

# Определение оптимального фильтра для алгоритма поиска оптического потока Lucas-Kanade

А. Ю. Ермолчев, А. Н. Бугров

Государственный университет «Дубна», г. Дубна, Россия

**Аннотация.** Работа посвящена определению оптимального фильтра для алгоритма поиска оптического потока Lucas-Kanade. В данной работе приведено описание алгоритма Lucas-Kanade, выбраны несколько фильтров, реализованных в библиотеке OpenCV. В качестве примеров изображений был выбран набор данных Middlebury. В эксперименте сначала проводилось сравнение фильтров при их использовании только на входных изображениях, а затем и на всех уровнях пирамиды. Для сравнения методов фильтрации между собой были использованы такие метрики, как средняя угловая ошибка и средняя конечная ошибка. В результате был выбран один метод для фильтрации изображений при использовании алгоритма Lucas-Kanade.

**Ключевые слова:** компьютерное зрение, оптический поток, фильтрация изображений, алгоритм Lucas-Kanade.

DOI 10.14357/20718632190202

## Введение

В отличие от обработки статических изображений, из изменяющихся во времени последовательностей изображений можно извлечь гораздо больше информации, что является одной из основных функций систем компьютерного зрения. Получение информации о движении является сложной задачей для машин, однако было разработано несколько методов для её решения. По определению, оптический поток – это изображение видимого движения объектов, поверхностей или краев сцены, получаемое в результате перемещения наблюдателя (глаз или камеры) относительно сцены.

В 1981 году были предложены два дифференциальных алгоритма вычисления оптического потока, которые теперь считаются классическими: один – Хорном и Шунком [1], а

другой – Лукасом и Канаде [2]. Следуя определению Хорна, полем движения является 2D-проекция трехмерного поля физического движения точек поверхностей объектов сцены, тогда как оптический поток является видимым движением яркостной структуры изображения. С другой стороны, подход Лукаса-Канаде предполагает, что поток, по существу, постоянен в локальной окрестности рассматриваемого пикселя, и решает основные уравнения оптического потока для всех пикселей в этой окрестности по методу наименьших квадратов. С 1981 года было разработано множество различных алгоритмов поиска оптического потока, в том числе расширения и модификации подходов Хорна-Шунка и Лукаса-Канаде.

Данная статья посвящена повышению точности оценки оптического потока с помощью методов фильтрации. Поскольку фильтрация

изображения необходима во многих задачах, таких как сглаживание изображения, удаление шума или обнаружение краев, в случае оптического потока была рассмотрена методика фильтрации как необходимый этап предварительной обработки. Кроме того, в данной статье проанализированы различные методы фильтрации и выбран наиболее подходящий.

## Модель поиска оптического потока Lucas Kanade и грубо-точный подход (Coarse-to-Fine Approach)

В данной статье рассматривается алгоритм определения оптического потока *Lucas-Kanade*. В нём используется градиент интенсивности пикселей изображения:

$$I(x, y, t) = I(x + dx, y + dy, t + dt) \quad (1)$$

Уравнение ограничения оптического потока, полученное из разложения Тейлора для (1), было введено Хорном и Шунком в [1]. Имея две неизвестные переменные в одном уравнении, получаем проблему, называемую *aperture problem* (неоднозначность сдвига при ограниченном поле зрения для периодических изображений):

$$I_x v_x + I_y v_y + I_t = 0 \quad (2)$$

где  $I_x$ ,  $I_y$  и  $I_t$  обозначают производные функции изображения  $I(x, y)$  по  $x$ ,  $y$  и  $t$  (Рис. 1). Вектор  $V = (v_x, v_y)$  определяет вектор скорости в направлениях  $x$  и  $y$ .

Это уравнение не может быть решено, поскольку в нём два неизвестных, но, если предпо-

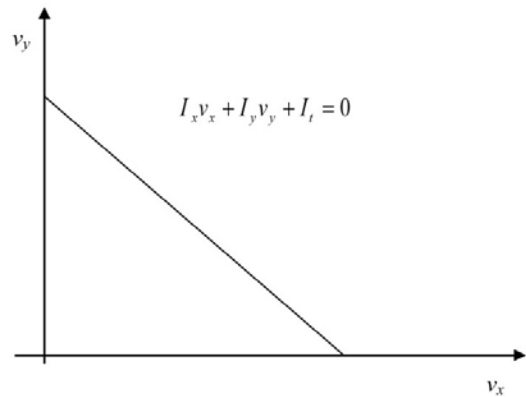


Рис. 1. Линия ограничения оптического потока

лагается, что небольшая область имеет одинаковую скорость, задача имеет решение. Таким образом,  $V$  можно найти на пересечении ограничений Хорна-Шунка для каждого пикселя. Если мы рассмотрим только два пикселя, мы получим одну точку пересечения, как на Рис. 2 (а).

Согласно *Lucas-Kanade* считается, что область из нескольких пикселей имеет одинаковую скорость. Система уравнений теперь переопределена. Поэтому предполагается, что решение с помощью метода наименьших квадратов должно дать корректную оценку значения оптического потока для пикселя, как показано на Рис. 2 (b).

Предполагается, что уравнение оптического потока используется для всех пикселей в пределах окна с центром в пикселе  $p$ . Вектор локального потока  $(v_x, v_y)$  должен удовлетворять ограничению оптического потока для области пикселей с одинаковой скоростью, выраженной как:

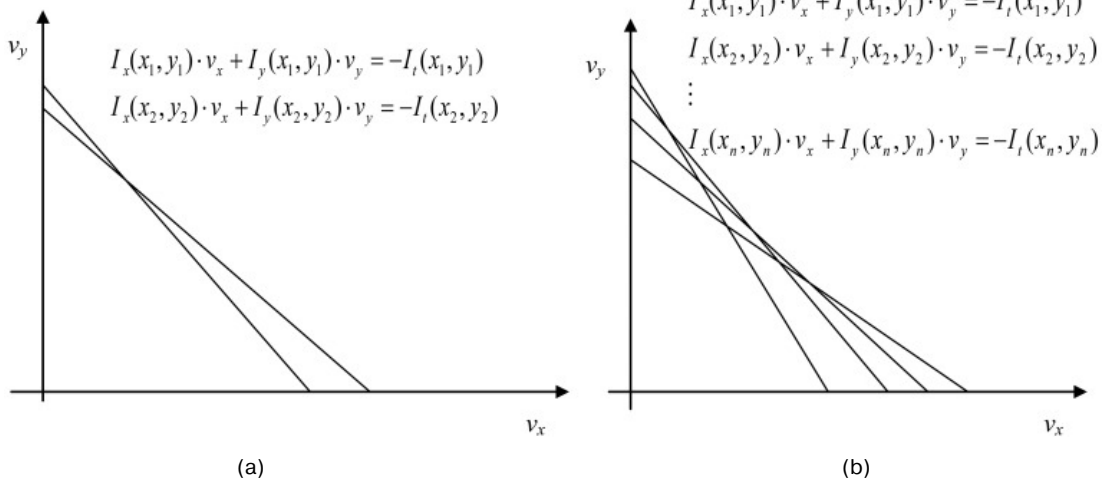


Рис. 2. Пересечение (а) двух оптических потоков и (б) нескольких линий ограничения оптического потока

$$\begin{aligned}
I_x(x_1, y_1) \cdot v_x + I_y(x_1, y_1) \cdot v_y &= -I_t(x_1, y_1) \\
I_x(x_2, y_2) \cdot v_x + I_y(x_2, y_2) \cdot v_y &= -I_t(x_2, y_2) \\
&\vdots \\
I_x(x_n, y_n) \cdot v_x + I_y(x_n, y_n) \cdot v_y &= -I_t(x_n, y_n)
\end{aligned} \quad (3)$$

Система уравнений (3) может быть переписана с использованием матрично-векторной записи:

$$\underbrace{\begin{pmatrix} I_x(x_1, y_1) & I_y(x_1, y_1) \\ I_x(x_2, y_2) & I_y(x_2, y_2) \\ \vdots & \vdots \\ I_x(x_n, y_n) & I_y(x_n, y_n) \end{pmatrix}}_A \cdot \underbrace{\begin{pmatrix} v_x \\ v_y \end{pmatrix}}_v = - \underbrace{\begin{pmatrix} I_t(x_1, y_1) \\ I_t(x_2, y_2) \\ \vdots \\ I_t(x_n, y_n) \end{pmatrix}}_b \quad (4)$$

Эта система имеет больше уравнений, чем неизвестных, следовательно, она переопределена. Метод *Lucas-Kanade* позволяет получить решение с использованием метода наименьших квадратов. В итоге, решается система  $2 \times 2$ :

$$A^T A v = A^T b$$

или

$$v = (A^T A)^{-1} A^T b \quad (5)$$

Поскольку информация о градиенте получается по соседним пикселям, движение реального объекта не может выходить за пределы рассматриваемой области. Кроме того, локальная окрестность, учитываемая для метода наименьших квадратов, является конечной, и большие смещение вычисляются достаточно плохо. Поэтому обычно используется пирамидальная реализация. Разрешение входных изображений изменяется до более низкого, сначала путем фильтрации с помощью фильтра нижних частот, а затем с дополнительной дискретизацией с коэффициентом 2. Начинается вычисление оптического потока с изображениями с самым низким разрешением, на самом высоком пирамидальном уровне. Затем результат передается на более высокий уровень разрешения в качестве начальной оценки. Запуск алгоритма на более высоких разрешениях приведет к более высокой точности для поля потока.

## Эмпирический метод выбора оптимального фильтра

В целях экспериментальной оценки был использован набор данных *Middlebury* [4]. В пользу репрезентативности этого набора гово-

рит наличие в нём как реальных, так и синтетических изображений, зашумлённых и «чистых» изображений, различных типов движения: и с малым неоднородным смещением и, напротив, со значительным и близким к линейному. Три набора данных «*Dimetrodon*», «*RubberWhale*» и «*Hydrangea*», содержат изображения реального мира, в то время как синтетическая компьютерная графика содержится в четырех наборах, названных «*Grove2*», «*Grove3*», «*Urban2*» и «*Urban3*». Последний набор под названием «*Venus*» содержит стереоизображения.

На основании [3] и [4] для измерения производительности оптического потока были использованы следующие метрики:

1. Средняя угловая ошибка (*AAE* – *average angular error*). Она представляет собой разность углов между правильным и оцененным векторами потока, определяемыми как:

$$AAE = \cos^{-1}(\hat{c} \cdot \hat{e}) \quad (6)$$

где  $\hat{c}$  – нормализованный эталонный вектор движения, а  $\hat{e}$  – нормализованный расчётный вектор.

2. Средняя конечная ошибка (*AEE* – *average endpoint error*), определяемая как:

$$AEE = \sqrt{(u - u_{GT})^2 + (v - v_{GT})^2} \quad (7)$$

где  $(u, v)$  – расчётный поток, а  $(u_{GT}, v_{GT})$  – эталонный оптический поток.

На самом первом этапе оценки был использован пирамидальный итеративный алгоритм *Lucas-Kanade* с тремя пирамидальными уровнями в сочетании с различными методами фильтрации в восьми наборах данных *Middlebury*. Для этапа предварительной обработки были использованы шесть различных методов фильтрации: фильтр Гаусса, Медианный фильтр, фильтр Лапласа, ЛОГ, Адаптивный фильтр и Билатеральный. Билатеральная фильтрация – это метод нелинейной фильтрации, впервые предложенный *Tomasi* [5]. Хотя существуют различные применения фильтров, идея состоит в том, чтобы сгладить изображения, сохранив их края.

В фильтре Гаусса использовалось стандартное отклонение  $\sigma = 1$ , медианный фильтр имел ядро  $3 \times 3$ , фильтр ЛОГ имел размер  $5 \times 5$  и стандартное отклонение  $\sigma = 0,5$ , фильтр Лапла-

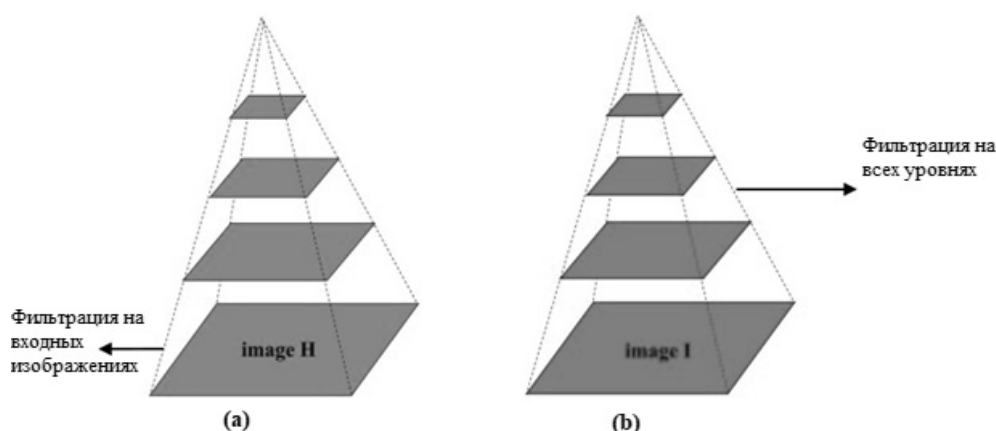


Рис. 3. Грубо-точная оценка оптического потока, (a) начальная фильтрация и (b) фильтрация на всех уровнях

са имел значение  $\alpha = 1$ , а для случая била-терального фильтра использовалось стандартное отклонение в пространственной области 0,1 и стандартное отклонение в области интенсивности 0,1, а в адаптивном фильтре была использована окрестность  $3 \times 3$  для оценки локального среднего значения изображения и стандартное отклонение. Ранее упомянутые параметры были получены после большого набора тестов, в которых параметры каждого филь-тра варьировались и выбирались на основе минимальной ошибки ( $A\bar{A}E$  и  $A\bar{E}E$ ).

## Методология эксперимента

В начале исследования цель состояла в том, чтобы найти подходящий метод фильтрации в методе оптического потока *Lucas-Kanade*. Поэтому эксперимент был разделён на три части (Рис. 3).

1. Фильтрация на входных изображениях для расчета пирамидального оптического потока.

2. Фильтрация применяется ко всем уровням пирамиды с измененными размерами для рас-чета пирамидального оптического потока.

3. Сравнение 1 и 2.

## Применение фильтрации на входных изображениях на начальном уровне расчета оптического потока

В этом разделе представлены эксперимен-тальные результаты для пирамидальной реали-зации *Lucas-Kanade*, в случае, когда фильтры были применены до расчёта оптического пото-ка. Основное внимание было сконцентрировано на средней угловой ошибке и средней конечной ошибке, оценённых для шести различных ме-тодов фильтрации. На Рис. 4 показано измене-ние  $A\bar{A}E$  (в градусах), вычисленное для наборов

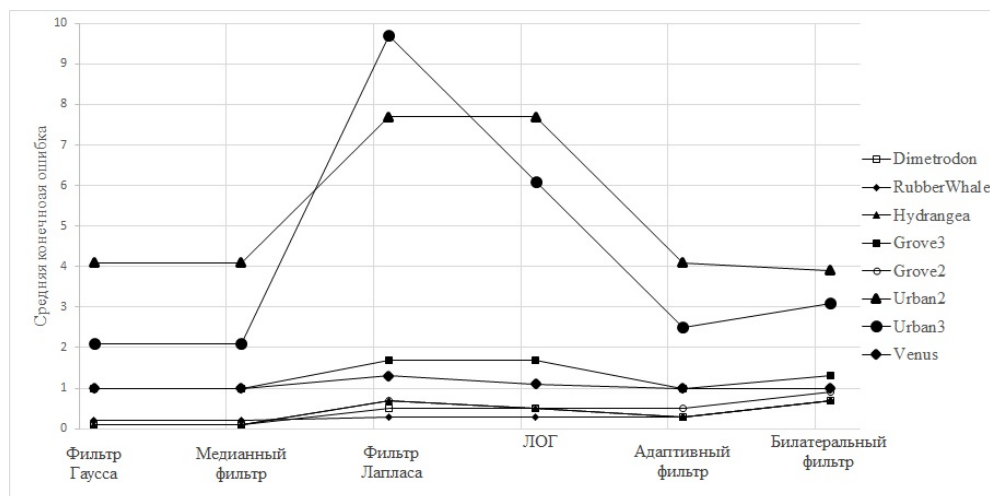


Рис. 4. Средняя конечная ошибка с использованием разных фильтров только на входных изображениях

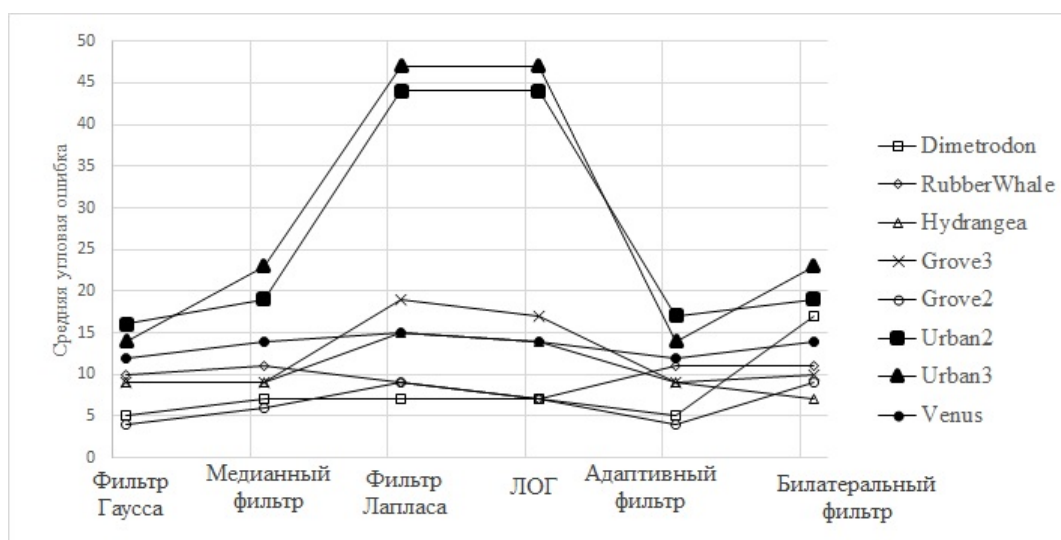


Рис. 5. Средняя угловая ошибка с использованием разных фильтров только на входных изображениях

Табл. 1. Наименьшая средняя ошибка конечной точки для наиболее эффективных фильтров

| Фильтр                                        | Dimetrodon | RubberWhale | Hydrangea | Grove3 | Grove2 | Urban2 | Urban3 | Venus |
|-----------------------------------------------|------------|-------------|-----------|--------|--------|--------|--------|-------|
| Фильтр Гаусса                                 | 4.7        | 8.71        | 8.37      | 8.52   | 3.99   | 14.7   | 11.53  | 10.70 |
| $\sigma$                                      | 0.6        | 0.3         | 0.4       | 1      | 0.7    | 4.6    | 3.8    | 0.3   |
| Медианный фильтр                              | 7.84       | 10.54       | 9.14      | 9.758  | 5.89   | 19.54  | 22.50  | 12.98 |
| Адаптивный фильтр                             | 5.82       | 10.12       | 9.17      | 9.17   | 4.39   | 17.19  | 14.5   | 11.97 |
| Билатеральный фильтр<br>$\sigma = [0.1, 0.1]$ | 17.44      | 10.59       | 8.72      | 10.38  | 8.34   | 19.24  | 22.19  | 14.36 |

данных *Middlebury*, в то время как изменение *AEE* показано на Рис. 5. Из представленных графиков, можно наблюдать следующее:

1. Результаты фильтров Гаусса, медианного, адаптивного и билатерального сравнительно лучше, чем у других.

2. Сглаживающий фильтр значительно повышает точность обнаруженного поля потока.

На основании этих результатов были выбраны пять самых эффективных фильтров, а затем их параметры изменялись, чтобы получить наименьшее *AEE*. Результаты указаны в Табл. 1.

В Табл. 1 показано, что фильтр Гаусса имеет наименьшую ошибку, но для разных стандартных отклонений, в зависимости от входных изображений.

### Применение фильтрации ко всем уровням для вычисления пирамидального оптического потока

В случае пирамидальной реализации *Lucas-Kanade*, входные изображения изменяются на каждом уровне до более низкого разрешения. Средняя угловая ошибка и средняя ошибка конечной точки представлены на Рис. 6 и Рис. 7, соответственно.

Из представленных результатов можно сделать вывод, что:

1. Результаты гауссовского, медианного, адаптивного и билатерального фильтров сравнительно лучше, чем у других.

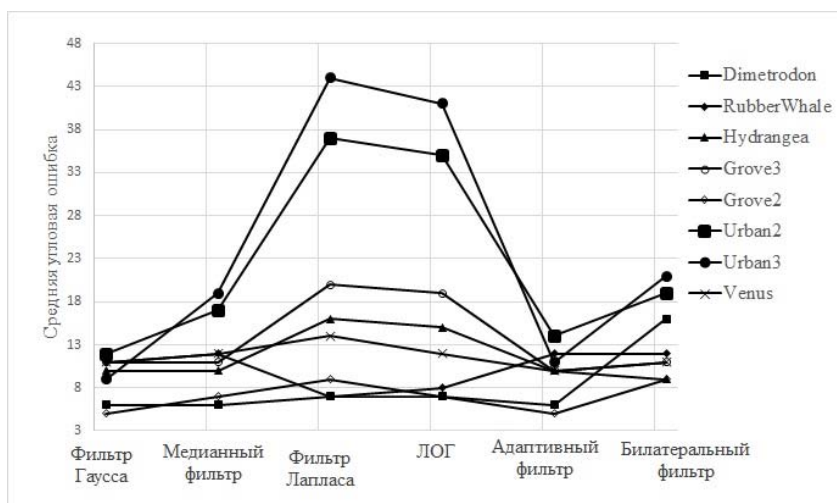


Рис. 6. Средняя угловая ошибка, полученная с использованием разных фильтров на всех уровнях

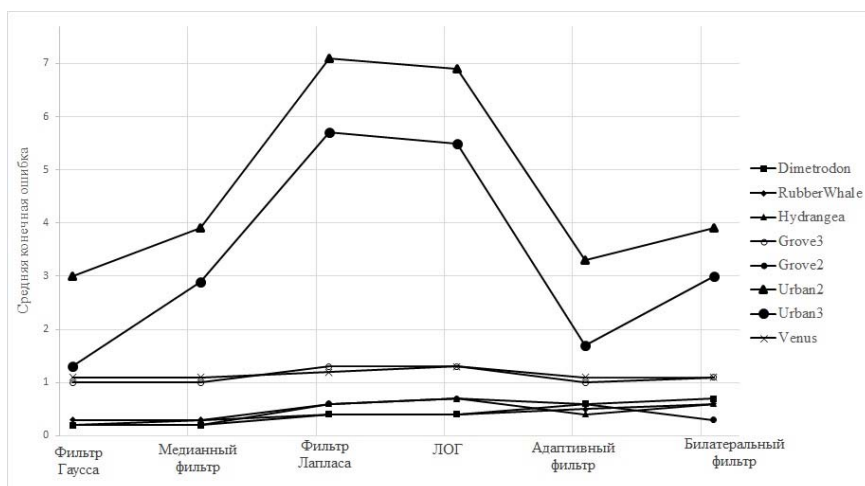


Рис. 7. Средняя конечная ошибка, полученная с использованием разных фильтров на всех уровнях

2. Фильтры сглаживания работают лучше, чем фильтры поиска краёв.

3. Для расчётов оптического потока *Lucas-Kanade* лучше использовать сглаживающие фильтры.

В качестве общего вывода из экспериментов, представленных в предыдущих подразделах, в случае пирамидального оптического потока *Lucas-Kanade*, рекомендуются сглаживающие фильтры, поскольку точность повышается. Чтобы определить, какой фильтр наиболее эффективен и когда его следует применить, было проведено сравнение, показанное в следующем подразделе.

## Сравнение методов фильтрации

Поскольку рассматривается несколько методов фильтрации, список фильтров и среднее ранжирование представлены в Табл. 2 и Табл. 3 для выбора оптимального метода фильтрации.

Изучив значения, представленные выше, можно сделать следующие выводы:

1. Фильтрация на всех пирамидальных уровнях лучше, чем фильтрация только исходных изображений.

2. Среди всех методов фильтрации фильтр Гаусса является оптимальным для вычисления оптического потока *Lucas-Kanade*, так как он имеет наименьшую ошибку.

Табл. 2. Сравнение между методами фильтрации на начальном уровне и на всех уровнях в пирамидальном оптическом потоке *Lucas-Kanade*

| Фильтр               | Dimetrodon | RubberWhale | Hydrangea | Grove3 | Grove2 | Urban2 | Urban3 | Venus |
|----------------------|------------|-------------|-----------|--------|--------|--------|--------|-------|
| Фильтр Гаусса        | x          | X           | x         | –      | –      | –      | –      | –     |
| Медианный фильтр     | x          | X           | x         | –      | –      | –      | –      | x     |
| Фильтр Лапласа       | –          | X           | –         | –      | –      | –      | –      | x     |
| ЛОГ                  | –          | X           | –         | –      | –      | –      | –      | x     |
| Адаптивный фильтр    | x          | X           | x         | –      | –      | –      | –      | x     |
| Билатеральный фильтр | –          | –           | –         | –      | –      | –      | –      | –     |

x – рекомендуется для начальной фильтрации, X – рекомендуется для фильтрации на всех уровнях

Табл. 3. Сравнение методов фильтрации на начальном уровне и всех пирамидальных уровнях с использованием среднего ранжирования

| Фильтр               | Фильтрация только на исходных изображениях |                         | Фильтрация на всех пирамидальных уровнях |                         |
|----------------------|--------------------------------------------|-------------------------|------------------------------------------|-------------------------|
|                      | Средняя угловая ошибка                     | Средняя конечная ошибка | Средняя угловая ошибка                   | Средняя конечная ошибка |
| Фильтр Гаусса        | 9.81                                       | 1.16                    | 8.95                                     | 0.95                    |
| Медианный фильтр     | 12.28                                      | 1.35                    | 11.96                                    | 1.31                    |
| Фильтр Лапласа       | 20.86                                      | 2.69                    | 18.05                                    | 2.08                    |
| ЛОГ                  | 20.6                                       | 2.27                    | 17.3                                     | 2.01                    |
| Адаптивный фильтр    | 10.29                                      | 1.14                    | 9.65                                     | 1.04                    |
| Билатеральный фильтр | 14                                         | 1.35                    | 13.91                                    | 1.35                    |

Поскольку фильтр Гаусса работает лучше, чем любой другой рассматриваемый фильтр, была определена зависимость средней угловой ошибки от параметра  $\sigma$  фильтра Гаусса. Значения, представленные на Рис. 8, относятся к случаю пирамидального оптического потока *Lucas-Kanade* с использованием фильтра Гаусса.

## Заключение

В этой статье представлено исследование по фильтрации изображений на этапе предварительной обработки для вычисления оптического потока *Lucas-Kanade*. В процессе исследования был сделан вывод о том, что в

оптимальном случае фильтрация должна выполняться на каждом пирамидальном уровне. Кроме того, было обнаружено, что фильтр Гаусса работает значительно лучше относительно других фильтров.

Данный эксперимент касался правильного использования фильтрации не только на начальном уровне, но впоследствии на каждом пирамидальном уровне. Поскольку было доступно несколько тестовых последовательностей вместе с эталонными, было получено уменьшение значения ошибки. Для проведения экспериментов были использованы язык программирования *C++* и библиотека *OpenCV*.

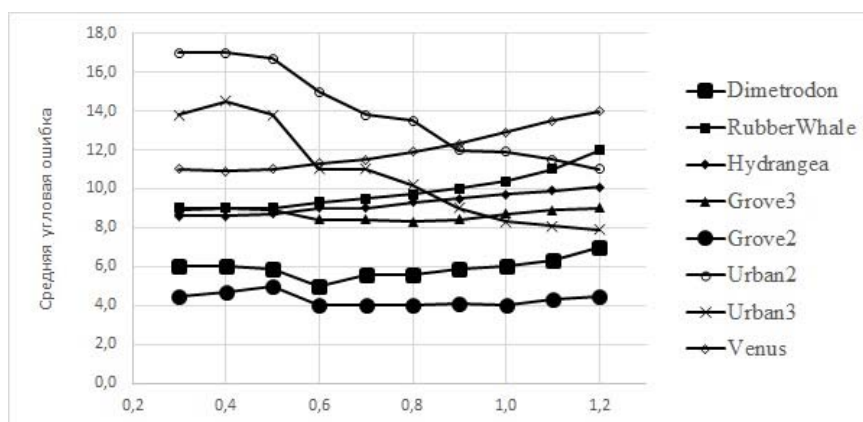


Рис. 8. Средняя угловая ошибка для разных значений  $\sigma$

## Литература

1. Horn, B.K.P. Determining optical flow [Электронный ресурс] / B.K.P. Horn, B.G. Schunk // AI Memo 572, MIT. – 1981.
2. Lucas, B.D. An iterative image registration technique with an application to stereovision [Электронный ресурс] / B.D. Lucas, T. Kanade // In Proc. of the DARPA Image Understanding Workshop. – 1981.
3. Barron, J.L. Performance of optical flow techniques [Text] / J.L. Barron, D.J. Fleet, S. Beauchemin // IJCV. – 1994. – № 12 (1).
4. Baker, S. A database and evaluation methodology for optical flow [Электронный ресурс] / S. Baker, D. Scharstein, J. Lewis [et. al.] // In Proc. ICCV. – 2007.
5. Tomasi C. Bilateral filtering for gray and color images [Электронный ресурс] / C. Tomasi, R. Manduchi. // In Proc. of the Sixth International Conference on Computer Vision. – 1998.
6. Paris S. A gentle introduction to bilateral filtering and its application [Электронный ресурс] / S. Parins, P. Kornprobst, J. Tumblin, F. Durand // In Proc. of the Special Interest Group on Computer Graphics and Interactive Techniques Conference. – 2007.
7. А.А. Шебалов. Исследование производительности методов вычисления оптического потока / А.А. Шебалов, А.Н. Баженов // Научно-технические ведомости СПбГПУ. – 2012. – № 6. – С. 152–158.
8. Прэтт У. Цифровая обработка изображений / У. Прэтт. – Пер. с англ. – М.: Мир, 1982. – 480 с.
9. И.С. Грузман. Цифровая обработка изображений в информационных системах: учеб. пособие / И.С. Грузман, В.С. Киричук, В.П. Косых, Г.И. Перетягин, А.А. Спектор. – Новосибирск: Изд-во НГТУ, 2002. – 352 с.
10. Василенко Г.И. Теория восстановления сигналов / Г.И. Василенко. – М.: Сов. радио. 1979. – 272 с.
11. Набор данных Middlebury. – Режим доступа: <http://vision.middlebury.edu/>. – (Дата обращения: 17.02.2019)
12. Документация по библиотеке OpenCV. – Режим доступа: <http://docs.opencv.org/>. – (Дата обращения: 17.02.2019)

**Ермолчев Алексей Юрьевич.** Аспирант института системного анализа и управления университета «Дубна», город Дубна. Область научных интересов: системный анализ и обработка информации. E-mail: a.ermolchev@gmail.com

**Бугров Алексей Николаевич.** Доцент, кандидат физико-математических наук, Государственный университет «Дубна», город Дубна. Область научных интересов: системный анализ и обработка информации. E-mail: bugrov2004@mail.ru

## Determining the optimal filter for the Lucas-Kanade optical flow algorithm

A. Y Ermolchev, A. N. Bugrov

Dubna state university, Dubna, Russia

**Abstract.** The work is devoted to determining of optimal filter for the Lucas-Kanade optical flow algorithm. The work shows description of Lucas-Kanade algorithm, selecting of filters from OpenCV library. Middlebury data set was selected as examples of images which were used for experiment. At the beginning filters were compared with using only on input images and then on all pyramid levels. For filters comparison average angular error and average endpoint error were used. As a result, one filtering method was selected for image filtering in Lucas-Kanade algorithm.

**Keywords:** computer vision, optical flow, image filtering, Lucas-Kanade algorithm.

**DOI** 10.14357/20718632190202



## References

1. Horn, B.K.P., Schunk, B.G. Determining optical flow. [AI Memo 572, MIT]. 1981.
2. Lucas, B.D., Kanade, T. An iterative image registration technique with an application to stereovision. [In Proc. of the DARPA Image Understanding Workshop]. 1981.
3. Barron, J.L., Fleet, D.J., Beauchemin, S. Performance of optical flow techniques. [IJCV]. 1994.
4. Baker, S., Scharstein, D., Lewis, J. A database and evaluation methodology for optical flow. [In Proc. ICCV]. 2007.
5. Tomasi, C., Manduchi, R. Bilateral filtering for gray and color images. [In Proc. of the Sixth International Conference on Computer Vision]. 1998.
6. Paris, S., Kornprobst, P., Tumblin, J., Durand, F. A gentle introduction to bilateral filtering and its application. [In Proc. of the Special Interest Group on Computer Graphics and Interactive Techniques Conference]. 2007.
7. A.A. Shebalov, A.N. Bazhenov. Performance study of optical flow calculation methods. [Scientific and technical statements SPbGPU]. 2012.
8. W. Prett. Digital image processing. Moscow. 1982. – 480 c.
9. I.S. Gruzman, V.S. Kirichuk, V.P. Kosyh, G.I. Peretyagin, A.A. Spektor. Digital image processing in information systems. Novosibirsk. 2002. – 352 c.
10. G.I. Vasilenko. Theory of signal recovery. Moscow. 1979. – 272 c.
11. Muddlebury dataset. Available at: <http://vision.middlebury.edu/> (accessed February 17, 2019).
12. OpenCV documentation. Available at: <http://docs.opencv.org/> (accessed February 17, 2019).

**Ermolchev A.**, graduate student of Dubna state university, Dubna city. Email: [a.ermolchev@gmail.com](mailto:a.ermolchev@gmail.com)

**Bugrov A.**, Associate Professor Dubna state university, Candidate of Physical and Mathematical Sciences, Dubna city.  
Email: [bugrov2004@mail.ru](mailto:bugrov2004@mail.ru)