

# Практичный подход ускорения синтеза изображений методом трассировки лучей Монте-Карло

А. Груздев, В.Фролов, А. Игнатенко

Московский государственный университет им. М.В. Ломоносова,  
Факультет вычислительной математики и кибернетики, Москва, Россия  
aleks-gruzdev92@yandex.ru, vfrolov@graphics.cs.msu.ru, ignatenko@graphics.cs.msu.ru

## Аннотация

Данная работа предлагает новый высококачественный метод ускорения синтеза изображений методом трассировки путей Монте-Карло. Ключевой идеей данного метода является выделение и обработка (фильтрация) падающего освещения в пространстве экрана на основе дополнительной информации о пикселях (глубина, нормаль и т.п.), что позволяет добиться высокого качества фильтрации и избежать размытия границ. Не накладывая никаких требований на алгоритм трассировки лучей, предложенный метод позволяет ускорить построение изображений методом трассировки путей Монте-Карло на 1 – 2 порядка.

**Keywords:** трассировка путей Монте-Карло, многомерная фильтрация, интенсивность освещения

## 1. ВВЕДЕНИЕ

Алгоритм трассировки путей Монте-Карло [Ritschel et al 2011] – физически корректный и высококачественный алгоритм синтеза. Но его существенным недостатком является медленная сходимости.

Возможно ускорение синтеза в результате трассировки небольшого числа лучей, а затем удаления оставшегося шума с помощью фильтрации изображения. В работе предлагается алгоритм, превосходящий аналогичные методы по качеству фильтрации и позволяющий существенно сократить время синтеза изображения.

## 2. ОБЗОР СУЩЕСТВУЮЩИХ МЕТОДОВ

### 2.1. Кэш освещенности

Кэш освещенности [Pharr and Humphreys 2004] является одним из способов ускорения вычисления интеграла освещенности на множестве точек. Метод предназначен для работы с диффузной составляющей освещения. Компонента вторичного диффузного освещения на изображении меняется плавно, поэтому ее можно вычислить каким-либо из методов лишь в очень небольшом числе точек, а в остальных – интерполировать.

В работе [Kontkanen et al 2006] рассматривается метод ускорения синтеза методом трассировки путей Монте-Карло или фотонных карт с помощью объединения алгоритма кэша освещенности [Pharr and Humphreys 2004] и фильтрации. Метод позволяет значительно сократить количество выборок, необходимых для формирования кэша, путем фильтрации на этапе выборки значений из кэша. Для фотонных карт фильтрация применяется на этапе финального сбора. Веса фильтра подбираются индивидуально для каждой точки кэша, в зависимости от сложности геометрии в этой точке. Алгоритм позволяет синтезировать изображение в 5 раз быстрее, чем с помощью кэша освещенности. Метод позволяет избежать артефактов, характерных для кэша освещенности, но обладает артефактами, вызванными размытием, и усложняет операцию выборки из кэша освещенности.

### 2.2. Адаптивная фильтрация

Алгоритм [Suykens and Willems 2000] ускоряет синтез изображений прогрессивным методом трассировки лучей Монте-Карло за счёт фильтрации результата каждой итерации. Авторы предлагают использовать фильтр с ядром Епанечникова [Silverman 1986], размер которого варьируется в зависимости от плотности распределения выборок Монте-Карло по эмпирическому закону. Учитывая лишь распределение выборок, а не особенности сцены, алгоритм плохо сохраняет границы на изображениях.

### 2.3. Многомерная фильтрация

Авторами статьи [Gastal and Oliveira 2012] предложен многомерный фильтр для изображений, применимый к обработке произвольных сигналов. Рассмотренный фильтр позволяет учесть большое количество измерений, помимо цвета, содержащих дополнительную информацию. Показывая высококачественный результат, сравнимый с билатеральной фильтрацией [Kornprobst and Tumblin 2009], метод обладает линейной сложностью и высокой скоростью работы.

### 2.4. Фильтрация с wavelet-преобразованием

В [Dammertz et al 2010] предлагается алгоритм для фильтрации освещения в пространстве экрана в реальном времени. Статья предлагает усовершенствование билатерального фильтра с wavelet-преобразованием A-Trous [Holschneider et al 1989], добавляя к преобразованию дополнительную весовую функцию для определения границ на изображении. Для вычисления функции используется дополнительная информация о пикселях: направление нормали и координаты в мировом пространстве. Дополнительный вес в wavelet-преобразовании определяется на основе близости пикселей по нормали и координате. Метод хорошо сохраняет границы на изображениях, но фильтрация страдает наличием артефактов в виде цветных пятен на однородных поверхностях. Реализация на графическом процессоре позволяет обрабатывать изображения 0.25 мегапикселя со скоростью примерно 4 кадра в секунду.

### 2.5. Адаптивное распределение выборок Монте-Карло.

В статье [Nachisuka et al 2008] рассматривается метод адаптивного распределения Монте-Карло выборок по изображению. Каждый луч рассматривается как точка многомерного пространства (например, учитываются координаты на линзе, время), тем самым задаётся многомерная функция в пространстве изображения. Многомерная функция анизотропно интерполируется, и для следующих выборок определяются позиции ближе к резким изменениям многомерной функции, которые соответствуют тем областям на изображении, где необходимо больше выборок: границы, мягкие тени, области вне фокуса камеры.

**Минимизация MSE.** Работа [Rousselle et al 2011] описывает итеративный метод расчёта Монте-Карло выборок и синтеза изображений. На каждой итерации метод выбирает наиболее

оптимальное распределение некоторого фиксированного числа Монте-Карло выборок в пространстве изображения, которое минимизирует ошибку по метрике MSE [Lehmann and Casella 1998]. Так как невозможно вычислить MSE без эталонного изображения, то используется аппроксимация [Silverman 1986]. Затем реконструируется изображение по вычисленным выборкам: производится размытие рассчитанного изображения предопределённым набором фильтров (размытие по Гауссу с разным размером ядра), а затем для каждого пикселя выбирается значение, которое минимизирует ошибку по MSE. Оценка ошибки текущей итерации учитывается в следующей итерации для выбора распределения.

**Использование нелокальной фильтрации.**  
Итеративный метод расчёта и фильтрации выборок Монте-



Рис. 1. Пример работы предложенного метода: (а) – без фильтрации, (б) – предложенный алгоритм (~1 мин), (с) – эталон (~50)

## 2.6. Guided-фильтрация

[He et al 2010] предлагает guided-фильтр, обрабатывающий изображение, руководствуясь дополнительной информацией о каждом пикселе. При удачном выборе дополнительной информации guided-фильтр превосходит по качеству билатеральную фильтрацию. В [Bauszat et al 2011] рассматривается применение guided-фильтрации в трассировке лучей Монте-Карло. В рассмотренном методе освещение разделяется на первичное и вторичное. Вторичное освещение обрабатывается с помощью guided-фильтра, где в качестве дополнительной информации используются карта нормалей, карта глубины и дополнительный буфер цвета, полученный расчётом лучей, проходящих через центр пикселя. Предложенный подход демонстрирует лучшие результаты и по качеству, и по времени по сравнению с кросс-билатеральной фильтрацией [Kornprobst and Tumblin 2009] и A-Trous фильтрацией [Dammertz et al 2010].

## 2.7. Вероятностная фильтрация

Рассматривая процесс трассировки лучей Монте-Карло как Пуассоновский процесс, авторы [Doidge and Jones 2013] предлагают эффективный способ фильтрации выборок. Для каждого пикселя изображения все выборки кластеризуются по глубине луча, и, исходя из предположения о Пуассоновском процессе, возможно предположение о том, что распределение выборок в кластере одного пикселя может сравниться с распределением в кластерах соседних пикселей, что позволяет фильтру переиспользовать освещение с соседних пикселей. Также фильтр использует информацию о нормальных и глубине для обнаружения границ. Рассмотренный алгоритм позволяет получать изображение при малом количестве выборок на пиксель, по качеству превосходящее кросс-билатеральную фильтрацию [Kornprobst and Tumblin 2009].

В статье [Sen and Darabi 2012] авторы представили метод многомерной фильтрации, учитывающий стохастическую зависимость между случайными величинами в методе

Карло [Rousselle et al 2012] существенно ускоряет процесс трассировки лучей Монте-Карло и превосходит по качеству метод [Rousselle et al 2011]. Каждая итерация делится на три шага. На первом шаге производится расчёт фиксированного числа лучей, и результат записывается в буфер. Затем, на втором шаге, выполняется нелокальная фильтрация буфера методом non-local means denoising [Buades et al 2005]. Третий шаг вычисляет попиксельную невязку и значения невязки, затем используются на первом шаге следующей итерации для распределения лучей в пространстве изображения. Для первой итерации используется равномерное распределение. Быстродействие метода ограничивается вычислительно сложным алгоритмом нелокальной фильтрации, синтез 1 мегапикселя занимает 8 – 16 секунд с ускорением на графическом процессоре.

трассировки лучей Монте-Карло и значениями в полученном семпле. Для каждого пикселя рассматривается вектор, включающий в себя набор случайных величин и дополнительную информацию о сцене (например, распределение точек на линзе). Если шум на изображении вызван случайным распределением точек прохождения луча через линзу, то между распределением точек на линзе и между шумом изображения существует некоторая функциональная зависимость. Наличие такой информации позволяет снизить вклад в ядро фильтра тех пикселей, на цвет которых повлиял шум. В алгоритме предлагается использование билатеральной фильтрации [Kornprobst and Tumblin 2009], параметры которой вычисляются на основе взаимной информации между распределением дополнительной информации в точке и случайных величин.

## 2.8. Выводы

Кэш освещенности [Pharr and Humphreys 2004], [Kontkanen et al 2006] решает задачу ускорения синтеза изображений, но обладает артефактами и требует существенного изменения алгоритма синтеза.

Работы [Hachisuka et al 2008], [Rousselle et al 2011] и [Rousselle et al 2012] описывают использование более эффективного распределения выборок в пространстве экрана, чем равномерное, однако эти подходы требуют модификации трассировщика лучей.

Методы фильтрация позволяют обойтись меньшими изменениями алгоритма синтеза. Методы [Suykens and Willems 2000], [Kornprobst and Tumblin 2009], [Dammertz et al 2010] обрабатывают изображения, не имея информации о геометрии, и, следовательно, плохо сохраняют границы изображений. Аналогично подходам [Gastal and Oliveira 2012], [Dammertz et al 2010], [He et al 2010], [Doidge and Jones 2013], [Sen and Darabi 2012] предлагается использовать дополнительную информацию о сцене для улучшения качества фильтрации на границах. Метод [Gastal and Oliveira 2012] позволяет объединить идею guided-фильтрации [He et al 2010] для



Рис. 2. Вычисление цвета финального изображения: а) интенсивность первичного освещения; б) интенсивность вторичного освещения; в) цвет текстуры; д) отражённое освещение; е) финальное изображение

сохранения границ и нелокальной фильтрации [Buades et al 2005], улучшающей обработку однородных областей.

### 3. ПРЕДЛОЖЕННЫЙ АЛГОРИТМ

В данной работе предлагается применение идей, описанных в статье [Gastal and Oliveira 2012], к задаче синтеза изображений методом трассировки путей Моне-Карло. Метод основан на предположении, что изображение сцены  $F$  может быть представлено в виде следующей формулы:

$$F = (D + I) * T + R, \quad (1)$$

где  $D$  – интенсивность первичного освещения;  $I$  – интенсивность вторичного освещения;  $T$  – цвет текстуры;  $R$  – освещение, отражённое от зеркальных поверхностей (рис. 2). Трассировщик лучей, таким образом, должен иметь возможность отдельно рассчитывать указанные изображения. Ключевой идеей такого подхода является отделение интенсивности падающего освещения от цвета пикселя. Такой подход позволяет отделить фильтрацию зашумлённой интенсивности падающего освещения от рисунка текстуры, т.е. переиспользовать значения соседних пикселей с одинаковой освещённостью, но разным цветом текстуры. В результате чего в предложенном подходе фильтр не размывает текстуру в отличие от [Gastal and Oliveira 2012].

Предложенный алгоритм можно разделить на три шага:

1. Предобработка освещения: перевод пикселей в non-local-means space [Buades et al 2005], [Gastal and Oliveira 2012].
2. Фильтрация освещения. Применяется сепарабельный фильтр, описанный в [Gastal and Oliveira 2012] использующий в качестве дополнительной информации освещение в non-local-means space, карту нормалей и карту глубины.
3. Суммирование освещения и синтез финального изображения по формуле (1).

Первые два шага применяются одинаковым образом к первичному ( $D$ ) и вторичному ( $I$ ) освещению. Затем выполняется третий шаг, использующий результаты

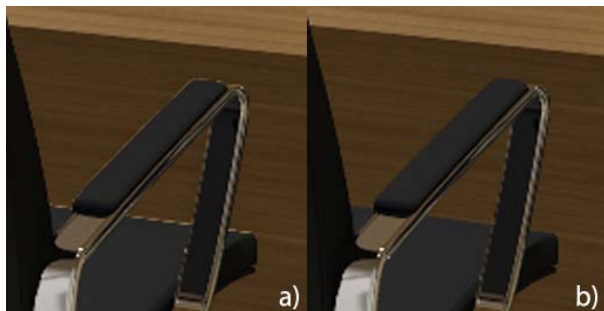


Рис. 3. Устранение артефактов на границах

предыдущих шагов. При расчёте изображения по формуле (1) могут возникать артефакты на границах (рис. 3) из-за того, что на границах объектов изображения не соблюдается введенное предположение (формула 1), поскольку цвет пикселей, принадлежащих граничным объектам, складывается из освещения различных поверхностей, и в этом случае формула 1 не выполняется.

#### 3.1. Устранение ступенчатости

Для устранения артефактов используется аналогичная MSAA техника (рис. 3). [Lelegr 1980]. Для того чтобы устранить артефакт ступенчатости, фильтр на основе дополнительной информации выделяет множество граничных пикселей  $G$ , не похожих по нормали и глубине на соседние пиксели. Для этих пикселей трассировщик лучей сохраняет дополнительно 16 суб-пикселей (4x4), содержащих нормаль и глубину. Далее если пиксел принадлежит множеству  $G$ , фильтрация для него выполняется 16 раз с различной доп. информацией суб-пикселей, а результат усредняется.

#### 3.2. Отраженное освещение

Отражённое освещение  $R$  не обрабатывается фильтром, так как для него нужен другой набор дополнительной информации. Однако возможно рекурсивное применение метода по глубине переотражения лучей. Т.е. алгоритм полностью повторяется для отражённого освещения: сохраняется интенсивность освещения, вычисляется дополнительная информация для отражённых лучей. Затем в формуле (1) более высокого уровня используется отфильтрованное освещение более низкого уровня.

### 4. РЕЗУЛЬТАТЫ

Для проведения экспериментальных исследований производился синтез изображений в разрешении 1024x768. Для синтеза входных данных фильтрации рассчитывалось 256 путей на пиксель, для эталонного изображения – 4096 путей на пиксель.

MSE	Adaptive manifolds filtering	Adobe Photoshop denoising	Предложенный алгоритм	Коэф. освещённости
«Переговорная комната»	7.73	7.15	1.50	2.05
«Cornel box»	8.12	7.05	1.64	1.58
«Спальня»	6.73	6.11	3.12	2.65

Таблица 1. Сравнение качества фильтрации с альтернативными методами



(a) Adobe Photoshop (b) Adaptive Manifolds (c) Предложенный метод (d) Кэш освещённости

Рис. 4. Сравнение предложенного метода с альтернативными методами: а) Adobe Photoshop (реализация билатеральной фильтрации); б) Adaptive Manifolds [Gastal and Oliveira 2012]; в) Предложенный метод; д) Кэш освещённости [Pharr and Humphreys 2004]

Время синтеза (сек)	Синтез входной информации	Фильтрация на CPU	Суммарное время на CPU	Фильтрация на GPU	Суммарное время на GPU	Кэш освещённости	Синтез эталона
«Cornel box»	53,40	6,42	59,82	0,16	53,56	24,02	611,13
«Переговорная комната»	56,83	6,40	63,23	0,16	59,99	98,23	629,04
«Спальня»	70,21	6,53	76,74	0,17	70,38	111,41	716,22
«Спонза»	69,04	6,59	75,63	0,16	69,20	26,83	643,65

Таблица 2. Сравнение скорости фильтрации с альтернативными методами

По результатам оценки качества фильтрации (табл. 1 и 2) видно, что предложенный метод обладает серьёзным преимуществом перед Adaptive Manifold filtering [Gastal and Oliveira 2012] и Adobe Photoshop denoising, но остаётся наравне с кэшем освещённости [Pharr and Humphreys 2004] (рис. 4). Тем не менее, предложенный метод лишен характерных артефактов в кэша освещённости в углах и является более простым в реализации.

## 5. ССЫЛКИ

[1] [Bauszat et al 2011] Bauszat, P., Eisemann, M., Magnor, M.: *Guided Image Filtering for Interactive High-quality Global Illumination*. In Computer Graphics Forum, Vol. 30, No. 4. – P. 1361 - 1368. Blackwell Publishing Ltd, June, 2011.

[2] [Buades et al 2005] Buades, A., Coll, B., Morel, J.M.: *A non-local algorithm for image denoising*. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2005.

[3] [Dammertz et al 2010] Dammertz, H., Sewtz, D., Hanika, J., Lensch H.: *Edge-Avoiding A-Trous Wavelet Transform for fast Global Illumination Filterin*. In High Performance Graphics, 2010.

[4] [Doidge and Jones 2013] Doidge, I. Jones. – M.: *Probabilistic illumination-aware filtering for Monte Carlo rendering*. In The Visual Computer, 29(6 - 8). – P. 707 - 716, 2013.

[5] [Gastal and Oliveira 2012] Gastal, E., and Oliveira. – M.: *Adaptive manifolds for real-time high-dimensional filtering*. In ACM Transactions on Graphics (TOG), 31(4), 33, 2012.

[6] [Gorban et al 2008] Gorban, A., Kégl, B., Wunsch, D., and Zinovyev, A.: *Principal manifolds for data visualization and dimension reduction*. ISBN 978-3-540-73750-6, 2008.

[7] [Hachisuka et al 2008] Hachisuka, T., Wojciech, J., Weistroffer, R., Dale, K., Humphreys, G., Zwicker, M., Jensen, H.: *Multidimensional Adaptive Sampling and Reconstruction for Ray Tracing*. In ACM Transactions on Graphics, Vol. 27, No. 3, Article 33, August, 2008.

[8] [He et al 2010] He, K., Sun, J. and Tang, X.: *Guided image filtering*. In Computer Vision–ECCV 2010. – P. 1 - 14. Springer Berlin Heidelberg, 2010.

[9] [Holschneider et al 1989] Holschneider, M., Kronald-Martinet, R., Morlet, J., Tchamitchian, P.: *A real-time algorithm for signal analysis with the help of the wavelet transform*. Springer-Verlag, 1989.

[10] [Kontkanen et al 2006] Kontkanen, J., Räsänen, J. and Keller, A.: *Irradiance filtering for monte carlo ray tracing*. In “Monte Carlo and Quasi-Monte Carlo Methods 2004” – P. 259 - 272. Springer Berlin Heidelberg, 2006.

[11] [Kornprobst and Tumblin 2009] Kornprobst, P., and Tumblin, J.: *Bilateral filtering: Theory and applications*. Now Publishers Inc, 2009.

[12] [Lehmann and Casella 1998] Lehmann, E. and Casella, G.: *Theory of point estimation*. ISBN 0-387-98502-6, 1998.

[13] [Leler 1980] Leler, W.: *Human Vision, Anti-aliasing, and the Cheap 4000 Line Display*. In ACM SIGGRAPH Computer Graphics 14 (3): 308 – 313, July, 1980.

[14] [Pharr and Humphreys 2004] Pharr, M. and Humphreys, G.: *Physically based rendering: from theory to implementation*. Morgan Kaufmann Publishers Inc., 2004.

[15] [Ritschel et al 2011] Ritschel, T., Dashesbacher, C., Grosch, T., Kautz, J.: *The state of the art in interactive global illumination*. In Computer Graphics Forum, vol 31. – P. 160 - 188, February 2012.

[16] [Rousselle et al 2012] Rousselle, F., Knaus, C. and Zwicker, M.: *Adaptive Rendering with Non-Local Means Filtering*. In ACM Transactions on Graphics, Vol. 31, No. 6, November 2012

[17] [Rousselle et al 2011] Rousselle, F., Knaus, C. and Zwicker, M.: *Adaptive sampling and reconstruction using greedy error minimization*. In Proceedings of the 2011 SIGGRAPH Asia Conference, ACM, New York, NY, USA, 159:1 – 159:12, 2011.

[18] [Sen and Darabi 2012] Sen, P., and Darabi, S.: *On filtering the noise from the random parameters in Monte-Carlo rendering*. In ACM Trans. Graph. 31, 3, June, 2012.

[19] [Silverman 1986] Silverman, B.: *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, New York, 1986.

[20] [Suykens and Willems 2000] Suykens, F. and Willems, Y.: *Adaptive filtering for progressive monte carlo image rendering*. In WSCG, February, 2000.