

# РЕАЛИЗАЦИЯ АЛГОРИТМА СОПОСТАВЛЕНИЯ СТЕРЕОИЗОБРАЖЕНИЙ

DOI: 10.36724/2072-8735-2021-15-11-40-44

Manuscript received 12 May 2021;  
Accepted 28 June 2021

**Мезенцева Екатерина Михайловна,**  
Поволжский государственный университет телекоммуникаций  
и информатики, г. Самара, Россия, [katya-mem@psuti.ru](mailto:katya-mem@psuti.ru)

**Малахов Сергей Валерьевич,**  
Поволжский государственный университет телекоммуникаций  
и информатики, г. Самара, Россия, [malakhov-sv@psuti.ru](mailto:malakhov-sv@psuti.ru)

**Ключевые слова:** стереоизображения,  
локальные алгоритмы, глобальные алгоритмы,  
semi-global block matching, карта глубин

В последнее время с развитием технологий, таких как дополненная реальность и беспилотные транспортные средства, получающие информацию об окружающей среде при помощи видеокамер, появилась необходимость в быстрой и точной оценке глубины на изображениях. Данная проблема является актуальной, и послужила к углубленным исследованиям в данной области. В данной статье рассматриваются основополагающие принципы локальных и глобальных алгоритмов сопоставления стереоизображений. Более подробно рассмотрена работа алгоритма Semi-Global Block Matching. Суть алгоритма состоит в том, чтобы выполнить оптимизацию линии по нескольким направлениям и вычислить агрегированные затраты путем суммирования затрат на достижение пикселя с неравенством в каждом направлении. Количество направлений влияет на время выполнения алгоритма, и хотя 16 направлений обычно обеспечивают хорошее качество, меньшее число может использоваться для достижения более быстрого выполнения. Типичная реализация алгоритма в восьми направлениях может вычислять стоимость за два прохода: при прямом проходе накапливается стоимость слева, сверху-слева, сверху и справа сверху, а при обратном проходе накапливается стоимость справа, снизу, справа, снизу и снизу слева. Также затронута реализация алгоритма путём переноса вычисления карты глубин на графический процессор (GPU) для ускорения обработки. Показаны результаты построения карты глубин, а также зависимость времени работы алгоритма от размеров входных изображений. Результаты исследований показывают, что алгоритм Semi-Global Block Matching имеет хороший баланс между производительностью и точностью результирующей карты глубин и является весьма неплохим способом оценки глубины сцены на стереоизображениях.

#### Информация об авторах:

**Мезенцева Екатерина Михайловна**, к.т.н., доцент кафедры Программное обеспечение и управление в технических системах, Поволжский государственный университет телекоммуникаций и информатики, Самара, Россия

**Малахов Сергей Валерьевич**, к.т.н., доцент кафедры Программное обеспечение и управление в технических системах, Поволжский государственный университет телекоммуникаций и информатики, Самара, Россия

#### Для цитирования:

Мезенцева Е.М., Малахов С.В. Реализация алгоритма сопоставления стереоизображений // Т-Comm: Телекоммуникации и транспорт. 2021. Том 15. № 11. С. 40-44.

#### For citation:

Mezentseva E.M., Malakhov S.V. (2021) Implementation of the stereo image matching algorithm. T-Comm, vol. 15, no. 11, pp. 40-44. (in Russian)

## Введение

В последнее время с развитием технологий, таких как дополненная реальность и беспилотные транспортные средства, получающие информацию об окружающей среде при помощи видеокамер, появилась необходимость в быстрой и точной оценке глубины на изображениях. Построение карты глубин с использованием локальных методов, как правило, является быстрым, но неточным способом, а работа с использованием глобальных методов даёт точные результаты, но выполняется весьма медленно.

Метод, называемый Semi-Global Block Matching, объединяет локальные и глобальные методологии, чтобы найти баланс между скоростью построения карты глубин и её точностью. В статье производится обзор вариации исходного алгоритма, изложенного Н. Hirsh Muller [1], в котором для ускорения расчёты выполняются при помощи технологии CUDA. Результаты показывают достаточно высокую скорость обработки изображений, вкпе с небольшими ошибками, что даёт возможность использовать алгоритм в системах реального времени.

Получение информации о глубине из наборов изображений очень важно во многих отраслях, таких как дополненная реальность, робототехника, беспилотные транспортные средства (автомобили, квадрокоптеры) и т.д.

Большинство алгоритмов для вычисления карты глубин используют локальный или глобальный подход. Локальные методы, такие как Winner-Takes-All (WTA) или Scanline Optimization (SO) [5], получают результаты независимо для групп пикселей и требуют минимальных вычислений, но из-за отсутствия учёта глобальных тенденций обычно приводят к неточному результату. Подход динамического программирования [4] также эффективен в вычислительном плане, но поскольку алгоритм просматривает только одну строку на одну итерацию, он также не учитывает глобальные тенденции и приводит к тому, что на результирующей карте глубин появляются области с полосами.

С другой стороны, глобальные методы, такие как Graph Cuts [2] и Belief Propagation [6], дают весьма точные результаты и позволяют избежать ошибок, встречающихся в локальных методах. Однако они требуют гораздо более сложных вычислений, интенсивно используют память и в конечном итоге работают значительно медленнее, чем локальные алгоритмы.

Чтобы получить как приемлемую точность, так и производительность в реальном времени, рассмотрим алгоритм, которые использует, так называемое «полуглобальное сопоставление» [2]. В алгоритме, в некоторой степени, используется как глобальный, так и локальный подходы. Кроме того, перенос вычислений на графический процессор, который идеально подходит для многопоточной обработки данных SIMD (single-instruction-multiple-data), позволяет распараллелить вычисления карты глубин на основе стереоизображений и значительно сократить время их обработки.

Для решения этой задачи предположим, что входные изображения являются ректифицированной (rectified) стереопарой. По сути, это тот случай, когда две камеры ортогональны к базовой линии и «смотрят» в одном направлении. Популярные наборы данных для исследования стереозрения, такие как набор данных Middlebury, который используется в

этой статье [3], предоставляют стереопары, которые были ректифицированы.

Особенность ректифицированных изображений состоит в том, что эпиполярные линии являются горизонтальными и находятся на одной высоте на каждом изображении. Это упрощает задачу сопоставления, потому что соответствующие точки объекта будут лежать на одной и той же эпиполярной линии на обоих изображениях, и поэтому мы должны искать только в горизонтальном направлении.

Локальные методы более подвержены влиянию шума в картах диспаратности из-за того, что в их функции стоимости может быть несколько локальных минимумов. Из-за этого полуглобальный подход использует модель, которая «штрафует» изменения в значениях несоответствия в локальных окрестностях. Это приводит к тому, что результирующая карта диспаратности сглаживается за счет ослабления высокочастотного шума, что обеспечивает более четкую оценку истинной относительной глубины объектов в сцене.

## Реализация алгоритма

Реализация метода Semi-Global Block Matching сводится к поиску минимума функции, описывающей качество потенциального несоответствия. Это представлено следующим выражением (1):

$$E(D) = \sum_p (C(p, D_p) + \sum_{q \in N_p} P_1 1\{|D_p - D_q| = 1\} + \sum_{q \in N_p} P_2 1\{|D_p - D_q| > 1\}) \quad (1)$$

Здесь  $D$  – карта диспаратности,  $p$  – местоположение пикселя на карте,  $D_p$  – значение диспаратности для пикселя  $p$ .  $N_p$  – окрестность пикселей вокруг точки  $p$ .  $1\{?\}$  – это показательная функция, которая равна единице, если аргумент в фигурных скобках true, и нулю, если false, а  $P_1$  и  $P_2$  представляют собой значения штрафа, присваиваемое различным изменениям диспаратности в локальной окрестности.  $C(p, d)$  – это начальная функция стоимости, которая основана на абсолютной разнице между уровнем серого в пикселе  $p$  в эталонном изображении и пикселе  $p$ , смещенном на  $d$  вдоль эпиполярной линии во втором изображении (предполагается, что это правое изображение в стереопаре) (2):

$$C(p, d) = |I_b(p_x, p_y) - I_m(p_x - d, p_y)| \quad (2)$$

Таким образом, функция значительно штрафует (с  $P_2$ ) большие скачки в карте диспаратности и менее сильно (с  $P_1$ ) небольшие флуктуации. Обратите внимание, что  $P_1 < P_2$ . Это позволяет уменьшить высокочастотный шум в результирующей карте диспаратности.

Оптимизация этой функции с использованием глобальной минимизации в 2-х измерениях является  $NP$ -полной и требует слишком большого объёма вычисления, что недопустимо для применения во многих практических задачах. Поиск минимум в одномерном пространстве по строкам изображения, используя динамическое программирование [4], является более простым в плане вычислений, но страдает проблемой точности, как говорилось выше. Для решения этой проблемы полуглобальное сопоставление использует несколько одномерных функций поиска минимума.

Первым шагом в реализации алгоритма является вычисление начальных затрат для всех пикселей в паре изображений при различиях в диапазоне от 0 до некоторого выбран-

ного значения  $d_{max}$ . Экспериментальным путём обнаружено, что использование значения  $d_{max}$ , равное 64, более чем достаточно для всех протестированных изображений. Это вычисление может быть эффективно выполнено на GPU, которое распределяет каждому потоку вычисления для одного пикселя и одного значения диспаратности. Система принимает стереопару RGB в качестве входных данных и затем преобразует изображение в градации серого.

Функция оценки стоимости использует разницу в значениях интенсивности в градациях серого в качестве метрики для определения того, насколько хорошим или плохим является потенциальное совпадение. Как только изображения преобразуются в градации серого, они сохраняются в памяти GPU для увеличения скорости доступа к ним.

На следующем шаге выполняется, так называемая, «агрегация стоимости». Для этого вычисляем функцию поиска локального минимума по 8 направлениям (два горизонтальных, два вертикальных, по два на каждую диагональ). Пример рекурсивного выражения показан ниже для горизонтального направления, идущего слева направо по изображению:

$$E(p_x, p_y, d) = C(p, d) + \min(E(p_x - 1, p_y, d), E(p_x - 1, p_y, d - 1) + P_1, E(p_x - 1, p_y, d + 1) + P_1, \min(E(p_x - 1, p_y, i) + P_2))$$

Этот шаг можно распараллелить, используя отдельный блок вычислений для каждого направления, и в каждом блоке каждый поток обрабатывает одну итерацию для одной диспаратности. Стоит также обратить внимание, что после каждого шага вдоль направления потока должны быть синхронизированы.

После прохождения всех итераций вычислений результаты складываются в одно значение по формуле (3):

$$S(p, d) = \sum_r W_r * E_r(p, d) \quad (3)$$

Здесь значение  $r$  – это направление вычислений, а  $W_r$  представляет собой т.н. вес для этого конкретного направления. Вводя этот термин, мы учитываем тот факт, что результаты, полученные с какого-то определенного направления, могут привести к лучшим оценкам, чем результаты с других направлений. Следовательно, мы можем взвешивать каждое направление в зависимости от типа сцены.

Последний шаг заключается в том, чтобы для каждого пикселя выполнить проход по вычисленным значениям стоимости и определить, какое значение диспаратности является минимальным. Это представлено следующей функцией (4):

$$D(p) = \operatorname{argmin}_S(p, d) \quad (4)$$

Этот шаг также может быть эффективно распараллелен на графическом процессоре (GPU), если каждый поток будет обрабатывать все различия для данного пикселя. Расхождение, возвращаемые на данном этапе, представляют собой оптимальную карту диспаратности.

Как только мы получим оптимальную карту диспаратности, следующим этапом будет определение областей окклюзии на изображении, то есть таких областей, которые видны на основном изображении (предполагается, что это левое изображение стереопары), но заслонены на соответствующем ему втором изображении стереопары.

Это можно сделать, запустив немного модифицированную версию алгоритма, чтобы сгенерировать т.н. карту несоответствий. Единственная модификация требуется для функции, которой мы первоначально вычисляли стоимость, и она принимает следующий вид (5):

$$C(p, d) = |I_m(p_x, p_y) - I_b(p_x + d, p_y)| \quad (5)$$

Как только мы получим карты несоответствия как для базового изображения, так и для сопоставленного изображения, мы можем сравнить результаты, чтобы определить заблокированные области. Для каждого пикселя в базовой карте диспаратности мы берём его текущее значение диспаратности. Затем мы сравниваем это со значением в карте несоответствия в том же месте пикселя, смещенном на базовое значение диспаратности, которое мы взяли.

Если эти два значения одинаковы (с небольшим допуском), мы расцениваем их как истинные соответствия, в ином случае мы устанавливаем значение яркости для этих пикселей, равное нулю, в базовой карте диспаратности (рис. 1).

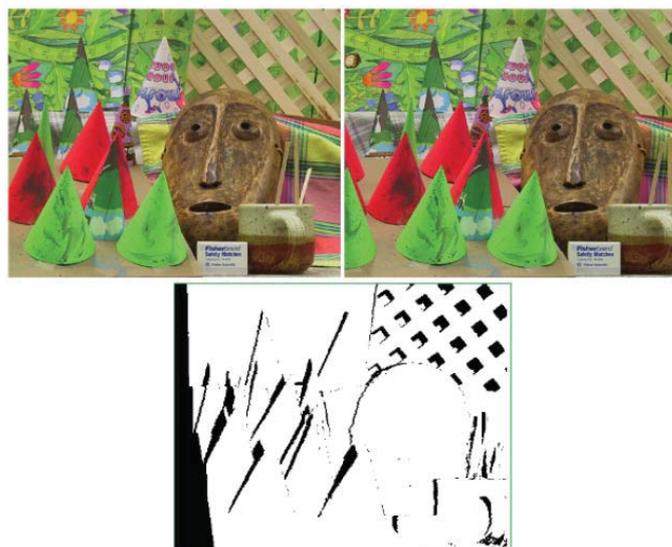


Рис. 1. Стереопара и сформированная для неё карта несоответствий

Наконец, чтобы устранить остаточный шум на результирующем изображении, мы фильтруем карту диспаратности, используя медианный фильтр. Хорошие результаты получаются при использовании небольшого ядра, например, 4x4. Это позволяет сохранить края объектов на изображении и детали на результирующей карте глубин, удаляя при этом нежелательные шумы.

Также необходимо учитывать требования к памяти для этого алгоритма. Количество используемой памяти масштабируется как  $O(mnd_{max})$ , где  $m$  – количество строк в изображении,  $n$  – количество столбцов в изображении, а  $d_{max}$  – это максимальное количество значений диспаратности.

### Результаты работы

Качество работы алгоритма было проверено с использованием набора данных Middlebury. В табл.1 показано время работы алгоритма, а также среднеквадратичная ошибка (MSE) между вычисленными значениями диспаратности и соответствующим им эталонным значениям. Алгоритм был запущен на ноутбуке с графическим процессором nVIDIA GeForce MX150 с 2 ГБ видеопамяти.

Таблица 1

MSE и время выполнения для набора данных Middleburry

Название стереопары	MSE	Время выполнения, мс
aloe	0.0286	3083
books	0.0622	2817
dolls	0.0756	2810
laundry	0.0898	2919
pots	0.1139	2994
bowling	0.0983	3064
art	0.0965	2810
wood	0.0702	2824

Как можно заметить, значения ошибок, довольно низкие и не превышают 10% ни для одного из протестированных изображений. Хотя производительность алгоритма, по-видимому, снижается в сильно нагруженных объектами сценах, таких как Art, в наборе Middleburry, или в сценах с сильными градиентными переходами, таких как пара Pots.

Взаимосвязь между временем выполнения и размером входного изображения показана на рисунке 2. Таким образом, можно отметить, что алгоритм Semi-Global Block Matching имеет время выполнения, которое линейно зависит от количества пикселей во входных изображениях.

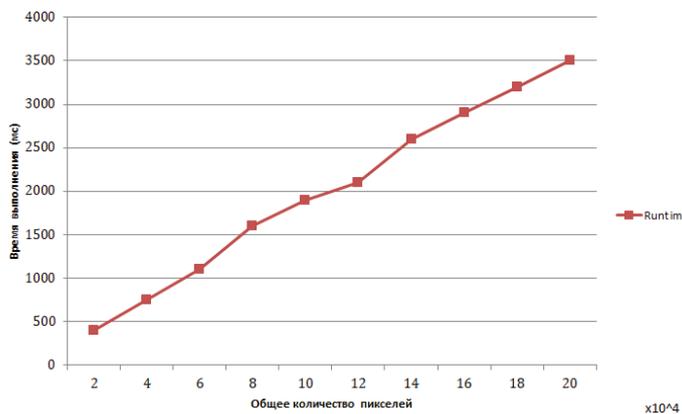


Рис. 2. Зависимость времени работы алгоритма от размера изображения

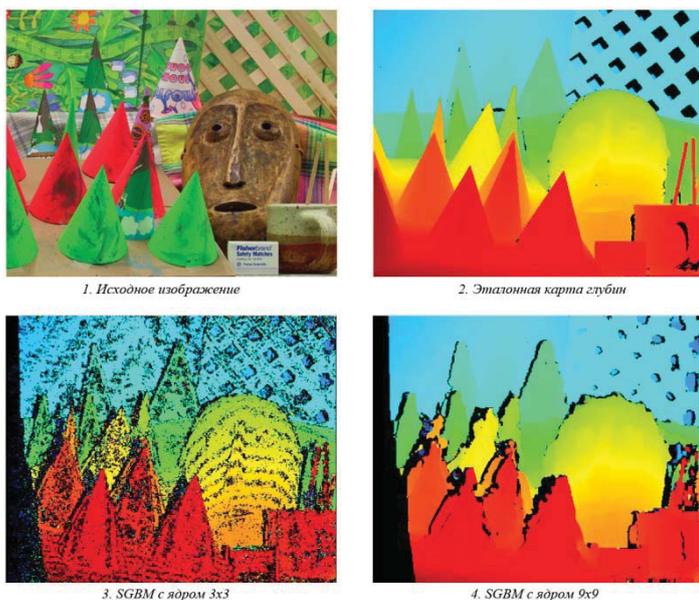


Рис. 3. Результаты работы алгоритма Semi-Global Block Matching

Для демонстрации работы алгоритма Semi-Global Block Matching на графическом процессоре, выберем произвольную пару стереоизображений. Результаты выполнения показаны на рисунке 3. Хотя некоторые области имеют ярко выраженный шум (включая области окклюзии, которые выделены чёрным цветом), изображение карты глубин показывает довольно большое количество деталей.

Как можно заметить, при использовании медианного фильтра с ядром 3x3 карта глубин получается более детализированной по границам объектов, но при этом присутствует большое количество шумов. С ядром 9x9 шумы практически отсутствуют, но границы объектов получаются нечёткими.

**Выводы**

В данной статье показана эффективность алгоритма Semi-Global Block Matching для оценки глубины на основе набора стереоизображений. Используя комбинацию глобальных и локальных методов, результаты показали низкий процент ошибок по сравнению с эталонными изображениями. Благодаря более мощному оборудованию и реализации вычислений на процессорах CUDA алгоритм показал приемлемую скорость работы для применения в режиме реального времени.

Процент ошибок может быть ещё больше уменьшен, если изменить способ обработки окклюзий. Сейчас мы можем обнаруживать окклюзии, сравнивая карту глубины как для базового, так и для соответствующего ему изображения. Как только найдены окклюзии, яркость этих пикселей устанавливается в значение, равно нулю, что может вызвать увеличение измеренного MSE. Применяя метод интерполяции как предложено в [2], мы можем повысить четкость карты глубины и повысить её точность.

Результаты исследований показывают, что алгоритм Semi-Global Block Matching имеет хороший баланс между производительностью и точностью результирующей карты глубины и является весьма неплохим способом оценки глубины сцены на стереоизображениях.

**Литература**

1. Hirschmuller H. Accurate and efficient stereo processing by Semi-Global Block Matching and mutual information // 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005, vol. 2, pp. 807-814.
2. Boykov Y., Veksler O., Zabih R. Efficient approximate energy minimization via graph cuts // IEEE Transactions on Pattern Analysis and Machine Intelligence, 23(11), 2001, pp. 1222-1239.
3. Hirschmuller H., Scharstein D. Evaluation of cost functions for stereo matching // IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), Minneapolis, MN, June 2007.
4. Van Meerbergen G., Vergauwen M., Pollefeys M., Van Gool L. A hierarchical symmetric stereo algorithm using dynamic programming. International Journal of Computer Vision, 47(1/2/3), pp. 275-285, April-June 2002.
5. Scharstein D., Szeliski R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms // International Journal of Computer Vision, vol. 47, 2002, pp. 7-42.
6. Sun J., Shum H. Y., Zheng N. N. Stereo matching using belief propagation // IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(7), pp. 787-800, July 2003.

## IMPLEMENTATION OF THE STEREO IMAGE MATCHING ALGORITHM

**Ekaterina M. Mezentseva**, Povolzhskiy State University of Telecommunications and Informatics, Samara, Russia, [katya-mem@psuti.ru](mailto:katya-mem@psuti.ru)  
**Sergey V. Malakhov**, Povolzhskiy State University of Telecommunications and Informatics, Samara, Russia, [malakhov-sv@psuti.ru](mailto:malakhov-sv@psuti.ru)

**Abstract**

Recently, with the development of technologies such as augmented reality and unmanned vehicles that receive information about the environment using video cameras, there is a need for fast and accurate depth assessment in images. This problem is relevant, and has led to in-depth research in this area. This article discusses the fundamental principles of local and global stereo image matching algorithms. The operation of the Semi-Global Block Matching algorithm is described in more detail. The essence of the algorithm is to perform line optimization in multiple directions and calculate the aggregated costs by summing the costs of reaching a pixel with an inequality in each direction. The number of directions affects the execution time of the algorithm, and while 16 directions usually provide good quality, a smaller number can be used to achieve faster execution. A typical implementation of the algorithm in 8 directions can calculate the cost in two passes: the forward pass accumulates the cost on the left, the top-left, top and top-right, and the reverse pass accumulates the cost on the right, bottom, right, bottom, and bottom left. Its implementation is also affected by transferring the calculation of the depth map to a graphics processor (GPU) to speed up processing. The results of the construction of the depth map, as well as the dependence of the time of the algorithm on the size of the input images are shown.

**Keywords:** stereo images, local algorithms, global algorithms, semi-global block matching, depth map.

**References**

1. H. Hirschmuller, "Accurate and efficient stereo processing by Semi-Global Block Matching and mutual information" 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005, pp. 807-814 vol. 2.
2. Y. Boykov, O. Veksler, and R. Zabih. Efficient approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence, 23(11):1222-1239, 2001.
3. H. Hirschmuller and D. Scharstein. Evaluation of cost functions for stereo matching. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), Minneapolis, MN, June 2007.
4. G. Van Meerbergen, M. Vergauwen, M. Pollefeys, and L. Van Gool. A hierarchical symmetric stereo algorithm using dynamic programming. International Journal of Computer Vision, 47(1/2/3):275-285, April-June 2002.
5. D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," International Journal of Computer Vision, vol. 47, pp. 7-42, 2002.
6. J. Sun, H. Y. Shum, and N. N. Zheng. Stereo matching using belief propagation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(7):787-800, July 2003.

**Information about authors:**

**Ekaterina M. Mezentseva**, Povolzhskiy State University of Telecommunications and Informatics, Assistant Professor, Software and Management in technical Systems Department, Samara, Russia

**Sergey V. Malakhov**, Povolzhskiy State University of Telecommunications and Informatics, Assistant Professor, Software and Management in technical Systems Department, Samara, Russia