, сегментация изображения.

**Цитирование:** Белим, С.В. Алгоритм сегментации изображений, основанный на поиске сообществ на графах / С.В. Белим, С.Б. Ларионов // Компьютерная оптика. – 2016. – Т. 40, № 6. – С. 904-910. – DOI: 10.18287/2412-6179-2016-40-6-904-910.

### Введение

Традиционные алгоритмы сегментации, основанные на представлении изображения в виде некоторой функции с последующим анализом либо её значений, либо значений её производной, имеют ряд ограничений при обработке фотографических изображений. Эти ограничения обычно преодолеваются предобработкой изображений, например, фильтрацией. Однако предобработка, как правило, приводит к размытию изображения, что отрицательно сказывается на результатах сегментации. В связи с этим в последнее время активно стал развиваться подход к сегментации, основанный на сопоставлении изображению некоторого графа с последующим разбиением на подграфы [1–5]. Однако при разрезании графа приходится сталкиваться с задачей выбора наилучшего разбиения. Для решения данной проблемы Ньюманом [6] была предложена функция модульности, численно характеризующая качество разделения графа. Активное использование данной функции связано с решением проблемы выделения сообществ на графах. Наилучшим считается разбиение вершин графа на сообщества, максимизирующее функцию модульности. В ряде работ были предложены алгоритмы сегментации, в той или иной степени использующие методы выделения сообществ на графах.

В статье [7] реализован подход к сегментации изображений на основе выделения сообществ на графах с помощью алгоритмов Fast Greedy [8] и Label Propagation [9]. Однако данные алгоритмы для реальных изображений требуют больших вычислительных ресурсов, в связи с чем авторами предварительно реализована процедура выделения суперпикселей. Под суперпикселем понимается сегмент изображения, включающий в себя некоторое количество реальных пикселей изображения [10]. При этом суперпиксели не пересекаются. Всё изображение разбивается на ограниченное количество суперпикселей. Представление

изображения в виде набора суперпикселей используется на этапе предобработки для снижения трудоемкости последующего процесса сегментации. Существует несколько подходов для формирования суперпикселей [10, 11, 12]. В работе [13] представлен эффективный метод выделения квазиуниверсальных суперпикселей с низкими требованиями к вычислительным ресурсам. Однако данный подход к сегментации имеет ограничение по точности, связанное с размерами суперпикселей. Несмотря на то, что границы суперпикселей могут иметь любую форму, их размер все равно сказывается на результатах сегментации.

В статье [14] предложен алгоритм сегментации с использованием трех алгоритмов выделения сообществ – Левена (Louvain), infomap и оптимизации устойчивости на основе алгоритма Левена, позволяющих получить наибольшее значение модульности. При построении графа, соответствующего изображению, в данной работе использован пороговый принцип. Ребро в графе между двумя вершинами, соответствующими пикселям, присутствует и имеет единственный вес, если разность интенсивностей пикселей превышает некоторое пороговое значение. Такой подход значительно упрощает исходное изображение и не чувствителен к небольшим изменениям цвета. Также данный метод дает хорошие результаты преимущественно на монохромных изображениях. Как показали компьютерные эксперименты, результаты сегментации чувствительны к выбранному пороговому значению и зависят от его удачного выбора для данного конкретного изображения.

В работе [15] алгоритм выявления сообществ используется на этапе постобработки сегментирования изображения. На начальном этапе авторы используют сглаживающие фильтры, на втором этапе проводится сегментация каким-либо алгоритмом, на третьем этапе производится проверка оптимальности сегментации с помощью вычисления модульности полученных сегментов. На основе результатов третьего этапа

происходит дополнительное объединение некоторых сегментов.

Аналогичный подход представлен в статье [16] для решения задачи выделения отдельных предметов на изображении. Сегментация выполняется одним из известных способов, после чего для построения графа меток, отделяющего предмет от фона, используется алгоритм выделения сообществ. В статье [17] предложен алгоритм постобработки сверхсегментированных изображений, то есть изображений, содержащих большое количество мелких деталей. На первом этапе производится сегментирование изображения каким-либо известным способом. Затем выполняется объединение сегментов, повышающее значение модульности. Аналогичный подход объединения мелких областей с помощью алгоритма выделения сообществ использован в работе [18] для выделения контуров на изображении. Контурные области на изображении определяются как границы сегментов. Более расширенный алгоритм объединения сегментов на основе методов выделения сообществ с построением иерархического дерева сегментов предложен в работе [19].

Таким образом, на сегодняшний день алгоритмы выявления сообществ в задачах сегментации изображений используются в основном как дополнение к другим методам либо требуют использования алгоритмов предварительной обработки. Целью данной статьи является построение алгоритма сегментации изображений на основе метода выявления сообществ без предварительной обработки и использования других алгоритмов сегментации.

### 1. Постановка задачи

Пусть задано изображение, размеры которого составляют  $N$  пикселей по горизонтали и  $M$  пикселей по вертикали. Положение каждого пикселя определяются двумя координатами  $(x, y)$ , где  $x$  – целое число из отрезка  $[0, N-1]$ ,  $y$  – целое число из отрезка  $[0, M-1]$ . Если цвет задан в градациях серого, то будем считать, что палитра содержит  $m$  цветов, то есть каждый пиксель характеризуется целым числом из отрезка  $[0, m-1]$ . Если изображение цветное, то будем использовать модель RGB. В этом случае каждому пикселю сопоставляется три целых числа из отрезка  $[0, m-1]$ . Введем обозначения для цветовых составляющих пикселя с координатами  $(x, y)$ :  $r(x, y)$  – интенсивность красной составляющей,  $g(x, y)$  – интенсивность зеленой составляющей,  $b(x, y)$  – интенсивность синей составляющей. В дальнейшем будем рассматривать общий случай цветного изображения. Более простой случай монохромного изображения может быть получен из него простым обнулением двух из трех цветовых составляющих каждого пикселя.

Сопоставим изображению неориентированный взвешенный граф  $G$ . В качестве множества вершин графа будем использовать множество пикселей изображения. Ребрами соединим только ближайших соседей. Под ближайшими соседями понимаются два пикселя  $v_i$  и  $v_j$ , каждая из координат которых отличается

не больше чем на единицу  $(|x_i - x_j| \leq 1) \vee (|y_i - y_j| \leq 1)$ . Рассмотрение ребер только между ближайшими соседями оправдано тем, что при сегментации ставится целью выделение односвязных областей изображения. Под односвязной областью будем понимать фрагмент изображения, между любыми двумя пикселями которого существует путь, составленный из отрезков, соединяющих ближайших соседей, и не пересекающий границы сегмента. Таким образом, каждый пиксель, расположенный не на границе изображения, имеет восемь ближайших соседей, с которыми он соединен ребрами. Четыре точки по углам изображения имеют всего по три ближайших соседа. Точки, расположенные на границе изображения, имеют по пять ближайших соседей.

Значительное влияние на результаты сегментации оказывает выбор функции, определяющей вес ребра, характеризующий меру сходства цветов двух ближайших соседей. Данная функция должна достаточно быстро убывать при переходе между пикселями, цвета которых значительно различаются. Как показано в работе [19], необходимыми свойствами обладает экспоненциальная функция. Будем вычислять вес ребра между вершинами  $v_i = (x_i, y_i)$  и  $v_j = (x_j, y_j)$  с помощью формулы:

$$d(v_i, v_j) = \exp\left(-\frac{1}{h} \sqrt{(r_i - r_j)^2 + (g_i - g_j)^2 + (b_i - b_j)^2}\right).$$

Здесь  $r_i = r(x_i, y_i)$ ,  $g_i = g(x_i, y_i)$ ,  $b_i = b(x_i, y_i)$ . Параметр  $h$  определяется пользователем и является общим для всего изображения. Данный параметр влияет на чувствительность алгоритма сегментации к градиенту изменения интенсивности цвета на изображении. Чем выше значение  $h$ , тем менее чувствителен вес ребра к перепадам цвета и тем больше размеры сегментов. Параметр  $h$  определяет величину перепада цвета, которая воспринимается алгоритмом как отклонение от равномерной заливки.

Задача сегментации изображения сводится к разбиению построенного графа  $G$  на подграфы. Вершины внутри подграфа должны быть связаны друг с другом сильнее, чем с вершинами из другого подграфа. Данная задача сводится к поиску сообществ на графах [6]. Каждое сообщество будет определять сегмент на изображении.

### 2. Алгоритм сегментации

При выделении сообществ одной из основных проблем является численная оценка качества разбиения графа. Наибольшее распространение получила функция модульности Ньюмана [6, 8, 20], на основе которой построено большое количество алгоритмов выделения сообществ на графах [21].

Будем задавать построенный граф  $G$  с помощью матрицы весов  $E$ . Элементы матрицы весов задаются следующим образом. Диагональный элемент  $E_{ii}$  показывает вес вершины  $v_i$ . На входе алгоритма веса всех вершин имеют нулевое значение. Элемент  $E_{ij}$  ( $i \neq j$ ) показывает вес ребра между вершинами  $v_i$  и  $v_j$ . По по-

строению графа из каждой вершины выходит не более восьми ребер, поэтому в каждой строке будет не более восьми ненулевых значений. Учитывая, что граф неориентированный, матрица  $E$  будет симметрична относительно главной диагонали. Следовательно, в памяти необходимо хранить  $4MN$  значений весов ребер. Для вычисления функции модульности необходимо перейти к приведенному виду матрицы весов  $e = E/m$ , где  $m = \sum_{i,j=1}^{MN} E_{ij}$ . В приведенной матрице весов элемент  $e_{ij}$  показывает долю веса заданного ребра в общем весе графа. В дальнейшем под матрицей весов будет пониматься именно приведенный вид. Легко увидеть, что  $\sum_{i,j=1}^{MN} e_{ij} = 1$ .

Величина модульности, численно определяющая качество разбиения графа, в общем виде [6, 8] определяется выражением:

$$Q(G) = \sum_{i=1}^K e_{ii} - \sum_{i=1}^K a_i b_i,$$

где  $K$  – количество вершин графа,  $a_i$  – приведенная исходящая степень вершины  $v_i$  ( $a_i = \sum_{j=1, j \neq i}^K e_{ij}$ ),  $b_i$  – приведенная входящая степень вершины  $v_i$  ( $b_i = \sum_{j=1, j \neq i}^K e_{ji}$ ). В

связи с тем, что в рассматриваемом случае граф является неориентированным, то входящая и исходящая степень для всех вершин совпадают ( $a_i = b_i$ ;  $i = 1, \dots, K$ ). Функция модульности принимает более простой вид:

$$Q(G) = \sum_{i=1}^K e_{ii} - \sum_{i=1}^K a_i^2.$$

Величина модульности показывает разность между долей всех ребер, лежащих внутри сообществ, для данного графа и ожидаемым значением, вычисленным для графа, в котором все вершины имеют одинаковую степень, а ребра размещены случайно. Нулевое значение функция модульности будет иметь для графа с равномерным случайным распределением ребер. Отклонение от случайности приводит к положительному значению величины модульности. Для случая графа, соответствующего изображению, отклонение от случайности в расположении ребер свидетельствует о наличии закономерностей в цветах пикселей, то есть об их принадлежности к одному сегменту.

Для поиска сообществ на графе будем использовать процедуру образования стяжек. Под стяжкой понимается преобразование, при котором некоторый подграф  $H$  графа  $G$  заменяется одной вершиной  $v_H$ . Если какая-то из вершин подграфа  $H$  была связана ребром с вершиной  $v$  из подграфа  $G/H$ , то вершина  $v_H$  также будет связана ребром того же веса с вершиной  $v$ . Вес новой вершины будет равен сумме весов ребер и вершин, входящих в подграф  $H$ . Новый граф обозначим  $G_H$ . Подграф  $H$  будем считать сообществом, если  $Q(G_H) > Q(G)$ . Следует отметить, что при

образовании стяжки уменьшается число вершин графа  $K$ .

Поиск подграфов  $H$ , увеличивающих функцию модульности, будем осуществлять с помощью жадного алгоритма.

Шаг 1. Выбираем произвольную вершину, имеющую нулевой вес и не рассматривавшуюся ранее. Обозначим её  $v_0$ . Начальный подграф, включающий только одну вершину  $v_0$ , обозначим  $H^{(0)}$ . Если таких вершин нет, происходит выход из алгоритма.

Пусть на некотором этапе получен подграф  $H^{(k)}$  ( $k = 0, 1, \dots$ ). образуем из этого графа стяжку. Вершину, которая заменяет граф  $H^{(k)}$ , обозначим  $v_0^{(k)}$ .

Шаг 2. Рассматриваем ближайших соседей вершины  $v_0^{(k)}$ . Обозначим соседние вершины  $v_{0i}^{(k)}$  ( $i = 1, \dots, 8$ ). Последовательно рассматриваем подграфы, состоящие из двух вершин  $v_0^{(k)}$  и  $v_{0i}^{(k)}$  ( $i = 1, \dots, 8$ ), и образуем из них стяжки. Вычисляем изменение функции модульности  $\Delta Q$  и переходим к шагу 3.

Шаг 3. Если одна или несколько стяжек обеспечивают положительное изменение функции модульности  $\Delta Q$ , то выбираем из них ту, которая обеспечивает максимальное значение  $\Delta Q$ , и добавляем её к подграфу  $H^{(k)}$ . Новый подграф обозначим  $H^{(k+1)}$  и переходим к шагу 2. В противном случае переходим к шагу 4.

Шаг 4. Если объединение вершины  $v_0^{(k)}$  со всеми ближайшими соседями в стяжку не дает положительных значений  $\Delta Q$ , то формирование сообщества завершено, переходим к шагу 1.

На выходе алгоритма получим граф, состоящий из стяжек. При этом образование новых стяжек понижает значение функции модульности. Каждой стяжке соответствует подграф в исходном графе. Пиксели, соответствующие вершинам одного подграфа, относятся к одному сегменту.

Каждый раз на втором шаге алгоритма происходит перебор не более чем  $NM$  вершин. Для каждой вершины вычисляется величина изменения функции модульности  $\Delta Q$ . Второй шаг алгоритма может быть выполнен не более чем  $NM$  раз. Отсюда можно сделать вывод, что асимптотическая трудоемкость алгоритма  $O(N^2M^2)$ . Для оценки величины константы перед старшим членом оценки трудоемкости нужно определить количество операций, необходимых для вычисления  $\Delta Q$ . Пусть в результате образования стяжки объединены вершины  $v_k$  и  $v_l$ . В результате этого граф  $G$  преобразовался в граф  $G'$ . Изменение функции модульности будет иметь вид:

$$\Delta Q = Q(G') - Q(G) = 2(e_{kl} - a_k a_l).$$

Следовательно, изменение функции модульности  $\Delta Q$  определяется значениями параметров графа  $G$  и имеет низкую трудоемкость.

Данный алгоритм может использоваться как в интерактивном, так и в автоматическом режиме. В интерактивном режиме необходимо определить один сегмент, к которому принадлежит выбранная пользо-

вателем точка. В этом случае начальная вершина алгоритма выбирается пользователем и ищется один сегмент. В автоматическом режиме необходимо сегментировать все изображение. Поэтому начальная вершина алгоритма выбирается случайным образом.

### 3. Компьютерный эксперимент

Эксперимент проводился как на искусственных изображениях, так и на фотографиях различных объектов. Рассмотрим результаты работы алгоритма на искусственных изображениях с четкими контурами и равномерной заливкой (рис. 1).

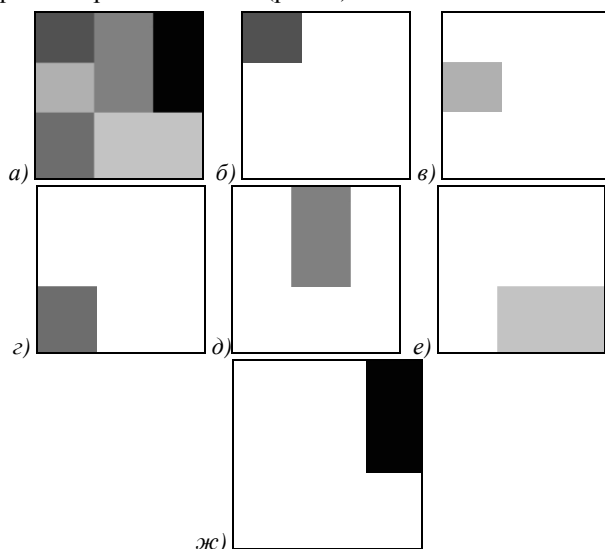


Рис. 1. Результаты сегментации искусственного изображения с равномерной заливкой: а) исходное изображение, б-ж) выделенные сегменты

Как хорошо видно из рисунка, алгоритм абсолютно точно разделяет изображение на области, причем результат не зависит от выбора значения параметра  $h$ . Рассмотрим результаты работы алгоритма на искусственном изображении с градиентной заливкой (рис. 2).

Как видно из рис. 2, при  $h = 0,04$  внутренний круг с градиентной заливкой определяется как один сегмент, а при  $h = 0,03$  он разбивается на четыре сегмента. При уменьшении значения  $h$  круг с градиентной заливкой разбивается на все большее количество сегментов. Таким образом, выбор параметра  $h$  влияет на величину изменения интенсивности цвета, которая определяется как граница между сегментами.

На рис. 3 приведены наиболее крупные сегменты, получаемые на фотографическом изображении «Перцы» при  $h = 0,05$ . Общее количество выделяемых сегментов равно 174.

При уменьшении параметра  $h$  увеличивается количество сегментов, на которые разбивается изображение. Выделим отдельный сегмент, например, рис. 3г и проследим его разбиение при уменьшении параметра  $h$  от значения 0,05 до значения 0,04. Результат представлен на рис. 4.

На рис. 5 приведен график зависимости количества сегментов, на которые разбивается изображение «Перцы», в зависимости от параметра  $h$ .

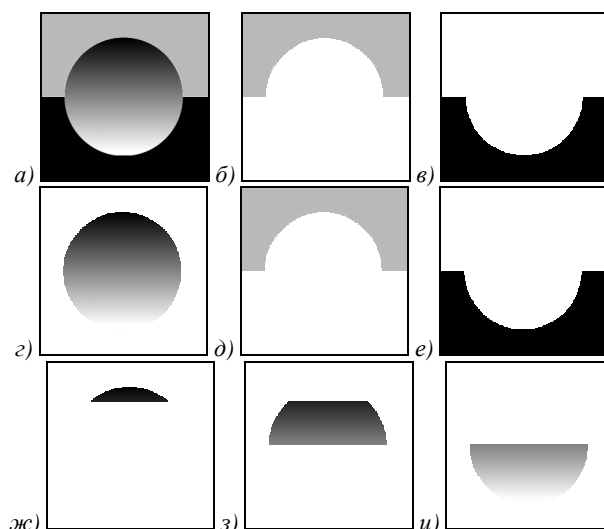


Рис. 2. Результат сегментации искусственного изображения с градиентной заливкой (размеры изображения  $256 \times 256$ ): а) исходное изображение, б-г) сегменты, выделенные при  $h = 0,04$ , д-и) сегменты, выделенные при  $h = 0,03$

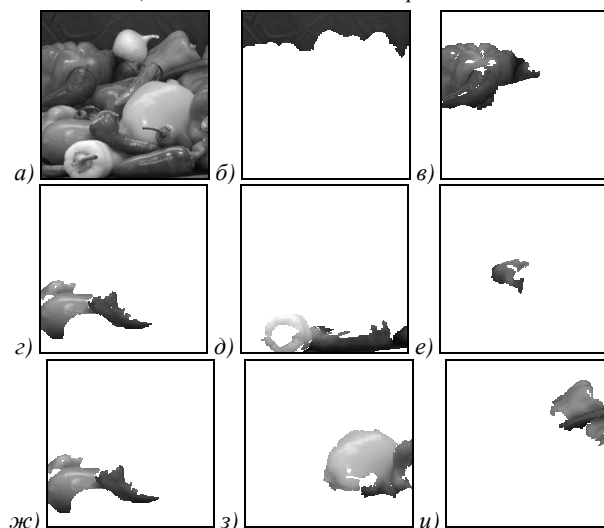


Рис. 3. Сегментация фотографического изображения «Перцы» при  $h = 0,05$ : а) исходное изображение, б-и) наиболее крупные сегменты

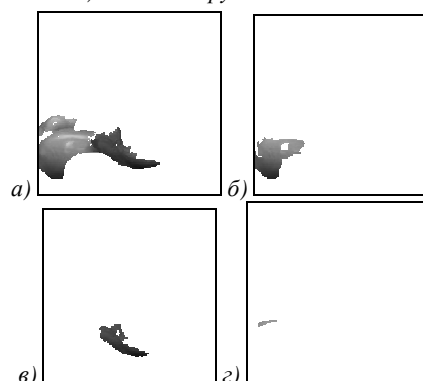


Рис. 4. Разбиение сегмента на более мелкие сегменты при переходе от  $h = 0,05$  к  $h = 0,04$ : а) один из сегментов, выделенных на изображении «Перцы» при  $h = 0,05$ , б-г) наиболее крупные сегменты, на которые разбивается исходный сегмент при  $h = 0,04$

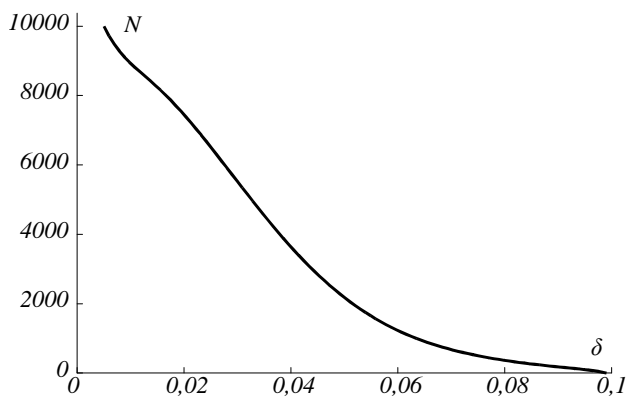


Рис. 5. График зависимости количества сегментов  $n$ , на которые разбивается изображение «Перцы», от параметра  $h$

Как видно из рис. 5, количество сегментов достаточно быстро убывает с ростом значения  $h$ , что свидетельствует о высокой чувствительности предложенного метода к выбору данного параметра. Начиная со значения  $h=0,01$  количество сегментов становится равным 1, то есть всё изображение воспринимается как один сегмент.

Проведем исследование предложенного алгоритма на устойчивость к наличию случайно поврежденных пикселей. Компьютерный эксперимент состоял в том, что  $p$  процентов случайно выбранных пикселей изображения заменялись пикселями случайного цвета. После чего осуществлялась сегментация. При сегментации не учитывались сегменты размером в один пиксель. Результаты эксперимента для изображения «Перцы» при  $p=10\%$  и  $h=0,05$  приведены на рис. 6.

Как видно из рис. 6, наличие поврежденных пикселей не влияет на выделение крупных сегментов, так как подавляющее большинство поврежденных пикселей идентифицируется как кластеры размером в один пиксель. Данный результат является ожидаемым, так как логически вытекает из предложенного алгоритма. При более сильном зашумлении изображения поврежденные пиксели могут оказаться ближайшими соседями и образовывать отдельный сегмент. Предложенный алгоритм легко адаптируется к данной ситуации и может быть настроен на игнорирование сегментов меньше любого заранее определенного размера.

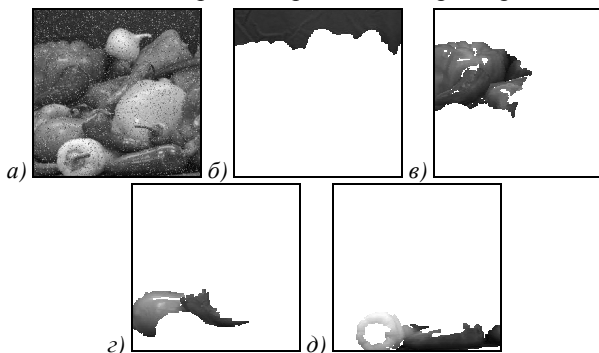


Рис. 6. Результат сегментации фотографического изображения «Перцы», поврежденного на 10% при  $h=0,05$ : а) зашумленное изображение, б-д) наиболее крупные сегменты

#### 4. Обсуждение результатов и выводы

Таким образом, предложенный подход, основанный на жадном алгоритме поиска сообществ на графах, позволяет проводить сегментацию изображений с контролируемым уровнем детализации. Размеры сегментов определяются единственным параметром алгоритма. Следует отметить, что разработанный алгоритм тестировался нами не только на монохромных изображениях, представленных в статье, но и на цветных рисунках и фотографиях и продемонстрировал хорошие результаты.

В сравнении с другими методами сегментации следует отметить, что предложенный подход использует разность интенсивностей цветов соседних пикселей, характерную для дифференциальных методов сегментации. Однако получаемые сегменты имеют четкие границы, которые свойственны методам, основанным на абсолютном значении интенсивности, например, методу водораздела. Дополнительной положительной характеристикой предложенного алгоритма является устойчивость к импульсному шуму, которая состоит в малом влиянии испорченных пикселей на внешние границы крупных сегментов. Большинство алгоритмов сегментации приводят к усилению импульсного шума и требуют предварительной обработки изображения сглаживающими фильтрами, приводящей к размытию границ сегментов.

Как уже было сказано во введении, поиск сообществ на графах в проведенных ранее работах использовался только на завершающем этапе для объединения сегментов, выделенных каким-либо алгоритмом. Предложенный алгоритм не требует использования дополнительных методов сегментации или построения суперпикселей.

#### Литература

1. **Park, I.K.** Color image retrieval using hybrid graph representation / I.K. Park, I.D. Yun, S.U. Lee // Image and Vision Computing. – 1999. – Vol. 17, Issue 7. – P. 465-474. – DOI: 10.1016/S0262-8856(98)00139-5.
2. **Xu, B.** EMR: A scalable graph-based ranking model for content-based image retrieval / B. Xu, J. Bu, C. Chen, C. Wang, D. Cai, X. He // IEEE Transactions on knowledge and data engineering. – 2015. – Vol. 27, Issue 1. – P. 102-114. – DOI: 10.1109/TKDE.2013.70.
3. **Johnson, J.** Image retrieval using scene graphs / J. Johnson, R. Krishna, M. Stark, L.-J. Li, D. Shamma, M. Bernstein, L. Fei-Fei // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. – 2015. – P. 3668-3678. – DOI: 10.1109/CVPR.2015.7298990.
4. **Белим, С.В.** Использование алгоритма кластеризации для разбиения изображения на односвязные области / С.В. Белим, П.Е. Кутлуни // Наука и образование: электронное научно-техническое издание. – 2015. – № 3. – URL: <http://technomag.bmstu.ru/doc/759275.html>. – DOI: 10.7463/0315.0759275.
5. **Белим, С.В.** Выделение контуров на изображениях с помощью алгоритма кластеризации / С.В. Белим, П.Е. Кутлуни // Компьютерная оптика. – 2015. – Т. 39, № 1. – С. 119-124. – DOI: 10.18287/0134-2452-2015-39-1-119-124.
6. **Newman, M.E.J.** Analysis of weighted networks / M.E. Newman // Physical Review E. – 2004. – Vol. 70, Issue 5 Pt 2. – 056131. – DOI: 10.1103/PhysRevE.70.056131.

7. **Cuadros, O.** Segmentation of large images with complex networks / O. Cuadros, G. Botelho, F. Rodrigues, J.B. Neto // SIBGRAPI '12, Proceedings of the 2012 25<sup>th</sup> SIBGRAPI Conference on Graphics, Patterns and Images. – 2012. – P. 24-31. – DOI: 10.1109/SIBGRAPI.2012.13.
8. **Clauset, A.** Finding community structure in very large networks / A. Clauset, M.E.J. Newman, C. Moore // Physical Review E. – 2004. – Vol. 70, Issue 6 Pt 2. – 066111. – DOI: 10.1103/PhysRevE.70.066111.
9. **Raghavan, U.S.** Near linear time algorithm to detect community structures in large-scale networks / U.S. Raghavan, R. Albert, S. Kumara // Physical Review E. – 2007. – Vol. 76(3 Part 2). – P. 036106. – DOI: 10.1103/PhysRevE.76.036106.
10. **Ren, X.** Learning a classification model for segmentation / X. Ren, J. Malik // Proceedings on the Ninth IEEE International Conference on Computer Vision. – 2003. – Vol. 2. – P. 10-17. – DOI: 10.1109/ICCV.2003.1238308.
11. **Levinshstein, A.** TurboPixels: Fast superpixels using geometric flows / A. Levinshstein, A. Stere, K.N. Kutulakos, D.J. Fleet, S.J. Dickinson, K. Siddigi // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2009. – Vol. 31, Issue 12. – P. 2290-2297. – DOI: 10.1109/TPAMI.2009.96.
12. **Achanta, R.** Slic superpixels / R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, S. Susstrunk // EPFL Technical Report 149300, June 2010.
13. **Cigla, C.** Efficient graph-based image segmentation via speeded-up turbo pixels / C. Cigla, A.A. Alatan // 17<sup>th</sup> International Conference on Image Processing. – 2010. – P. 3013-3016. – DOI: 10.1109/ICIP.2010.5653963.
14. **Mourchid, Y.** New image segmentation approach using community detection algorithms / Y. Mourchid, M.E. Hassouni, H.A. Cherif // International Journal of Computer Information Systems and Industrial Management Applications. – 2016. – Vol. 8. – P. 195-204.
15. **Nair, R.S.** Modularity based color image segmentation / R.S. Nair, K.V. Vineetha // IJIREEICE. – 2016. – Vol. 3, Special Issue 1. – P. 109-113.
16. **Pourian, P.** Weakly supervised graph based semantic segmentation by learning communities of image-parts / P. Pourian, S. Karthikeyan, B.S. Manjunath // The IEEE International Conference on Computer Vision (ICCV). – 2015. – P. 1359-1367. – DOI: 10.1109/ICCV.2015.160.
17. **Li, S.** Modularity-based image segmentation / S. Li, D.O. Wu // IEEE Transactions on Circuits and Systems for Video Technology. – 2015. – Vol. 25, Issue 4. – P. 570-581. – DOI: 10.1109/TCSVT.2014.2360028.
18. **Verma, S.** An increased modularity based contour detection / S. Verma, A. Chugh // International Journal of Computer Applications. – 2016. – Vol. 135, Issue 12. – P. 41-44. – DOI: 10.5120/ijca2016908588.
19. **Browet, A.** Community detection for hierarchical image segmentation / A. Browet, P.-A. Absil, P. Van Dooren // IWCI'11 Proceedings of the 14<sup>th</sup> International Conference on Combinatorial Image Analysis. – 2011. – P. 358-371. – DOI: 10.1007/978-3-642-21073-0\_32.
20. **Newman, M.E.J.** Mixing patterns in networks / M.E.J. Newman // Physical Review E. – 2003. – Vol. 67(2). – P. 026126. – DOI: 10.1103/PhysRevE.67.026126.
21. **Fortunato, S.** Community detection in graphs / S. Fortunato // Physics Reports. – 2010. – Vol. 486(3-5). – P. 75-174. – DOI: 10.1016/j.physrep.2009.11.002.

#### Сведения об авторах

**Белим Сергей Викторович**, д.ф.-м.н., профессор, заведующий кафедрой информационной безопасности Омского государственного университета им. Ф.М. Достоевского. Область научных интересов: обработка изображений, интеллектуальный анализ данных, системы защиты информации. E-mail: [sbelim@mail.ru](mailto:sbelim@mail.ru).

**Ларионов Станислав Борисович**, аспирант кафедры информационной безопасности Омского государственного университета им. Ф.М. Достоевского. Область научных интересов: компьютерное зрение, криптография, программирование. E-mail: [me@stas-larionov.ru](mailto:me@stas-larionov.ru).

ГРНТИ: 28.23.15

Поступила в редакцию 6 апреля 2016 г. Окончательный вариант – 7 сентября 2016 г.

## AN ALGORITHM OF IMAGE SEGMENTATION BASED ON COMMUNITY DETECTION IN GRAPHS

S.V. Belim<sup>1</sup>, S.B. Larionov<sup>1</sup>

<sup>1</sup> F.M. Dostoevskiy Omsk State University, Omsk, Russia

### Abstract

This article suggests an algorithm of image segmentation based on the community detection in graphs. The image is represented as a non-oriented weighted graph on which the community detection is carried out. Each pixel of the image is associated with a graph vertex. Only adjacent pixels are connected by edges. The weight of the edge is defined by subtracting the intensities of three color components of pixels. A Newman modularity function is used to check the quality of the graph partition into sub-graphs. It is suggested that a greedy algorithm should be applied to solving the image segmentation problem. Each community corresponds to a segment in the image. A computer experiment was carried out. The influence of the algorithm parameter to the segmentation results was revealed. The proposed algorithm was shown to be insensitive to random impulse noise.

**Keywords:** community detection, image segmentation.

**Citation:** Belim SV, Larionov SB. An algorithm of image segmentation based on community detection in graphs. Computer Optics 2016; 40(6): 904-910. DOI: 10.18287/2412-6179-2016-40-6-904-910.

**References**

- [1] Park IK, Yun ID, Lee SU. Color image retrieval using hybrid graph representation. *Image and Vision Computing* 1999; 17(7): 465-474. DOI: 10.1016/S0262-8856(98)00139-5.
- [2] Xu B, Bu J, Chen C, Wang C, Cai D, He X. EMR: A scalable graph-based ranking model for content-based image retrieval. *IEEE Transactions on knowledge and data engineering* 2015; 27(1): 102-114. DOI: 10.1109/TKDE.2013.70.
- [3] Johnson J, Krishna R, Stark M, Li L-J, Shamma D, Bernstein M, Fei-Fei L. Image retrieval using scene graphs. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 2015: 3668-3678. DOI: 10.1109/CVPR.2015.7298990.
- [4] Belim SV, Kutlunin PE. Usage of clustering algorithm to segment image into simply connected domains. *Science & Education BMSTU* 2015; 15(3). Source: <http://technomag.bmstu.ru/doc/759275.html>. DOI: 10.7463/0315.0759275.
- [5] Belim SV, Kutlunin PE. Boundary extraction in images using a clustering algorithm. *Computer Optics* 2015; 39(1): 119-124. DOI: 10.18287/0134-2452-2015-39-1-119-124.
- [6] Newman MEJ. Analysis of weighted networks. *Physical Review E* 2004; 70(5 Pt 2): 056131. DOI: 10.1103/PhysRevE.70.056131.
- [7] Cuadros O, Botelho G, Rodrigues F, Neto JB. Segmentation of large images with complex networks. *SIBGRAPI* 2012: 24-31. DOI: 10.1109/SIBGRAPI.2012.13.
- [8] Clauset A, Newman MEJ, Moore C. Finding community structure in very large networks. *Physical Review E* 2004; 70(6 Pt 2): 066111. DOI: 10.1103/PhysRevE.70.066111.
- [9] Raghavan US, Albert R, Kumara S. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E* 2007; 76(3 Pt 2): 036106.
- [10] Ren X, Malik J. Learning a classification model for segmentation. *ICCV* 2003; 2: 10-17. DOI: 10.1109/ICCV.2003.1238308.
- [11] Levinstein A, Stere A, Kutulakos KN, Fleet DJ, Dickinson SJ, Siddigi K. TurboPixels: Fast superpixels using geometric flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2009; 31(12): 2290-2297. DOI: 10.1109/TPAMI.2009.96.
- [12] Achanta R, Shaji A, Smith K, Lucchi A, Fua P, Susstrunk S. Slic superpixels. EPFL Technical Report 149300, June 2010.
- [13] Cigla C, Alatan AA. Efficient graph-based image segmentation via speeded-up turbo pixels. *ICIP '17* 2010: 3013-3016. DOI: 10.1109/ICIP.2010.5653963.
- [14] Mourchid Y, Hassouni ME, Cherif HA. New Image Segmentation Approach using Community Detection Algorithms. *International Journal of Computer Information Systems and Industrial Management Applications* 2016; 8: 195-204.
- [15] Nair RS, Vineetha KV. modularity based color image Segmentation. *IJIREEICE* 2016; 3(Spec 1): 109-113.
- [16] Pourian P, Karthikeyan S, Manjunath BS. Weakly supervised graph based semantic segmentation by learning communities of image-parts. *The IEEE International Conference on Computer Vision (ICCV)* 2015: 1359-1367. DOI: 10.1109/ICCV.2015.160.
- [17] Li S, Wu DO. Modularity-based image segmentation. *IEEE Transactions on Circuits and Systems for Video Technology* 2015; 25(4): 570-581. DOI: 10.1109/TCSVT.2014.2360028.
- [18] Verma S, Chugh A. An increased modularity based contour detection. *International Journal of Computer Applications* 2016; 135(12): 41-44. DOI: 10.5120/ijca2016908588.
- [19] Browet A, Absil P-A, Van Dooren P. Community Detection for hierarchical image segmentation. *IWCIA'11* 2011: 358-371. DOI: 10.1007/978-3-642-21073-0\_32.
- [20] Newman MEJ. Mixing patterns in networks. *Phys Rev E* 2003; 67(2): 026126. DOI: 10.1103/PhysRevE.67.026126.
- [21] Fortunato S. Community detection in graphs. *Physics Reports* 2010; 486(3-5): 75-174. DOI: 10.1016/j.physrep.2009.11.002.

**Authors' information**

**Sergey Viktorovich Belim**, Doctor in Physic and Mathematic, professor, chairman of Information Security department of Omsk State University n.a. F.M. Dostoevskiy. His research interests are image processing, intelligent data analysis, information security systems. E-mail: [sbelim@mail.ru](mailto:sbelim@mail.ru).

**Stanislav Borisovich Larionov**, postgraduate student in Information Security department of Omsk State University n.a. F.M. Dostoevskiy. His research interests are computer vision, cryptography, and programming. E-mail: [me@stas-larionov.ru](mailto:me@stas-larionov.ru).

*Received April 6, 2016. The final version – September 7, 2016.*