

## Точный и быстрый алгоритм вычисления геодезического расстояния для суперпиксел-сегментации

М.Г. Мозеров\*, В.Н. Карнаухов\*, В.И. Кобер\*, Л.В. Зимина\*\*

\* *Институт проблем передачи информации, Российская академия наук, Москва, 127051, Россия*

\*\* *Московский политехнический университет, Москва, 107023, Россия*

Поступила в редколлегию 18.09.2023

**Аннотация**—Моделирование в аффинном пространстве на основе геодезического расстояния позволяет реализовывать важные методы компьютерного зрения. Одним из таких применений является суперпиксел-сегментация, где необходимо вычислить геодезическое расстояние от центра заданных сегментов до произвольной точки на изображении. Однако до сих пор предлагавшиеся алгоритмы вычисления таких расстояний для задач сегментации представляли собой эвристические, итеративные подходы, не гарантирующие ожидаемого результата. В этой статье мы предлагаем новый быстрый алгоритм вычисления геодезического расстояния и доказываем, что такой алгоритм точный, получаемые в результате применения этого алгоритма сегменты изображения односвязные области. То есть алгоритм позволяет получать на выходе односвязные суперпиксели, в отличие от многих других методов, основанных на использовании пространственной близости геодезического расстояния и требующих дополнительной корректировки. Предложенный метод превосходит аналоги по эффективности распознавания границ и по скорости вычисления.

**КЛЮЧЕВЫЕ СЛОВА:** Суперпиксель-сегментация, быстрые алгоритмы, вычисление точного геодезического расстояния.

DOI: 10.53921/18195822\_2023\_23\_3\_385

### 1. ВВЕДЕНИЕ

Алгоритмы на основе геодезического расстояния позволяют реализовывать важные методы компьютерного зрения. Одним из таких направлений является суперпиксел-сегментация, где необходимо вычислить геодезическое расстояние от центра заданных сегментов до произвольной точки на изображении. С другой стороны, целью суперпиксел-сегментации является создание согласованной группы пикселей на изображении, которая также известна как избыточная сегментация изображения [1, 2, 3]. В последнее время суперпиксел-сегментации уделяется большое внимание, особенно в области компьютерного зрения, поскольку она обеспечивает удобный способ вычисления характеристик изображения и снижает сложность последующих задач обработки изображений. Для решения задачи избыточной сегментации было предложено множество алгоритмов [3, 4, 5, 6, 7, 8, 9, 10].

Ценность суперпиксел-сегментации заключается в том, что этот подход находит применение во многих областях компьютерного зрения, включая распознавание объектов [11], сегментацию изображений [12], отслеживание объектов [13], сегментацию видео [14] и классификацию [15]. Это обусловлено тем, что суперпиксел по сути является локальным структурным элементом изображения, или иначе локальным признаком, что позволяет использовать такой признак для дальнейшего анализа изображения.

В этой работе мы предлагаем метод суперпиксел-сегментации на основе моделирования в аффинном пространстве геодезического расстояния. Сам по себе такой подход не является новым, и существует множество публикаций, где описывается реализация подобных алгоритмов [16, 17, 6, 18]. Однако, предлагавшиеся в этих работах алгоритмы вычисления таких расстояний для задач сегментации представляли собой эвристические, итеративные подходы, не гарантирующие ожидаемого результата. Как правило, такой алгоритм за одну итерацию сканировал изображение из одного угла в противоположный и обратно, применяя локальный оператор аддитивного увеличения геодезического расстояния. До полной релаксации итеративного процесса обычно требуется число, пропорциональное среднему числу пикселей в суперпикселе. То есть алгоритм для суперпиксел-сегментации с размером 100 пикселей требует порядка 400 операций. В то время, как наш алгоритм требует только 8-10 операций на каждый пиксель. И это число не превышает 12 операций на пиксел вне зависимости от числа пикселей в суперпикселе. Поэтому наш алгоритм является быстрым с вычислительным порядком  $O(1)$ . Кроме того, в результате работы эвристических алгоритмов, описанных выше, возникают артефакты разделения одного сегмента на несколько несвязанных сегментов, что не соответствует условиям задачи. В основе нашего метода вычисления геодезического расстояния лежит классический алгоритм Форда-Беллмана (1962 год) [19], поэтому алгоритм точный, и получаемые в результате применения этого метода сегменты изображения — односвязные области. То есть алгоритм позволяет получать на выходе односвязные суперпиксели, в отличие от многих других методов, основанных на использовании пространственной близости геодезического расстояния и требующих дополнительной корректировки. Следует отметить, что вычислительная сложность алгоритма Форда-Беллмана  $O(4N^2)$ , где  $N$  число вершин-пикселей для четырёхсвязного регулярного графа изображений, а также начальная вершина, от которой вычисляются расстояния, только одна. Поэтому наш алгоритм является быстрой модификацией классического алгоритма.

Эффективность предложенного метода сравнивается и измеряется по критерию совпадения границ областей, а также с помощью визуальной иллюстрации. Анализ результата работы предложенного алгоритма показал, что метод показывает уровень state-of-the-art при этом скорость вычисления — одна из самых быстрых для данного типа алгоритмов и превосходит ранее предложенные алгоритмы, основанные на вычислении геодезического расстояния.

Статья организована следующим образом: в разделе 2 ставится задача сегментации на основе геодезического расстояния, в разделе 3 предложен новый алгоритм суперпиксел-сегментации, основанный на быстром вычислении геодезического расстояния, в разделе 4 приведены экспериментальные результаты и, наконец, раздел заключение суммирует наши выводы.

## 2. ПОСТАНОВКА ЗАДАЧИ

Представим изображение как регулярный граф:  $G = \{V, E\}$ , где вершины  $v_i \in V$  совпадают с пикселями изображения, а ребра  $e_{ij} \in E$ , составляют регулярную четырёхсвязную решетку, как изображено на Рис. 1. Тогда расстояние между соседними вершинами  $v_i$  и  $v_j$  задается как функция ребра  $e_{ij}$  равная квадрату частной производной функции яркости изображения  $\varphi(v_i)$ :

$$d(e_{ij}) = (\varphi(v_j) - \varphi(v_i))^2 + \delta, \quad (1)$$

где  $\delta$  — усредненное значение расстояний на ребрах графа изображения:

$$\delta = \frac{1}{|E|} \sum_{d(e_{ij}) \in E} e_{ij}. \quad (2)$$

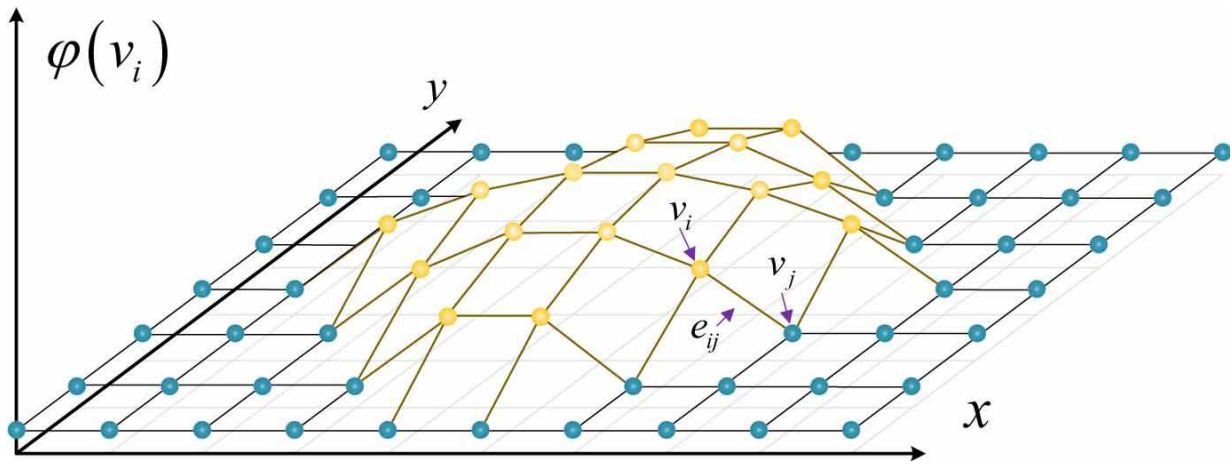


Рис. 1. Изображение как граф на регулярной четырёхсвязной решетке.

Важным термином для аффинного пространства геодезического расстояния является понятие пути между двумя вершинами  $p(v_i, v_j)$ , который определяется как конечная последовательность вершин и ребер:

$$p(v_i, v_j) = \{v_i, e_{i,i+1}, v_{i+1}, \dots, v_{j-1}, e_{j-1,j}, v_j\}. \tag{3}$$

Важно отметить, что для рассматриваемого аффинного пространства, вершины и ребра могут встречаться только один раз. Число возможных путей между двумя вершинами пропорционально экспоненте от числа ребер и длина произвольного пути  $|p(v_i, v_j)|$  определяется как:

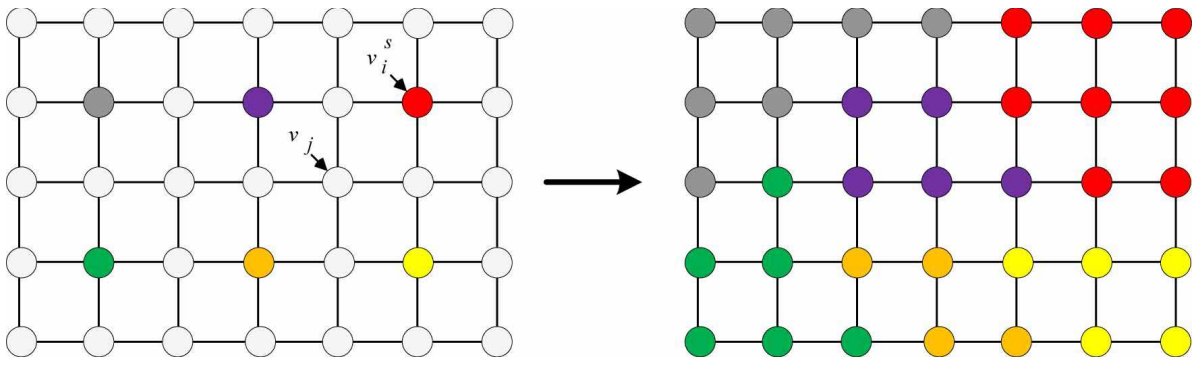
$$|p(v_i, v_j)| = \sum_{e_{k,l} \in p(v_i, v_j)} d(e_{k,l}). \tag{4}$$

Тогда расстояние между двумя произвольными вершинами  $|v_i, v_j|$  для аффинного пространства геодезического расстояния определяется как минимальная длина пути между этими вершинами:

$$|v_i, v_j| = \min \{|p_n(v_i, v_j)|\}. \tag{5}$$

В общем виде задача сегментации изображения  $\varphi(v_i)$  формулируется следующим образом: найти функцию  $s(v_i)$ , ставящую в соответствие каждому пикселю-вершине  $v_i$  некоторое натуральное число, означающее номер сегмента. При этом пиксел на изображении должен обладать некоторыми свойствами, по которым этот пиксел можно было бы отнести к тому или иному сегменту. Обычно, это некоторое расстояние в пространстве близости. Для суперпиксел-сегментации добавляется еще два важных условия: а) все пиксели одного сегмента должны образовывать односвязное множество; б) площадь суперпикселей должна быть примерно одинакова, а карта сегментов должна представлять из себя подобие регулярной сетки. Поэтому предполагается, что существует подмножество вершин  $v_j^s \in V^S$ , называемых центрами суперпикселей, геодезическое расстояние до которых определяет принадлежность всех остальных вершин к тому или иному сегменту, как это показано на Рис. 2. Таким образом для суперпиксел-сегментации на основе геодезического расстояния задача формулируется следующим образом: найти для всех вершин графа-изображения геодезическое расстояние до всех вершин-центров и, выбрав наименьшее, присоединить каждую вершину к множеству ближайшей вершины-центра  $v_j^s \in V^S$ :

$$s(v_i) = \arg \min_{v_j^s} \{|v_i, v_j^s|\}. \tag{6}$$



**Рис. 2.** Исходное распределение вершин-центров  $v_i^s \in V^S$  (левая часть рисунка) и результат сегментации (правая часть рисунка).

Предлагавшиеся до сих пор эвристические итеративные алгоритмы расчета расстояний представляли собой последовательное сканирование пикселей изображения по строкам, начиная с верхнего левого угла, с проверкой четырех соседних пикселей на предмет близости к одному из пиксел-центров, а затем сканирование проводилось в обратном направлении. Так как сканирование производилось последовательно, а не рекурсивно, то за одну итерацию алгоритм формировал приблизительную карту сегментации. Однако, чтобы достигнуть более или менее приемлемых результатов требуется десятки итераций. Более того, для полной итеративной сходимости требуются сотни итераций. Также не исключены варианты, когда полная сходимость невозможна. Кроме того, как правило, в результате такого алгоритма сегменты с одним номером сходились к несвязанным кластерам, что не является правильным решением задачи. С другой стороны, существуют два классических алгоритма расчета расстояний: Форда-Беллмана и Дейкстры, которые позволяют получить точное решение. Однако среди опубликованных методов, использующих геодезическое расстояние, эти алгоритмы не используются. Видимо, для этого существуют две причины.

Первая — сканирование изображения — привычная и простая парадигма для авторов, позволяющая получить относительно удовлетворительный результат за приемлемое время. Кроме того, артефакты, связанные с локальным вычислением расстояний — явление, присущее и другим методам суперпиксел-сегментации, поэтому авторы не ставили перед собой задачу получения точного решения.

Вторая причина — сами классические алгоритмы. Их прямая имплементация предполагает вычисление расстояния от одной вершины и требует квадратичной вычислительной сложности относительно числа вершин или ребер.

В этой работе мы предлагаем быстрый алгоритм, основанный на модификации метода Форда-Беллмана, позволяющий реализовать сегментацию со сложностью вычисления близкой к  $O(1)$ . Описание этого алгоритма дается в следующей секции.

### 3. БЫСТРЫЙ АЛГОРИТМ ВЫЧИСЛЕНИЯ ГЕОДЕЗИЧЕСКИХ РАССТОЯНИЙ

Классические алгоритмы Форда-Беллмана и Дейкстры для графов, где длины ребер — константа, вырождаются в два классических алгоритма теории графов *breadth-first search* и *depth-first search* соответственно. Вычислительная сложность обоих алгоритмов  $O(1)$  — пропорциональна числу вершин. Однако *depth-first search* (эквивалент метода Дейкстры для константной длины ребер) разработан строго для одной исходной вершины, и тривиальное расширение на множество таких вершин приводит к необходимости умножать вычислительную сложность такого алгоритма на число исходных вершин с увеличением задействованной памяти.

ти компьютера в разы. Нетривиальное расширение практически переводит алгоритм в разряд depth-first search. Для того чтобы описать наш быстрый алгоритм, необходимо ввести понятие соседства вершины, представляющего из себя совокупность ребер, исходящих из данной вершины, и вершин, с которыми данную вершину соединяют эти ребра:  $\mathcal{N}(v_i) = \{\{e_{i,j}\}, \{v_j\}\}$ . А также несколько новых функций, определенных на множестве вершин графа-изображения. Суммарная длина:

$$\sigma^{t=0}(v_i) = \begin{cases} 0 & |v_i \in V^s \\ \infty & |v_i \notin V^s, \end{cases} \quad (7)$$

$$\sigma^{t \neq 0}(v_{j(\mathcal{N}(v_i))}) = \begin{cases} \sigma(v_i) + e_{i,j(\mathcal{N}(v_i))} & | \sigma(v_i) + e_{i,j(\mathcal{N}(v_i))} < \sigma(v_{j(\mathcal{N}(v_i))}) \\ \sigma(v_{j(\mathcal{N}(v_i))}) & | \text{elsewhere,} \end{cases}$$

где  $V^s$  — множество исходных точек или центров суперпикселей (как пример, шесть разноцветных вершин на левом графе Рис. 2). Функция в (7) является вспомогательной и промежуточной при работе алгоритма, однако представляет собой геодезическое расстояние до ближайшей вершины-центра. Функция суммарной длины представляет собой оператор, который присваивает соседним вершинам величину суммарной длины заданной вершины плюс длину ребра, соединяющую обе вершины в случае, если ранее установленная величина суммарной длины меньше вычисленного значения.

Для того, чтобы не делать лишних вычислений, необходимо определить функцию активности вершины  $a(v_i)$ :

$$a^{t=0}(v_i) = \begin{cases} v_i & |v_i \in V^s \\ \emptyset & |v_i \notin V^s, \end{cases} \quad (8)$$

$$a^{t \neq 0}(v_i) = \begin{cases} v_i & | \sigma^{t-1}(v_i) \neq \sigma^t(v_i) \\ \emptyset & | \sigma^{t-1}(v_i) = \sigma^t(v_i) \end{cases},$$

где  $t$  — номер шага итерации. Смысл данной функции простой: если суммарная длина для данной вершины  $v_i$  не изменилась на предыдущем шаге итерации, то эта вершина не может быть источником желаемого изменения суммарных длин вершин окрестности  $v_{j(\mathcal{N}(v_i))}$  данной вершины  $v_i$  на следующем шаге итерации, и ее не надо рассматривать на следующем шаге. Иначе мы должны допустить, что, как минимум, одно из значений суммарных длин вершин этой окрестности увеличилось, что противоречит сущности оператора функции (7). Такая стратегия полностью повторяет стратегию алгоритма depth-first search.

Наконец, мы определяем самую важную функцию — определение номера сегмента  $s(v_i)$ .

$$s^{t=0}(v_i) = \begin{cases} s & |v_i \in V^s \\ \emptyset & |v_i \notin V^s, \end{cases} \quad (9)$$

$$s^{t \neq 0}(v_{j(\mathcal{N}(v_i))}) = \begin{cases} s(v_i) & | \sigma(v_i) + e_{i,j(\mathcal{N}(v_i))} < \sigma(v_{j(\mathcal{N}(v_i))}) \\ s(v_{j(\mathcal{N}(v_i))}) & | \text{elsewhere.} \end{cases}$$

Эта формула вторична к уравнениям (7) и (8), так как устанавливает до какой вершины-центра текущее промежуточное геодезическое расстояние минимальное.

Итак, полный алгоритм суперпиксел-сегментации состоит из следующих шагов:

- 1 Начальная инициализация всех функций  $\{\sigma^{t=0}(v_i), a^{t=0}(v_i), s^{t=0}(v_i)\}$   $t = 0$  для всех  $v_i \in V$  с помощью (7)-(9). Сформировать множество активных вершин для следующего шага итерации

$$V^a = \bigcup_{v_i \in V} a(v_i)$$

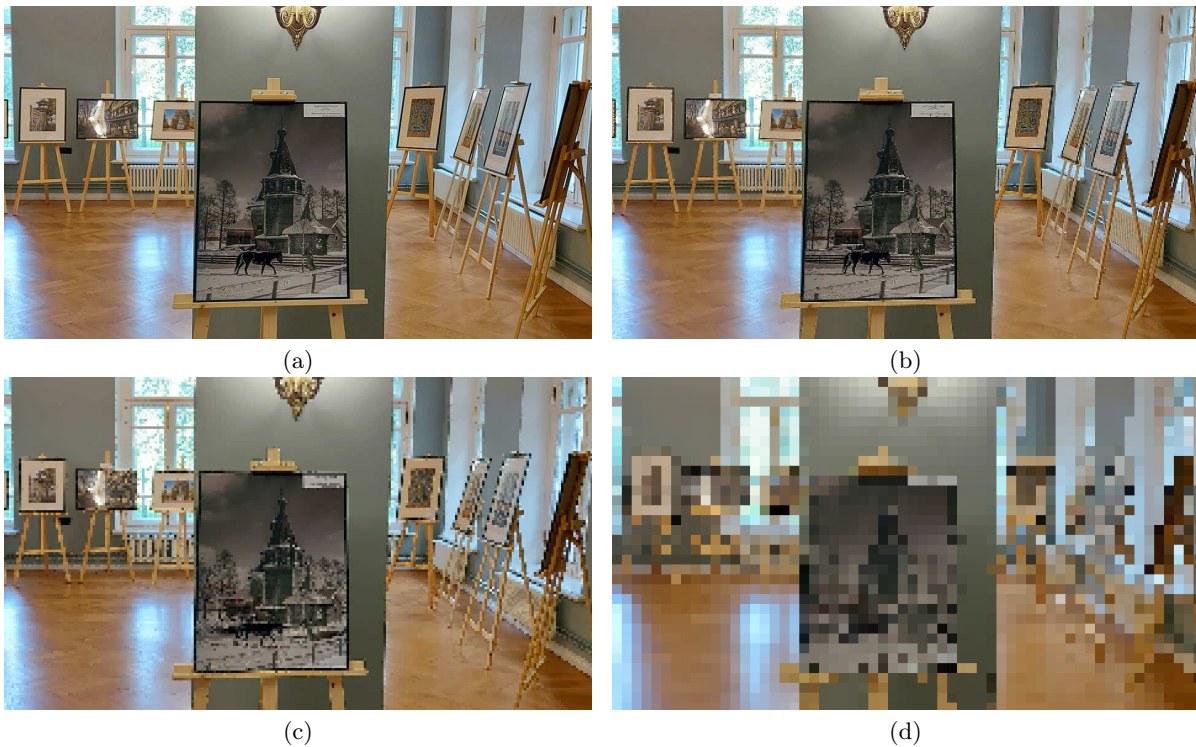
- 2 Вычисление всех функций  $\{\sigma^{t \neq 0}(v_i), a^{t \neq 0}(v_i), s^{t \neq 0}(v_i)\}$  для всех  $v_i \in V_{t-1}^a$  с помощью (7)-(9). Сформировать множество активных вершин для следующего шага итерации

$$V_t^a = \bigcup_{v_i \in V_{t-1}^a} a^t(v_i)$$

- 3 Если на шаге  $t$  выполняется условие  $V_t^a = \emptyset$ , то итерационный процесс останавливается и решение  $s^t(v_i)$  полагается точным решением задачи сегментации.

Доказательство того, что такое решение действительно является точным, практически полностью совпадает с аналогичными доказательствами алгоритмов Дейкстры и Форда-Беллмана.

В силу того, что наш алгоритм на каждом шаге итерации использует только активные вершины, число операций на пиксел в результате сильно сокращается. Так если средняя площадь суперпикселя составляет 9 пикселей, то среднее число итераций для обычных изображений составляет 32. То есть число операций на полный цикл сегментации пропорционально  $32 \cdot |V|$ , без использования функции активности, а с использованием —  $1.4 \cdot |V|$ . Для средней площади суперпикселя 2500 вершин эти цифры  $120 \cdot |V|$  и  $4 \cdot |V|$  соответственно. То есть ускорение алгоритма в 20-25 раз. Однако и эту цифру ускорения можно увеличить примерно в два-три раза с использованием пирамидального подхода. Суть подхода очень проста. Сначала мы сегментируем изображение с суперпикселями площадью от 4 до 16 пикселей. Такие суперпиксели назовем стракселами. Усреднив значения яркости по стракселю, создаем новое регулярное изображение, где каждый пиксел имеет значение яркости равное среднему значению по стракселю. Такое масштабирование исходного изображения показано на Рис. 3). В результате



**Рис. 3.** Пирамидальное масштабирование исходного изображения с помощью страксел-сегментации. (a) — исходное изображение; (b) — с размером стракселя размером  $2 \times 2$  пиксела; (c) — с размером стракселя размером  $4 \times 4$  пиксела; (d) — с размером стракселя размером  $16 \times 16$  пиксела.

мы получаем изображение, в котором число пикселей меньше, чем в исходном. Поэтому для

**Таблица 1.** Результаты сравнения по критериям отношения precision-recall нашего метода с четырьмя наиболее известными алгоритмами суперпиксел- сегментации: N-cut [1], TPS [25], SEEDS [24] и SLIC [23].

Алгоритмы:	N-cut [1]	TPS [25]	SEEDS [24]	SLIC [23]	Наш метод
$S = 300$ Precision	0.362	0.331	0.273	<u>0.269</u>	<b>0.261</b>
$S = 300$ Recall	0.875	0.814	<b>0.937</b>	0.932	<u>0.936</u>
$S = 600$ Precision	0.340	0.310	<b>0.236</b>	0.238	<u>0.236</u>
$S = 600$ Recall	0.962	0.843	0.977	<u>0.981</u>	<b>0.985</b>
$S = 1000$ Precision	0.288	0.287	0.225	<b>0.213</b>	<u>0.217</u>
$S = 1000$ Recall	<u>0.989</u>	0.981	0.986	<u>0.989</u>	<b>0.992</b>

сегментации масштабированного изображения требуется гораздо меньше операций, чем для исходного.

Описание алгоритма будет неполным, если мы не опишем допустимую в нашем методе предобработку сегментируемого изображения. Для улучшения точности совпадения границ сегментов с границами на изображении это изображение может фильтроваться с использованием свертки с ядром геодезического расстояния. За основу мы взяли реализацию фильтра, предложенную в работах [20, 21]. Параметры фильтра равны в этом случае  $\sigma_c = 3$  и  $\sigma_r = 3$  (обозначения параметров фильтра взяты из цитируемых статей).

#### 4. ЭКСПЕРИМЕНТАЛЬНЫЕ РЕЗУЛЬТАТЫ

Экспериментальная часть была задумана таким образом, чтобы показать основные достоинства предлагаемого алгоритма. Экспериментальная секция нашей работы поделена на две части где:

- мы анализируем численные результаты сегментации на основе критериев точность-отзыв.
- часть результатов экспериментов представлена в иллюстрациях статьи, чтобы читатель мог сравнить качество суперпиксел-сегментации.

В нашей работе численные оценки базируются на двух стандартных метриках  $PR$  — точность (precision) и  $RC$  — отзыв (recall), которые вычисляются по известным формулам:

$$\begin{aligned} RC(S, G) &= \frac{TP(S, G)}{TP(S, G) + FN(S, G)}, \\ PR(S, G) &= \frac{TP(S, G)}{TP(S, G) + FP(S, G)}, \end{aligned} \quad (10)$$

где  $S$  — множество всех граничных пикселей, полученных в результате сегментации,  $G$  — множество пикселей истинных границ (ground truth),  $TP$  — true positives,  $FP$  — false positives,  $FN$  — false negatives.

Большинство алгоритмов суперпиксел-сегментации оцениваются на базе данных университета Berkeley [22]. Для объективной оценки мы использовали алгоритмы суперпиксел-сегментации, исходный код которых или двоичные коды находятся в открытом доступе. Мы выбрали четыре наиболее известных метода суперпиксел-сегментации: SLIC [23], SEEDS [24], TPS [25] и N-cut [1]. В Таблице 1 показаны результаты этих методов в сравнении с предлагаемым алгоритмом.

##### 4.1. Эксперименты с численным анализом результатов

В этой части экспериментальной секции результаты, полученные с помощью предложенного метода, сравниваются с результатами четырех наиболее известных алгоритмов суперпиксель-сегментации: N-cut [1], TPS [25], SEEDS [24] и SLIC [23] на основе критериев precision-recall.

Основным достижением нашей работы мы считаем быстрый алгоритм. В самом деле, эвристические алгоритмы в работах по суперпиксел-сегментации на основе геодезического расстояния предложенные ранее, требуют в 20-25 раз больше вычислительных операций на пиксел, чем предложенный в этой работе алгоритм. Кроме того, точность соответствия сегментации границам на изображении выше, чем у аналогичных эвристических. В силу того, что наш алгоритм вычисляет точные расстояния и сегменты являются связанными областями.

Тем не менее, анализируя результат в Таблице 1, мы можем сказать, что предложенный в статье алгоритм значительно превосходит три из четырех алгоритмов и показывает примерно равные результаты с методом SEEDS [24], а значит этот алгоритм не только быстрый, но и точный.

#### *4.2. Результаты экспериментов для визуального анализа*

В этой части экспериментальной секции на Рис. 4, Рис. 5 и Рис. 6 представлены результаты сегментации при различных параметрах алгоритма для визуального анализа.

Верхняя строка представляет сегментацию, полученную после прямой реализации алгоритма, средняя — с предварительной фильтрацией изображения и нижняя строка — при пирамидальной реализации алгоритма с размером строкселя  $4 \times 4$ . Из рисунков мы можем видеть, что там, где границы сегментов совпадают с границами объектов на изображении, все три реализации метода работают примерно одинаково. Отличие заметно лишь в областях с отсутствием значимых границ на изображении. Для таких областей предпочтительной является прямая реализация алгоритма с предварительной фильтрацией изображения, так как в этом случае границы сегментов ближе всего к прямой линии, а следовательно, уменьшается число ложных граничных пикселей. Однако в этом случае требуется в два раза больше вычислительных операций, так как фильтрация с ядром геодезического расстояния затрачивает примерно такое же время, как и на основной процесс сегментации. Поэтому необходимо выбирать, что более важно для пользователя алгоритма - быстрота вычислений или точность. То же самое относится и к пирамидальной реализации алгоритма. Качество соответствия границам хуже в этом случае, но не на много, однако алгоритм ускоряется значительно — более чем в два раза.

## ЗАКЛЮЧЕНИЕ

В этой статье мы предлагаем новый, быстрый алгоритм вычисления геодезического расстояния и доказываем, что такой алгоритм точный. Получаемые в результате применения этого алгоритма сегменты изображения односвязные области. Эвристические алгоритмы в работах по суперпиксел-сегментации на основе геодезического расстояния, предложенные ранее, требуют в 20-25 раз больше вычислительных операций на пиксел, чем предложенный в этой работе алгоритм. Кроме того, точность соответствия сегментации границам на изображении выше чем у аналогичных эвристических. В силу того, что наш алгоритм вычисляет точные расстояния, сегменты являются связанными областями. То есть алгоритм позволяет получать на выходе односвязные суперпиксели, в отличие от многих других методов, основанных на использовании пространственной близости геодезического расстояния и требующих дополнительной корректировки. Предложенный метод превосходит аналоги по эффективности распознавания границ и по скорости вычисления.





**Рис. 4.** Визуальный результат сегментации предложенного алгоритма для 300 сегментов. Верхняя строка представляет сегментацию, полученную после прямой реализации алгоритма, средняя — с предварительной фильтрацией изображения и нижняя строка — при пирамидальной реализации алгоритма с размером стракслея  $4 \times 4$ .



**Рис. 5.** Визуальный результат сегментации предложенного алгоритма для 600 сегментов. Верхняя строка представляет сегментацию, полученную после прямой реализации алгоритма, средняя — с предварительной фильтрацией изображения и нижняя строка — при пирамидальной реализации алгоритма с размером стракселя  $4 \times 4$ .



**Рис. 6.** Визуальный результат сегментации предложенного алгоритма для 1000 сегментов. Верхняя строка представляет сегментацию, полученную после прямой реализации алгоритма, средняя — с предварительной фильтрацией изображения и нижняя строка — при пирамидальной реализации алгоритма с размером стракслея  $4 \times 4$ .

## СПИСОК ЛИТЕРАТУРЫ

1. Shi J., Malik J. Normalized cuts and image segmentation // *IEEE Transactions on pattern analysis and machine intelligence*. — 2000. — Vol. 22, no. 8. — P. 888–905.
2. Ren X., Malik J. Learning a classification model for segmentation // *Computer Vision, IEEE International Conference on / IEEE Computer Society*. — Vol. 2. — 2003. — P. 10–10.
3. Achanta R., Shaji A., Smith K. et al. Slic superpixels compared to state-of-the-art superpixel methods // *IEEE transactions on pattern analysis and machine intelligence*. — 2012. — Vol. 34, no. 11. — P. 2274–2282.
4. Li Z., Chen J. Superpixel segmentation using linear spectral clustering // *Proceedings of the IEEE conference on computer vision and pattern recognition*. — 2015. — P. 1356–1363.
5. Shen J., Du Y., Wang W., Li X. Lazy random walks for superpixel segmentation // *IEEE Transactions on Image Processing*. — 2014. — Vol. 23, no. 4. — P. 1451–1462.
6. Wang P., Zeng G., Gan R. et al. Structure-sensitive superpixels via geodesic distance // *International journal of computer vision*. — 2013. — Vol. 103. — P. 1–21.
7. Veksler O., Boykov Y., Mehrani P. Superpixels and supervoxels in an energy optimization framework // *Computer Vision—ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5–11, 2010, Proceedings, Part V 11 / Springer*. — 2010. — P. 211–224.
8. Moore A. P., Prince S. J., Warrell J. “Lattice cut”—constructing superpixels using layer constraints // *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition / IEEE*. — 2010. — P. 2117–2124.
9. Wang J., Wang X. Vcells: Simple and efficient superpixels using edge-weighted centroidal voronoi tessellations // *IEEE Transactions on pattern analysis and machine intelligence*. — 2012. — Vol. 34, no. 6. — P. 1241–1247.
10. Papon J., Abramov A., Schoeler M., Worgotter F. Voxel cloud connectivity segmentation-supervoxels for point clouds // *Proceedings of the IEEE conference on computer vision and pattern recognition*. — 2013. — P. 2027–2034.
11. Giordano D., Murabito F., Palazzo S., Spampinato C. Superpixel-based video object segmentation using perceptual organization and location prior // *Proceedings of the IEEE conference on computer vision and pattern recognition*. — 2015. — P. 4814–4822.
12. Arbelaez P., Maire M., Fowlkes C., Malik J. Contour detection and hierarchical image segmentation // *IEEE transactions on pattern analysis and machine intelligence*. — 2010. — Vol. 33, no. 5. — P. 898–916.
13. Yang F., Lu H., Yang M.-H. Robust superpixel tracking // *IEEE Transactions on Image Processing*. — 2014. — Vol. 23, no. 4. — P. 1639–1651.
14. Tian Z., Zheng N., Xue J. et al. Video object segmentation with shape cue based on spatiotemporal superpixel neighbourhood // *IET Computer Vision*. — 2014. — Vol. 8, no. 1. — P. 16–25.
15. Cheng J., Liu J., Xu Y. et al. Superpixel classification based optic disc and optic cup segmentation for glaucoma screening // *IEEE transactions on medical imaging*. — 2013. — Vol. 32, no. 6. — P. 1019–1032.
16. Pham T. Q. Parallel implementation of geodesic distance transform with application in superpixel segmentation // *2013 International Conference on Digital Image Computing: Techniques and Applications (DICTA) / IEEE*. — 2013. — P. 1–8.
17. Zhou Y., Pan X., Wang W. et al. Superpixels by bilateral geodesic distance // *IEEE Transactions on Circuits and Systems for Video Technology*. — 2016. — Vol. 27, no. 11. — P. 2281–2293.
18. Ye Z., Yi R., Yu M. et al. Fast computation of content-sensitive superpixels and supervoxels using q-distances // *Proceedings of the IEEE/CVF International Conference on Computer Vision*. — 2019. — P. 3770–3779.
19. Ford Jr L. R., Fulkerson D. R. *Flows in networks*. — Princeton university press, 2015. — Vol. 54.

20. Mozerov M., van de Weijer J. Improved recursive geodesic distance computation for edge preserving filter // IEEE Transactions on Image Processing. — 2017. — Vol. 26, no. 8. — P. 3696–3706.
21. Karnaukhov V., Mozerov M. Restoration of noisy multispectral images with a geodetic distance filter // Journal of Communications Technology and Electronics. — 2018. — Vol. 63, no. 6. — P. 612–615.
22. Martin D., Fowlkes C., Tal D., Malik J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics // Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001 / IEEE. — Vol. 2. — 2001. — P. 416–423.
23. Liu Y.-J., Yu C.-C., Yu M.-J., He Y. Manifold slic: A fast method to compute content-sensitive superpixels // Proceedings of the IEEE conference on computer vision and pattern recognition. — 2016. — P. 651–659.
24. Van den Bergh M., Boix X., Roig G. et al. Seeds: Superpixels extracted via energy-driven sampling. // ECCV (7). — 2012. — Vol. 7578. — P. 13–26.
25. Tang D., Fu H., Cao X. Topology preserved regular superpixel // 2012 IEEE international conference on multimedia and expo / IEEE. — 2012. — P. 765–768.

## Accurate and fast geodesic distance calculation algorithm for superpixel segmentation

M.G.Mozerov, V.N.Karnaukhov, V.I.Kober, L.V. Zimina

Modeling in affine space based on geodesic distance makes it possible to implement important computer vision methods. One of such applications is superpixel segmentation, where it is necessary to calculate the geodesic distance from the center of the specified segments to an arbitrary point in the image. However, the algorithms proposed so far for calculating such distances for segmentation problems have been heuristic, iterative approaches that do not guarantee the expected result. In this article, we propose a new fast algorithm for calculating geodesic distance and prove that such an algorithm is accurate, the image segments obtained as a result of applying this algorithm are simply connected regions. That is, the algorithm allows you to get single-connected super pixels at the output, unlike many other methods based on the use of spatial proximity of geodetic distance and requiring additional adjustments. The proposed method surpasses analogues in the efficiency of border recognition and in the speed of calculation.

**KEYWORDS:** Super-pixel segmentation, fast algorithms, calculation of the exact geodetic distance.