

УДК 004.932.72

DOI: 10.25559/SITITO.15.201902.320-330

Метод вычисления плотного оптического потока на ПЛИС в реальном времени

А. В. Братулин, М. Б. Никифоров*, П. В. Беляков, Е. Ю. Холопов

Рязанский государственный радиотехнический университет имени В.Ф. Уткина, г. Рязань, Россия

390005, Россия, г. Рязань, ул. Гагарина, д. 59/1

* nikiforov.m.b@evm.rsreu.ru

Аннотация

Одной из наиболее актуальных проблем, связанных с разработкой систем технического зрения, является проблема обнаружения и выделения объектов, находящихся в поле зрения датчика изображений. Область использования технологий автоматического обнаружения объектов и выделения объектов включает в себя задачи поиска и спасения, космический мониторинг Земли, анализ движения транспортных средств, контроль движения воздушного транспорта. Объекты могут быть неподвижными и движущимися. Движение на видеоизображении формализуется таким понятием, как оптический поток. Оптический поток является базовым инструментом для обработки и анализа видеопоследовательностей, а его вычисление – наиболее распространенный подход к обнаружению движущихся объектов. В статье рассмотрены особенности аппаратной реализации на ПЛИС метода вычисления плотного оптического потока на видеоизображении высокой четкости. Разработанный метод построен на базе корреляционно-экстремального алгоритма поиска минимальной суммы абсолютных разностей яркостей пикселей окон в окрестностях текущей точки соседних кадров видеоизображения. Приведено подробное описание основных этапов данного алгоритма. Рассмотрен принцип распараллеливания и конвейеризации, позволяющий минимизировать объем вычислений, благодаря повторному использованию промежуточных результатов, полученных на предыдущих шагах. Описанный подход позволяет вычислять оптический поток для каждого пикселя в темпе поступления входного видеоизображения. Проанализированы аппаратные затраты при реализации метода на ПЛИС Xilinx Zynq Ultra Scale+, входящей в состав отладочной платы Xilinx Zynq Ultra Scale+ MPSoC ZCU102 Evaluation Kit.

Ключевые слова: оптический поток, ПЛИС, критериальная функция, обнаружение движения, абсолютная разность.

Для цитирования: Братулин А. В., Никифоров М. Б., Беляков П. В., Холопов Е. Ю. Метод вычисления плотного оптического потока на ПЛИС в реальном времени // Современные информационные технологии и ИТ-образование. 2019. Т. 15, № 2. С. 320-330. DOI: 10.25559/SITITO.15.201902.320-330

© Братулин А. В., Никифоров М. Б., Беляков П. В., Холопов Е. Ю., 2019



Контент доступен под лицензией Creative Commons Attribution 4.0 License.
The content is available under Creative Commons Attribution 4.0 License.



Method for Computing Dense Optical Flow on FPGA in Real Time

A. V. Bratulin, M. B. Nikiforov*, P. V. Belyakov, E. Y. Kholopov

Ryazan State Radio Engineering University, Ryazan, Russia

59/1 Gagarina St., Ryazan 390005, Russia

* nikiforov.m.b@evm.rsreu.ru

Abstract

One of the most actual problems in the technical vision systems is the problem of objects detection and selection in the field of the image sensor view. Applications of objects automatic detection and object allocation technologies include search and rescue tasks, space monitoring of the Earth, traffic analysis, air traffic control, etc. Objects can be stationary and moving. The movement in the video can be formalized in a term such as optical flow. Optical flow is the basic tool for processing and analyzing video sequences, and its calculation is the most common approach to the detection of moving objects. The article describes the features of the hardware implementation of the FPGA method for calculating the dense optical flow on high-definition video. The developed method is based on the correlation-extreme algorithm for finding the minimum sum of absolute differences in the brightness of the pixels of the windows in the vicinity of the current point of neighboring frames of the video. A detailed description of the main stages of this algorithm is given. The principle of parallelization and pipelization, which allows us to minimize the amount of calculations due to the reuse of intermediate results obtained in the previous steps, is considered. The described approach allows us to calculate the optical flow for each pixel at the rate of input video. The hardware implementation cost of the method on the Xilinx Zynq Ultra Scale + FPGA, which is part of the Xilinx Zynq Ultra Scale + MPSoC ZCU102 Evaluation Kit, is considered.

Keywords: optical flow, FPGA, criterion function, motion detection, absolute difference.

For citation: Bratulin A.V., Nikiforov M.B., Belyakov P.V., Kholopov E.Y. Method for Computing Dense Optical Flow on FPGA in Real Time. *Sovremennye informacionnye tehnologii i IT-obrazovanie* = Modern Information Technologies and IT-Education. 2019; 15(2):320-330. DOI: 10.25559/SITITO.15.201902.320-330



Введение

В настоящее время системы технического зрения (СТЗ) являются одним из главных средств развития автоматических систем управления движением в условиях, когда объём априорной информации не достаточен и для решения задач управления необходим анализ внешней обстановки в режиме реального времени. СТЗ находят свое применение в современных космических, авиационных, наземных, надводных и подводных мобильных объектах. Благодаря дальнодействию и достаточно высоким уровням пространственного и цветового разрешения современных линейных и матричных приемников оптического излучения СТЗ могут служить незаменимыми источниками информации при автоматическом решении задач распознавания, навигации или наведения [1]. Характерными чертами СТЗ является необходимость работы в реальном времени, наличие автономного режима функционирования [2].

В качестве элементной базы для реализации СТЗ выступают цифровые сигнальные процессоры (DSP), графические процессоры (GPU), микроконтроллеры (MCU), заказные интегральные схемы (ASIC) и программируемые логические интегральные схемы (FPGA (ПЛИС)) [3].

В настоящее время ПЛИС является одними из самых перспективных элементов цифровой схемотехники и представляет собой кристалл, на котором расположено большое количество простых логических элементов. В отличие от других видов элементной базы, в ПЛИС можно в кратчайшие сроки организовать сложные параллельные алгоритмы цифровой обработки на аппаратном уровне. При этом быстродействие цифровой обработки резко возрастает¹. Достоинства технологии проектирования устройств на основе ПЛИС следующие:

- небольшой период от начала процесса проектирования устройств до выпуска серийной продукции;
- возможность организации мелкосерийного производства без больших начальных вложений в технологию;
- относительная простота исправления ошибок проектировщиков в процессе отладки изделий;
- упрощение решения задачи согласования интерфейсов микросхем за счет возможности перенастройки блоков ввода–вывода ПЛИС на множество различных сигнальных стандартов.
- упрощение отладки и тестирования аппаратуры, как за счет обязательной реализации стандарта периферийного сканирования JTAG, так и возможности встраивания в проекты (на стадии их отладки) схем логических анализаторов;
- высокая эффективность при реализации принципа массового параллелизма, например, в устройствах цифровой обработки сигналов².

Возможность применения ПЛИС для гибкой и быстрой разработки под специализированное применение является неоспоримым преимуществом по сравнению с применением процессоров общего назначения и графических процессоров в

задачах обработки видеоизображений в реальном времени [4]. Применение языка аппаратного дизайна Verilog позволяет совмещать эффективную реализацию алгоритмов и скорость разработки [5].

При решении задач технического зрения (например: задача обнаружения движущихся или появляющихся на изображении объектов [6]) часто возникает проблема оценки движения объекта на кадрах или смещения двух последовательных кадров в видеопотоке. Существует несколько подходов к оценке смещения изображения. Основные из них это: выделение ключевых особенностей на изображении, с последующим сопоставлением ключевых точек, определение фазовой корреляции частотного представления сигнала изображения и оптический поток. Последний в основном его виде не используется для поиска смещений изображений, он используется для определения наличия движения на изображении. Так оптический поток применяют в системах детектирования объектов и слежения за ними [7].

На изображении, сформированном камерой, каждому пикселю соответствует значение яркости, полученное путем проекции объекта в 3D пространстве на плоскость изображения. При перемещении объекта или камеры соответствующая проекция также меняет положение в плоскости изображения.

Оптический поток — это векторное поле, которое показывает направление и амплитуду изменений яркости от одной картинки к другой. Методы вычисления оптического потока позволяют оценить поле смещений между двумя изображениями [8].

На данный момент вычисление оптического потока является одной из ключевых проблем в техническом зрении и используется для обнаружения движущихся объектов [9], сегментации [10], компрессии видео³, антиалайзинговая фильтрация [11], 3D-реконструкции [12], комплексирования кадров мультиизображений, получаемых бортовыми изобразительными системами [13].

При расчете оптического потока для каждой точки предыдущего изображения осуществляется поиск соответствующей точки в текущем изображении, полученном от одной и той же камеры. Корреляционные алгоритмы сравнивают небольшие окна в окрестностях текущей точки двух изображений для поиска соответствующих точек. Они формируют плотную карту векторов и поэтому относятся к категории плотных методов. Метод, который вычисляет оптический поток для всех пикселей в изображении, называется плотным, тогда как методы, обрабатывающие некоторые отдельные пиксели, называются разреженными [14]. Основным недостатком данных методов является то, что они плохо работают в областях заслонения, вызванных движением камеры. Алгоритмы, основанные на поиске ключевых характеристик изображения: ребра, линии, углы (например: применение алгоритма поиска ключевых точек SIFT при вычислении оптического потока [15]) дают надежную, но разреженную картину, которую необходимо интерполировать. В настоящий момент активно разрабатываются новые методы вычисления оптического потока. При

¹ Исследование цифровых устройств на основе программируемых логических интегральных схем (ПЛИС) в среде Quartus II [Электронный ресурс] // Лаборатория Электронных Средств Обучения (ЛЭСО) СибГУТИ. 2009. URL: <http://www.labfor.ru/guidance/digital-leso2/1> (дата обращения: 06.06.2019).

² Программируемые логические интегральные схемы (ПЛИС) [Электронный ресурс]. URL: <https://pandia.ru/text/82/522/74829.php> (дата обращения: 06.06.2019).

³ ISO/IEC JTC/SC29. Coded Representation of Pictures, Audio and Multimedia/Hypermedia Information. Committee Draft of Standard ISO/IEC 11172, Dec 6, 1991.



этом успехи в вычислении оптического потока часто определяются новыми методами оптимизации. Например, эффективный метод прямо-двойственной (Primal-Dual) минимизации функционала [16] послужил развитием нескольких моделей оптического потока [17]. Развитие методов дискретной оптимизации, основанной на графовых разрезах [18] или передачи сообщений [19], также побудило развитие различных работ по оптическому потоку [20].

В аппаратных средствах широко применяются именно корреляционные алгоритмы, поскольку эти алгоритмы регулярны и просты для реализации.

В соответствии с [21] наиболее известные алгоритмы, основанные на сопоставлении пикселей:

1. Алгоритм поиска минимальной суммы квадратов разностей (СКР). В англоязычной литературе называется Sum of Squared Differences (SSD), подробно описан в [22];

2. Алгоритм поиска минимальной суммы абсолютных разностей (САР). В англоязычной литературе упоминается как Sum of Absolute Difference (SAD), подробно описан в [23].

В данном методе используется алгоритм САР, по причине простоты его реализации.

Алгоритм поиска минимальной суммы абсолютных разностей

Суть данного алгоритма заключается в следующем. С целью вычисления плотного оптического потока для каждой точки кадра $N-1$ видеоизображения осуществляется поиск соответствующей точки в кадре N видеоизображения, которая является точкой лучшего совмещения фрагмента кадра $N-1$ видеоизображения – эталона размером $w \times w$, центрированного относительно текущей точки (x, y) , и некоторой окрестности текущей точки кадра N видеоизображения – зоны поиска. С математической точки зрения, определяются значения смещений по оси Ox - ξ и Oy - η , при которых достигается минимум критериальной функции $SAD(x, y, \xi, \eta)$ (1):

$$SAD(x, y, \xi, \eta) = \sum_{i=-\frac{(w-1)}{2}}^{\frac{(w-1)}{2}} \sum_{j=-\frac{(w-1)}{2}}^{\frac{(w-1)}{2}} AD(x, y, \xi, \eta, i, j) \quad (1)$$

$$AD(x, y, \xi, \eta, i, j) = |I_{N-1}(x+i, y+j) - I_N(x+i+\xi, y+j+\eta)|, \quad (2)$$

где $AD(x, y, \xi, \eta, i, j)$ – абсолютная разность (Absolute Differences) яркостей точек $I_{N-1}(x+i, y+j)$ и

$I_N(x+i+\xi, y+j+\eta)$;

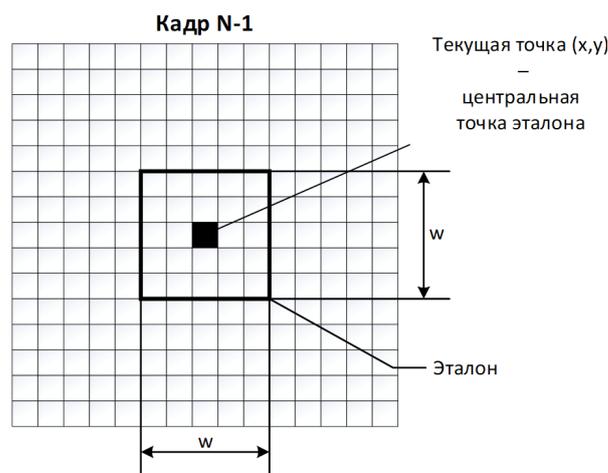
$I_{N-1}(x+i, y+j)$ – яркость точки $(x+i, y+j)$ кадра $N-1$ видеоизображения;

$I_N(x+i+\xi, y+j+\eta)$ – яркость точки $(x+i+\xi, y+j+\eta)$ кадра N видеоизображения;

w – размер эталона, центрированного относительно текущей точки с координатами (x, y) .

Диапазон изменения ξ по оси Ox и η , по оси Oy определяют зону поиска.

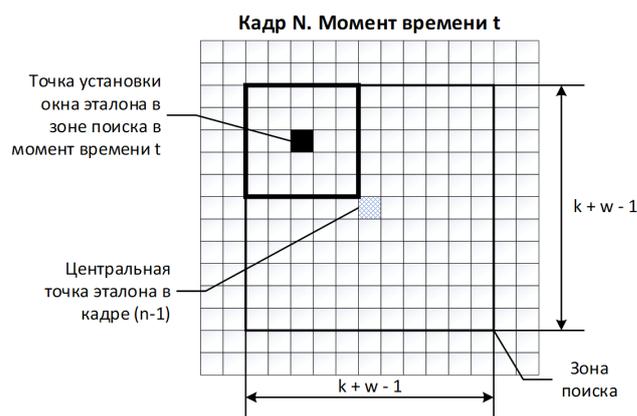
На рис. 1 изображено окно эталона, центрированное относительно точки с координатами (x, y) . Это окно сравнивается со всеми окнами зоны поиска.



Р и с. 1. Окно эталона, центрированное относительно точки с координатами (x, y)

Fig. 1. The pattern window centered on a point (x, y)

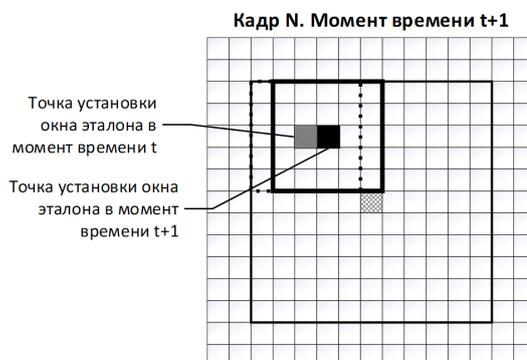
На рис. 2 показана установка окна эталона в зоне поиска в момент времени t .



Р и с. 2. Установка окна эталона в зоне поиска в момент времени t

Fig. 2. The setting of the pattern window in the search zone at the time t

После вычисления критериальной функции окно эталона смещается в зоне поиска вправо на одну точку (рис. 3).

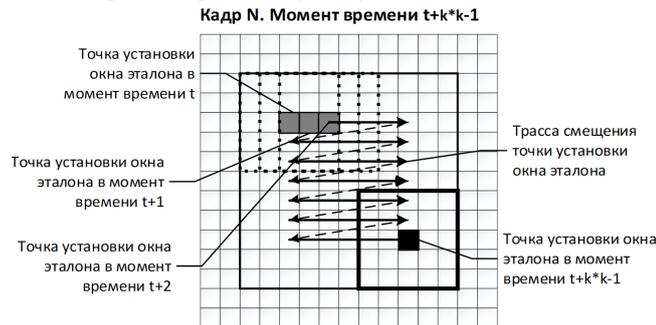


Р и с. 3. Смещение окна эталона в зоне поиска вправо на одну точку

Fig. 3. The shifting of the pattern window in the search zone to the right by one pixel



Далее с каждым шагом окно эталона движется по зоне поиска слева направо, сверху вниз (Рис. 4)



Р и с. 4. Принцип движения окна эталона по зоне поиска слева направо сверху вниз

Fig. 4. The principle of the pattern window movement in the search zone from left to right, top to bottom

Если размер зоны поиска $(k+w-1) \times (k+w-1)$, то количество точек установки окна эталона $k \times k$. При этом каждая точка эталона сравнивается с $k \times k$ точками зоны поиска $(\xi, \eta \in [-\frac{k-1}{2}; \frac{k-1}{2}])$. Таким образом вычисляются $k \times k$ значений критериальной функции $SAD(x, y, \xi, \eta)$. Далее выбирается окно, для которого $SAD(x, y, \xi, \eta)$ принимает минимальное значение и центральная точка этого окна с координатами $(x+\xi, y+\eta)$ в кадре N видеоизображения считается соответствующей точке (x, y) в кадре $N-1$ видеоизображения. В результате получаем вектор оптического потока с началом в точке (x, y) и концом в $(x+\xi, y+\eta)$. После этого осуществляется сдвиг текущей центральной точки эталона в кадре $N-1$ и зоны поиска в кадре $N, y)$ на одну координату вправо по горизонтали и выполняется процесс поиска минимального значения критериальной функции $SAD(x, y, \xi, \eta)$ для точки $(x+1, y)$.

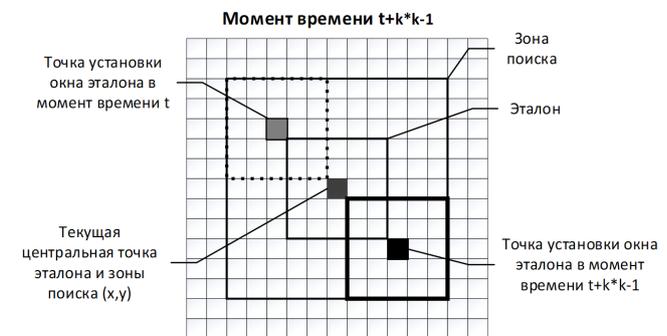
Алгоритм поиска минимальной суммы абсолютных разностей на основе повторного использования промежуточных результатов

Предположим, что размер видеоизображения $X \times Y$ при частоте кадров F кадр/сек. Для того чтобы найти векторы для всех пикселей кадров видеоизображения необходимо вычислить $X \times Y \times F$ векторов за 1 секунду. Это означает, что время вычисления одного вектора не более 9,6 наносекунд для видеоизображения высокой четкости (разрешение 1920×1080) с прогрессивной разверткой и частотой кадров 50 кадр/сек. А чтобы вычислить один вектор необходимо найти $k \times k$ значений критериальной функции $SAD(x, y, \xi, \eta)$ и $w \times w$ значений абсолютных разностей $AD(x, y, \xi, \eta, i, j)$ для вычисления каждого значения критериальной функции $SAD(x, y, \xi, \eta)$. Таким образом чтобы выполнить указанное выше требование, необходимо минимизировать объем вычислений, например, благодаря повторному использованию промежуточных результатов, полученных в процессе поиска минимума критериальной функции $SAD(x, y, \xi, \eta)$ при поиске минимума $SAD(x+1, y, \xi, \eta)$, $SAD(x+2, y, \xi, \eta)$ т.д.

Далее рассмотрим методику вычисления плотного оптического потока в реальном времени, адаптированную под логические элементы ПЛИС. Для достижения максимальной производительности необходимо, чтобы все операции были конвейеризованы и распараллелены. Следовательно, будем считать, что все операции, описанные далее выполняются параллельно и в конвейере.

На рис. 5, 6, 7 изображено повторное использование промежуточных результатов в процессе поиска минимума $SAD(x, y, \xi, \eta)$.

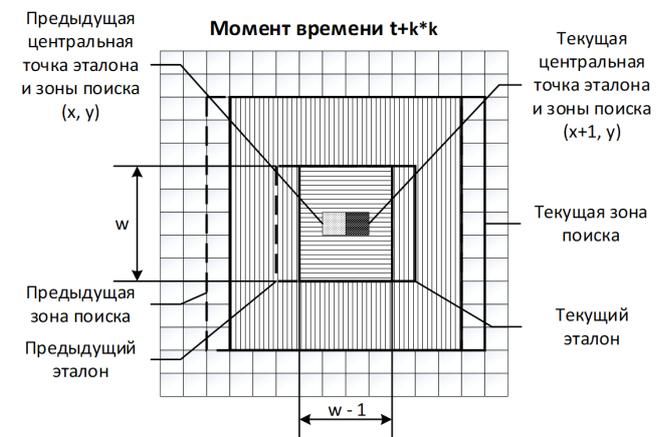
На рис. 5 изображен момент времени $t+k*k-1$, когда уже вычислено $k \times k$ значений $SAD(x, y, \xi, \eta)$ (при этом рассчитано $k \times k \times w \times w$ значений $AD(x, y, \xi, \eta, i, j)$) и выбрано минимальное значение $SAD(x, y, \xi, \eta)$ для получения вектора оптического потока в текущей центральной точке эталона (x, y) (для упрощения на рисунке показано вычисление только $SAD(x, y, 0, 0)$ и $SAD(x, y, k-1, k-1)$). В процессе этих вычислений операции над одним и тем же набором данных не выполняются.



Р и с. 5. Установка окна эталона в зоне поиска в момент времени $t+k*k-1$

Fig. 5. The setting of the pattern window in the search zone at the time $t+k*k-1$

Далее эталон и зона поиска сдвигаются на одну точку вправо с целью вычисления вектора оптического потока для точки $(x+1, y)$ (рис. 6).



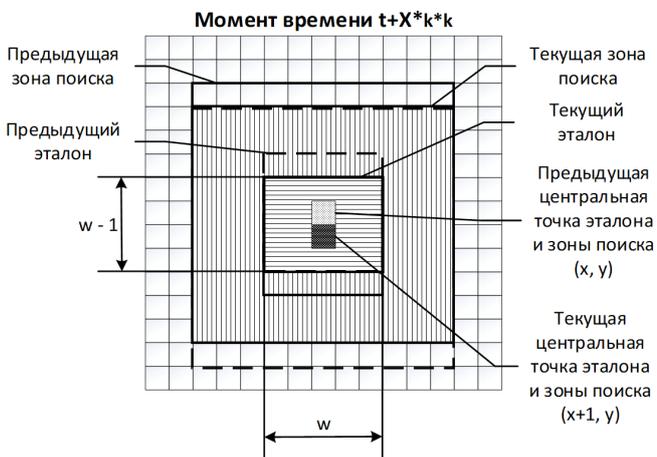
Р и с. 6. Смещение эталона и зоны поиска вправо на одну точку

Fig. 6. The shifting of the pattern and the search zone to the right by one pixel



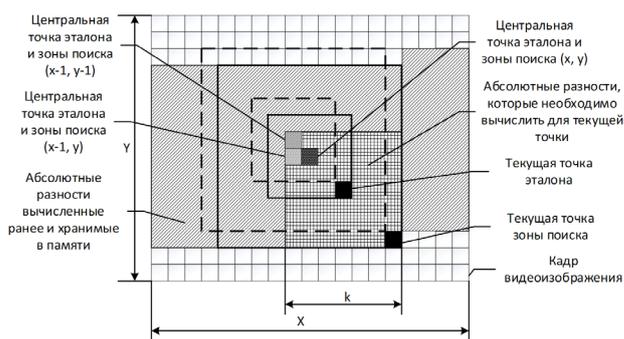
В этот момент времени точки эталона, заштрихованные горизонтальной линией уже сравнивались с точками зоны поиска, заштрихованными вертикальной линией, в процессе вычисления предыдущего вектора оптического потока. Следовательно, путем хранения $k \times k \times (w-1) \times w$ значений абсолютных разностей $AD(x, y, \xi, \eta, i, j)$ вычисленных ранее, количество новых $AD(x, y, \xi, \eta, i, j)$, которые необходимо вычислить для получения нового вектора оптического потока, может быть уменьшено до $k \times k \times w$ значений.

Когда эталон и зона поиска сдвигаются вниз на одну точку, как показано на рис. 7, точки эталона, заштрихованные горизонтальной линией, уже сравнивались с точками зоны поиска, заштрихованными вертикальной линией, в процессе вычисления предыдущего вектора оптического потока. В этом случае так же можно уменьшить количество операций вычисления абсолютных разностей $AD(x, y, \xi, \eta, i, j)$, путем хранения и повторного использования $k \times k \times w \times (w-1)$ значений абсолютных разностей $AD(x, y, \xi, \eta, i, j)$ вычисленных ранее.



Р и с. 7. Сдвигание эталона и зоны поиска вниз на одну точку
F i g. 7. The shifting of the pattern and the search zone to the bottom by one pixel

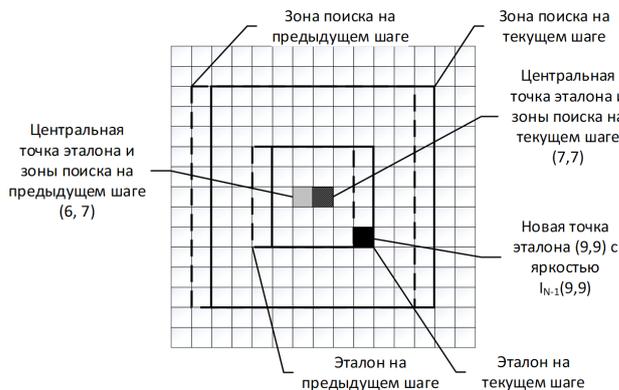
Теперь предположим, что выполняется обработка видеозображения размером $X \times Y$ (рис. 8) слева направо, сверху вниз т.е. зона поиска и эталон сначала сдвигаются вправо (вдоль оси X), а при достижении конца строки кадра видеозображения - сдвигаются влево до начала строки и вниз на следующую строку (на одну точку). В этом случае при сдвиге вправо на одну точку, уже вычислено $k \times k \times w \times (w-1)$ значений абсолютных разностей $AD(x, y, \xi, \eta, i, j)$ для $w \times (w-1)$ точек в сдвинутом эталоне в процессе получения предыдущего вектора оптического потока. Поэтому для получения каждого нового вектора оптического потока необходимо хранить $k \times k \times (w-1) \times X$ абсолютных разностей $AD(x, y, \xi, \eta, i, j)$ по строке (заштрихованная область рис. 8) и вычислить только $k \times k$ новых абсолютных разностей $AD(x, y, \xi, \eta, i, j)$ (между точкой эталона, помеченным черной точкой и $k \times k$ точками зоны поиска), которых не хватает для получения $k \times k$ значений критериальной функции $SAD(x, y, \xi, \eta)$.



Р и с. 8. Порядок вычисления абсолютных разностей при последовательной обработке кадра видеозображения
F i g. 8. The procedure for calculating absolute differences in sequential processing of a video frame

Логическое устройство и функционирование операционных узлов аппаратного вычислителя

Предположим, что видеозображение имеет размер 15×15 точек. На предыдущем шаге вычислен вектор оптического потока для точки (6, 7), эталон размером 5×5 ($w = 5$) и зона поиска размером 11×11 ($w = 7$) сдвинулись вправо на одну точку с целью вычисления вектора оптического потока для точки 7×7 т.е. на текущем шаге ($x = 7, y = 7$) (рис. 9).



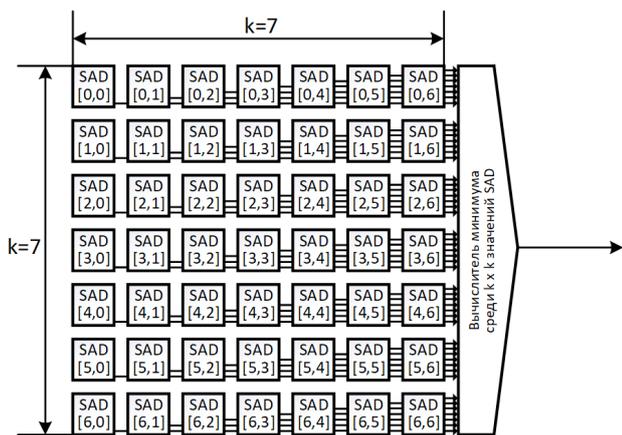
Р и с. 9. Расположение эталона 5×5 и зона поиска 11×11 на текущем шаге
F i g. 9. The pattern location 5×5 and the search zone 11×11 in the current step

Далее эталон сравнивается с 7×7 (в общем случае $k \times k$) областями зоны поиска т.е. вычисляются 7×7 (в общем случае $k \times k$) значений критериальной функции $SAD(x, y, \xi, \eta)$ и определяется минимальное.

С целью достижения максимальной производительности и вычисления плотного оптического потока в каждой точке строится конвейер из параллельно функционирующих 7×7 (в общем случае $k \times k$) блоков $SAD[m, n]$, где $m = 0(1)6, n = 0(1)6$ (в общем случае $m = 0(1)k - 1, n = 0(1)k - 1$ (рис. 7), каждый из которых соответствует определенной точке установки окна эталона в зоне поиска и вычисляет соответствующее значение



критериальной функции $SAD(x, y, \xi, \eta)$ т.е. блок $SAD[m, n]$ вычисляет значение критериальной функции $SAD\left(x, y, m - \frac{k-1}{2}, n - \frac{k-1}{2}\right)$ (например: блок $SAD[2, 5]$ вычисляет значение критериальной функции $SAD\left(x, y, 2 - \frac{7-1}{2}, 5 - \frac{7-1}{2}\right) = SAD(x, y, -1, 2)$). На выходе данного конвейера выполняется вычисление минимума среди полученных значений критериальной функции $SAD(x, y, \xi, \eta)$.



Р и с. 10. Конвейер вычисления минимального значения критериальной функции $SAD(x, y, \xi, \eta)$

Fig. 10. The pipeline for calculating the minimum of the criterion function $SAD(x, y, \xi, \eta)$

На рис. 11 изображено устройство блока $SAD[1, 0]$, а также содержимое сдвиговых регистров и памяти $M[1, 0]$, входящих в его состав, при обработке очередной новой точки эталона в точке (9, 9), соответствующему блоку $SAD[1, 0]$ т.е. $AD_{1,0}(7, 6)$ можно вычислить по формуле (3):

$AD_{1,0}(7, 6)$ - это абсолютная разность яркости новой точки эталона (9, 9) в кадре $N - 1$ и яркости соответствующей ей точки зоны поиска (7, 6) в кадре N при установке окна эталона в точке (5, 4), соответствующему блоку $SAD[1, 0]$ т.е. $AD_{1,0}(7, 6)$ можно вычислить по формуле (3):

$$AD_{m,n}\left(x + m - \frac{k-1}{2} + \frac{w-1}{2}, y + n - \frac{k-1}{2} + \frac{w-1}{2}\right) = AD\left(x, y, m - \frac{k-1}{2}, n - \frac{k-1}{2}, \frac{w-1}{2}, \frac{w-1}{2}\right) = (3)$$

$$\left| I_{N-1}\left(x + \frac{w-1}{2}, y + \frac{w-1}{2}\right) - I_N\left(x + m - \frac{k-1}{2} + \frac{w-1}{2}, y + n - \frac{k-1}{2} + \frac{w-1}{2}\right) \right|,$$

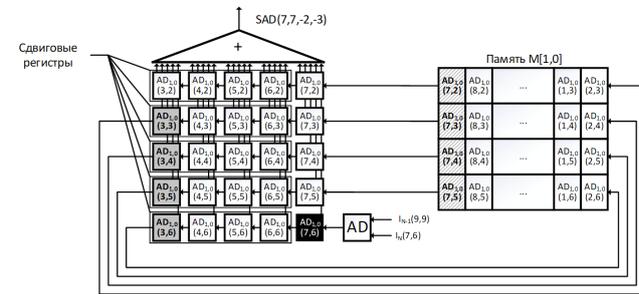
где m - координата блока $SAD[m, n]$ по оси Ox ,

n - координата блока $SAD[m, n]$ по оси Oy .

Таким образом для данного примера получим (4):

$$AD_{1,0}\left(7 + 1 - \frac{7-1}{2} + \frac{5-1}{2}, 7 + 0 - \frac{7-1}{2} + \frac{5-1}{2}\right) =$$

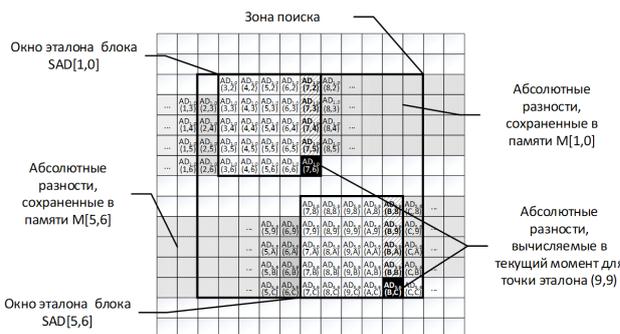
$$AD_{1,0}(7, 6) = AD(7, 7, -2, -3, 2, 2) = |I_{N-1}(7 + 2, 7 + 2) - I_N(7 + 2 - 2, 7 + 2 - 3)| = |I_{N-1}(9, 9) - I_N(7, 6)|. \quad (4)$$



Р и с. 11. Устройство блока $SAD[1, 0]$

Fig. 11. The design of the block $SAD[1, 0]$

На рис. 12 в качестве примера изображено содержимое памяти абсолютных разностей $M[1, 0]$ и $M[6, 5]$ блоков $SAD[1, 0]$ и $SAD[6, 5]$ соответственно.



Р и с. 12. Содержимое памяти абсолютных разностей $M[1, 0]$ и $M[6, 5]$

Fig. 12. Contents of absolute differences memories $M[1, 0]$ and $M[6, 5]$

Алгоритм вычисления нового значения критериальной функции $SAD\left(x, y, m - \frac{k-1}{2}, n - \frac{k-1}{2}\right)$ каждым блоком $SAD[m, n]$ следующая (на примере блока $SAD[1, 0]$):

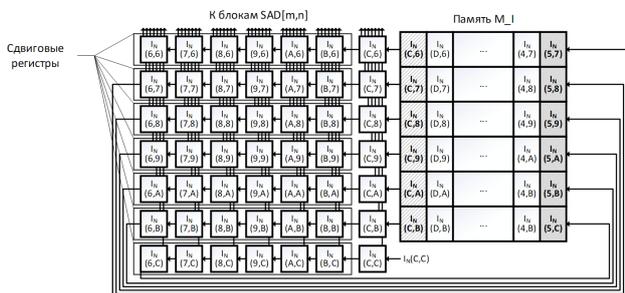
1. Из памяти $M[m, n]$ считываются $w - 1$ значений абсолютных разностей, вычисленные на предыдущих шагах. В данном случае из $M[m, n]$ считываются 4 значения абсолютных разностей $A_{1,0}(7, j)$, $j = 2(1)5$ (Квадраты, заштрихованные кривой чертой на рис. 11, 12).

2. Вычисляется новая абсолютная разность $A_{m,n}\left(x + m - \frac{k-1}{2} + \frac{w-1}{2}, y + n - \frac{k-1}{2} + \frac{w-1}{2}\right)$ по формуле (3). Для данного примера получим (4).

Следует отметить, что $I_{N-1}(9, 9)$ подается на вход всех блоков $SAD[m, n]$. Причем для каждого блока $SAD[m, n]$ точка, соответствующая новой точке эталона (9, 9) своя т.е. для $SAD[1, 0]$ - точка (7, 6), для $SAD[5, 6]$ - точка (C, C). Таким образом необходим буфер на 6 строк длиной 9 элементов и 7 сдвиговых регистров на 6 элементов (рис. 13) (в общем случае $k - 1$ строк



длиной $X - k + 1$ элементов и k сдвиговых регистров на $k - 1$ элемент).



Р и с. 13. Линия задержки с параллельным доступом к $k \times k$ элементам
F i g 13. The delay line with parallel access to $k \times k$ elements

3. Абсолютные разности, считанные из памяти $M[m, n]$ задерживаются на $w - 1$ такт на w сдвиговых регистрах для обеспечения одновременного доступа к $w \times w$ абсолютным разностям. При расчете нового значения критериальной функции значения абсолютных разностей продвигаются дальше по сдвиговому регистру

В данном примере $AD_{1,0}(7, j), j = 2(1)6$ задерживаются на 4 такта в блоке $SAD[1, 0]$ на сдвиговом регистре для обеспечения одновременного доступа к 5×5 абсолютным разностям.

4. Эти $w \times w$ абсолютных разностей с выходов сдвиговых регистров поступают на сумматор для вычисления нового значения критериальной функции $SAD\left(x, y, m - \frac{k-1}{2}, n - \frac{k-1}{2}\right)$.

В данном примере 5×5 абсолютных разностей $AD_{1,0}(i, j), i = 2(1)6, j = 3(1)7$ с выходов сдвиговых регистров поступают на сумматор для вычисления нового значения критериальной функции $SAD(7, 7, -2, -3)$.

5. Значения абсолютных разностей, вытаскиваемые из $w - 1$ сдвиговых регистров, записываются обратно в память $M[m, n]$. В данном примере $AD_{1,0}(3, j), j = 3(1)6$ - квадраты, заштрихованные плотной косой чертой на рис. 8, 9, вытаскиваются из 4-х сдвиговых регистров и записываются обратно в память $M[1, 0]$.

Таким образом, каждая абсолютная разность $AD_{m,n}\left(x + m - \frac{k-1}{2} + \frac{w-1}{2}, y + n - \frac{k-1}{2} + \frac{w-1}{2}\right)$ суммируется w раз, пока она в сдвиговом регистре и сохраняется и считывается из памяти $w - 1$ раз т.е. используется в процессе вычислений $w \times w$ значений критериальной функции $SAD(x, y, \xi, \eta)$.

Путем повторения описанной выше процедуры для всех конвейеризированных и параллельно работающих $k \times k$ блоков $SAD[m, n]$ будем получать новый вектор оптического потока в каждом такте системной частоты.

Необходимо отметить, что память $M[0, 1]$, а следовательно, и другие памяти $M[i, j], i = 0(1)k - 1, j = 0(1)k - 1$ блоков SAD должны хранить точки $w - 1$ строк длиной $X - w + 1$. Таким образом общее количество параллельно работающих банков памяти должно быть $k \times k$ размером $(w - 1) \times (X - w + 1)$.

Полученные результаты

Данный метод был реализован в виде отдельного модуля на ПЛИС Xilinx Zynq Ultra Scale+, входящей в состав отладочной платы Xilinx Zynq UltraScale+ MP SoC ZCU102 Evaluation Kit, с параметрами, приведенными в таблице 1.

Т а б л и ц а 1. Параметры разработанного модуля
T a b l e 1. Parameters of the developed module

№	Наименование параметра	Значение параметра
1	Разрешение видео	1920x1080
2	Частота кадров видео	60 Гц
3	Частота пикселей видео	148,500 МГц
4	Системная тактовая частота	150 МГц
5	Размер эталона	7x7
6	Размер зоны поиска	15x15

Выбор размеров эталона и зоны поиска, приведенных в таблице 1, осуществлялся эмпирическим путем. В результате моделирования установлено, что таких размеров достаточно, например, для реализации функции стабилизации входного видеоизображения т.е. устранения смещений, вызванных трясотой источника видеосигнала, выделения движущихся объектов.

Объем занимаемых модулем ресурсов в ПЛИС Xilinx Zynq UltraScale+⁴ приведен в таблице 2.

Т а б л и ц а 2. Объем занимаемых модулем ресурсов в ПЛИС Xilinx Zynq UltraScale+

T a b l e 2. The amount of resources occupied by the module in FPGA Xilinx Zynq UltraScale+

Наименование ресурса	Объем доступного ресурса	Объем занимаемого ресурса	Доля занимаемого ресурса
System Logic Cells (K)	599,550	34,313	5,7 %
Memory (Mb)	32,100	7,588	23,6 %
DSP Slices	2,520	0	0

Заключение

В данной статье предложен метод вычисления плотного оптического потока (в каждой точке) в темпе поступления видео на ПЛИС. Высокое быстродействие, простота реализации и сравнительно низкая ресурсоемкость алгоритма позволяют использовать его в различных задачах машинного зрения и реализовывать совместно с другими функциональными модулями системы такими как модулем улучшения видения, формирования знакографической информации, аффинных преобразований. Компромисс между затратами ресурсов ПЛИС и производительностью позволяет гибко и быстро модифицировать предложенную схему под конкретное применение, что позволяет отказаться от использования дорогостоящих БИС.

⁴ Xilinx Zynq UltraScale+ MPSoC ZCU102 Evaluation Kit [Электронный ресурс] // Xilinx - Adaptable. Intelligent, 2019. URL: <https://www.xilinx.com/products/boards-and-kits/ek-u1-zcu102-g.html> (дата обращения: 06.06.2019).



Список использованных источников

- [1] Техническое зрение в системах управления мобильными объектами-2010 // Труды научно-технической конференции-семинара. Вып. 4 / Под ред. Р. Р. Назирова. М.: КДУ, 2011. 328 с. URL: <http://www.iki.rssi.ru/books/2011tz.pdf> (дата обращения: 06.06.2019).
- [2] Алпатов Б. А., Бабаян П. В. Обнаружение и слежение за движущимися объектами в присутствии дрожания и линейных сдвигов изображения // Цифровая обработка сигналов и ее применения: тез. Докл. 6-й междунар. конф. Т. 2. 2004. С. 95-96. URL: http://www.autex.spb.su/download/dsp/dspa/dspa2004/t1_34.pdf (дата обращения: 06.06.2019).
- [3] Гуров В. С. и др. Обработка изображений в авиационных системах технического зрения / Под ред. Л. Н. Костяшкина, М. Б. Никифорова. М.: Физматлит, 2016. 240 с.
- [4] Визильтер Ю. В., Желтов С. Ю., Бондаренко А. В., Ососков М. Б., Моржин А. В. Обработка и анализ изображений в задачах машинного зрения. М.: Физматкнига, 2010. 672 с.
- [5] Беляков П. В., Ларкин Е. В., Никифоров М. Б. Модификация вариационного метода вычисления оптического потока для реализации на ПЛИС // Известия Тульского государственного университета. Технические науки. 2018. № 9. С. 19-28. URL: <https://elibrary.ru/item.asp?id=36394730> (дата обращения: 06.06.2019).
- [6] Козлов В. А., Потапов А. С. Анализ методов выделения движущихся объектов на видеопоследовательностях с шумами // Научно-технический вестник информационных технологий, механики и оптики. 2011. № 3(73). С. 39-43. URL: <https://elibrary.ru/item.asp?id=16223309> (дата обращения: 06.06.2019).
- [7] Агафонов В. Ю. Применение методов оптического потока для оценки совмещения изображений. // Образование и наука в России и за рубежом. 2017. Т. 35, № 6. URL: <https://www.gyrnal.ru/statyi/ru/314/> (дата обращения: 06.06.2019).
- [8] Belyakov P. V., Nikiforov M. B. An FPGA-optimized architecture of variational optical Flow // 2018 IEEE East-West Design & Test Symposium (EWDTS). Kazan, 2018. Pp. 1-7. DOI: 10.1109/EWDTS.2018.8524759
- [9] Холопов Е. Ю., Братулин А. В., Баранчиков А. И. Аппаратный видеоинформационный модуль обнаружения движущихся объектов в реальном времени // Известия Тульского государственного университета. Технические науки. 2018. № 9. С. 564-570. URL: <https://elibrary.ru/item.asp?id=36394789> (дата обращения: 06.06.2019).
- [10] Ваганов С. Е. Алгоритм динамической сегментации пары последовательных кадров // Компьютерная оптика. 2019. Т. 43, № 1. С. 83-89. DOI: 10.18287/2412-6179-2019-43-1-83-89
- [11] Shinya M. Spatial anti-aliasing for animation sequences with spatio-temporal filtering // Proceedings of the 20th annual conference on Computer graphics and interactive techniques (SIGGRAPH '93). ACM, New York, NY, USA, 1993. Pp. 289-296. DOI: 10.1145/166117.166154
- [12] Елесина С. И., Никифоров М. Б., Логинов А. А., Костяшкин Л. Н. Совмещение изображений в корреляционно-экстремальных навигационных системах / Под ред. Л. Н. Костяшкина, М. Б. Никифорова. М.: Радиотехника, 2015. 204 с.
- [13] Кузнецов П. К., Мартемьянов Б. В., Семавин В. И. Техническое зрение подвижных объектов. Метод анализа поля скоростей динамического изображения // Вестник компьютерных и информационных технологий. 2014. № 1(115). С. 3-9. DOI: 10.14489/vkit.2014.01.pp.003-009
- [14] Беляков П. В., Никифоров М. Б. Вариационный метод вычисления оптического потока в системе-на-кристалле // Цифровая обработка сигналов. 2018. № 3. С. 76-82. URL: <https://elibrary.ru/item.asp?id=36276676> (дата обращения: 06.06.2019).
- [15] Brox T., Malik J. Large Displacement Optical Flow: Descriptor Matching in Variational Motion Estimation // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2011. Vol. 33, No. 3. Pp. 500-513. DOI: 10.1109/TPAMI.2010.143
- [16] Кузнецов П. К., Мартемьянов Б. В., Семавин В. И., Чеботило Е. Ю. Метод определения вектора скорости движения подстилающей поверхности // Вестник Самарского государственного технического университета. Серия: Технические науки. 2008. № 2(22). С. 96-110. URL: <https://elibrary.ru/item.asp?id=13089112> (дата обращения: 06.06.2019).
- [17] Chao H., Gu Y., Napolitano M. A Survey of Optical Flow Techniques for Robotics Navigation Applications // Journal of Intelligent & Robotic Systems. 2014. Vol. 73, Issue 1-4. Pp. 361-372. DOI: 10.1007/s10846-013-9923-6
- [18] Cooke T. Two Applications of Graph-Cuts to Image Processing // 2008 Digital Image Computing: Techniques and Applications. Canberra, ACT, 2008. Pp. 498-504. DOI: 10.1109/DICTA.2008.32
- [19] Tao M., Bai J., Kohli P., Paris S. SimpleFlow: A Non-iterative, Sublinear Optical Flow Algorithm // Computer Graphics Forum. 2012. Vol. 31, Issue 2pt1. Pp. 345-353. DOI: 10.1111/j.1467-8659.2012.03013.x
- [20] Xu L., Jia J., Matsushita Y. Motion Detail Preserving Optical Flow Estimation // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2012. Vol. 34, No. 9. Pp. 1744-1757. DOI: 10.1109/TPAMI.2011.236
- [21] Алпатов Б. А. и др. Методы автоматического обнаружения и сопровождения объектов. Обработка изображений и управление. М.: Радиотехника, 2008. 176 с.
- [22] Hisham M. B., Yaakob S. N., Raof R. A. A., Nazren A. B. A., Embedded N. M. W. Template Matching using Sum of Squared Difference and Normalized Cross Correlation // 2015 IEEE Student Conference on Research and Development (SCOREd). Kuala Lumpur, 2015. Pp. 100-104. DOI: 10.1109/SCORED.2015.7449303
- [23] Vassiliadis S., Hakkennes E. A., Wong J. S. S. M., Pechanek G. G. The sum-absolute-difference motion estimation accelerator // Proceedings. 24th EUROMICRO Conference (Cat. No.98EX204). Vasteras, Sweden, 1998. Vol. 2. Pp. 559-566. DOI: 10.1109/EURMIC.1998.708071

Поступила 06.06.2019; принята к публикации 03.07.2019;
опубликована онлайн 25.07.2019.



Об авторах:

Братулин Алесандр Вадимович, аспирант кафедры электронных вычислительных машин, Рязанский государственный радиотехнический университет имени В.Ф. Уткина (390005, Россия, г. Рязань, ул. Гагарина, д. 59/1), ORCID: <http://orcid.org/0000-0003-3745-296X>, bratulin@mail.ru

Никифоров Михаил Борисович, директор НОЦ «СпецЭВМ», доцент кафедры электронных вычислительных машин, Рязанский государственный радиотехнический университет имени В.Ф. Уткина (390005, Россия, г. Рязань, ул. Гагарина, д. 59/1), кандидат технических наук, доцент, член-корреспондент Академии информатизации образования, ORCID: <http://orcid.org/0000-0002-4796-0776>, nikiforov.m.b@evm.rsreu.ru

Беляков Павел Викторович, инженер, кафедра электронных вычислительных машин, Рязанский государственный радиотехнический университет имени В.Ф. Уткина (390005, Россия, г. Рязань, ул. Гагарина, д. 59/1), ORCID: <http://orcid.org/0000-0002-0757-1418>, pvbel@rambler.ru

Холопов Евгений Юрьевич, аспирант кафедры электронных вычислительных машин, Рязанский государственный радиотехнический университет имени В.Ф. Уткина (390005, Россия, г. Рязань, ул. Гагарина, д. 59/1), ORCID: <http://orcid.org/0000-0002-0541-2878>, holopovevgu13@rambler.ru

Все авторы прочитали и одобрили окончательный вариант рукописи.

References

- [1] *Computer Vision in Control Systems of the Mobile Objects. Proceedings of the Conference-Workshop-2010*. Nazirov R.R. (ed). Issue. 4. Moscow: KDU, 2011. Available at: <https://http://www.iki.rssi.ru/books/2011tz.pdf> (accessed 06.06.2019). (In Russ., abstract in Eng.)
- [2] Alpatov B.A., Babayan P.V. Detection and tracking of moving objects in the presence of jitter and linear shifts of the image. *Digital signal processing and its applications. Thesis of the best works of the 6th international conference*. 2004; 2:95-96. Available at: http://www.autex.spb.su/download/dsp/dspa/dspa2004/t1_34.pdf (accessed 06.06.2019). (In Russ.)
- [3] Gurov V.S. et al. [The image processing in aircraft vision systems]. Kostyashkin L.N., Nikiforov M.B. (eds). Fizmatlit, Moscow, 2016. (In Russ.)
- [4] Vizilter Yu.V., Zheltov S.Yu., Bondarenko A.V., Ososkov M.B., Morzhin A.V. [Image Processing and Analysis Tasks in Machine Vision]. Fizmatkniga Publisher, Moscow, 2010. (In Russ.)
- [5] Belyakov P.V., Larkin E.V., Nikiforov M.B. Variotonal Optical Flow Method Modification FORFPGA Implementation. *News of the Tula state university. Technical sciences*. 2018; 9:19-28. Available at: <https://elibrary.ru/item.asp?id=36394730> (accessed 06.06.2019). (In Russ., abstract in Eng.)
- [6] Kozlov V.A., Potapov A.S. Analysis of extraction methods for moving objects on video sequences with noise. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*. 2011; 3(73):39-43. Available at: <https://elibrary.ru/item.asp?id=16223309> (accessed 06.06.2019). (In Russ., abstract in Eng.)
- [7] Agafonov V.Y. Application of Optical Flow Methods to Image Shift Estimation. *Education and science in Russia and abroad*. 2017; 35(6). Available at: <https://www.gyrnal.ru/statyi/ru/314/> (accessed 06.06.2019). (In Russ., abstract in Eng.)
- [8] Belyakov P. V., Nikiforov M. B. An FPGA-optimized architecture of variational optical Flow. *2018 IEEE East-West Design & Test Symposium (EWDTS)*. Kazan. 2018; 1-7. (In Eng.) DOI: 10.1109/EWDTS.2018.8524759
- [9] Kholopov E.Y., Bratulin A.V., Baranchikov A.I. Hardware Video Information Module Fordetecting Moving Objects in Real. *News of the Tula state university. Technical sciences*. 2018; 9:564-570. Available at: <https://elibrary.ru/item.asp?id=36394789> (accessed 06.06.2019). (In Russ., abstract in Eng.)
- [10] Vaganov S.E. A method for dynamic segmentation of a pair of sequential video-frames. *Computer Optics*. 2019; 43(1):83-89. (In Russ., abstract in Eng.) DOI: 10.18287/2412-6179-2019-43-1-83-89
- [11] Shinya M. Spatial anti-aliasing for animation sequences with spatio-temporal filtering. *Proceedings of the 20th annual conference on Computer graphics and interactive techniques (SIGGRAPH '93)*. ACM, New York, NY, USA. 1993; 289-296. (In Eng.) DOI: 10.1145/166117.166154
- [12] Elesina S.I., Kostyashkin L.N., Loginov A.A., Nikiforov M.B. [Images alignment in correlation-extreme navigation systems]. Radio Engineering, Moscow, 2015. (In Russ.)
- [13] Kuznetsov P.K., Martemyanov B.V., Semavin V.I. Machine Vision of Mobile Platforms. Method of the Optical Flow Analysis of Dynamic Images. *Vestnik komp'uternykh i informatsionnykh tekhnologii (Herald of computer and information technologies)*. 2014; 1(115):3-9. (In Russ., abstract in Eng.) DOI: 10.14489/vkit.2014.01.pp.003-009
- [14] Belyakov P.V., Nikiforov M.B. System-on-Chip variational optical flow computation. *Digital Signal Processing*. 2018; 3:76-82. Available at: <https://elibrary.ru/item.asp?id=36276676> (accessed 06.06.2019). (In Russ., abstract in Eng.)
- [15] Brox T., Malik J. Large Displacement Optical Flow: Descriptor Matching in Variational Motion Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2011; 33(3):500-513. (In Eng.) DOI: 10.1109/TPAMI.2010.143
- [16] Kuznesov P.K., Martemyanov B.V., Semavin V.I., Chekotilo E.Yu. Method for Computing Velocity of Moving Objects by Image Analysis. *Vestnik of Samara State Technical University (Technical Sciences Series)*. 2008; 2(22):96-110. Available at: <https://elibrary.ru/item.asp?id=13089112> (accessed 06.06.2019). (In Russ., abstract in Eng.)
- [17] Chao H., Gu Y., Napolitano M. A Survey of Optical Flow Techniques for Robotics Navigation Applications. *Journal of Intelligent & Robotic Systems*. 2014; 73(1-4):361-372. (In Eng.) DOI: 10.1007/s10846-013-9923-6
- [18] Cooke T. Two Applications of Graph-Cuts to Image Processing. *2008 Digital Image Computing: Techniques and Applications*. Canberra, ACT. 2008; 498-504. (In Eng.) DOI: 10.1109/DICTA.2008.32
- [19] Tao M., Bai J., Kohli P., Paris S. SimpleFlow: A Non-iterative, Sublinear Optical Flow Algorithm. *Computer Graphics Forum*. 2012; 31(2pt1):345-353. (In Eng.) DOI: 10.1111/j.1467-8659.2012.03013.x
- [20] Xu L., Jia J., Matsushita Y. Motion Detail Preserving Optical



- Flow Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2012; 34(9):1744-1757. (In Eng.) DOI: 10.1109/TPAMI.2011.236
- [21] Alpatov B.A., Babayan P.V., Balashov O.E., Stepashkin A.I. [The methods of the automatically object detection and tracking. Image processing and control]. Radiotekhnika, Moscow, 2008. (In Russ.)
- [22] Hisham M.B., Yaakob S.N., Raof R.A.A., Nazren A.B.A., Embedded N.M.W. Template Matching using Sum of Squared Difference and Normalized Cross Correlation. *2015 IEEE Student Conference on Research and Development (SCORED)*. Kuala Lumpur. 2015: 100-104. (In Eng.) DOI: 10.1109/SCORED.2015.7449303
- [23] Vassiliadis S., Hakkennes E.A., Wong J.S.S.M., Pechanek G.G. The sum-absolute-difference motion estimation accelerator. *Proceedings. 24th EUROMICRO Conference (Cat. No.98EX204)*. Vasteras, Sweden. 1998; 2:559-566. (In Eng.) DOI: 10.1109/EURMIC.1998.708071

Submitted 06.06.2019; revised 03.07.2019;
published online 25.07.2019.

About the authors:

Alexander V. Bratulin, postgraduate student, Department of Electronic Computers, Ryazan State Radio Engineering University (59/1 Gagarina St., Ryazan 390005, Russia), ORCID: <http://orcid.org/0000-0003-3745-296X>, bratulin@mail.ru

Mikhail B. Nikiforov, Director of the SEC "SpecEVM", Associate Professor of the Department of Electronic Computers, Ryazan State Radio Engineering University (59/1 Gagarina St., Ryazan 390005, Russia), Ph.D. (Engineering), Associate Professor, Corresponding member of the Academy of Education Informatization, ORCID: <http://orcid.org/0000-0002-4796-0776>, nikiforov.m.b@evm.rsreu.ru

Pavel V. Belyakov, engineer, Department of Electronic Computers, Ryazan State Radio Engineering University (59/1 Gagarina St., Ryazan 390005, Russia), ORCID: <http://orcid.org/0000-0002-0757-1418>, pvbel@rambler.ru

Yevgeny Yu. Kholopov, postgraduate student, Department of Electronic Computers, Ryazan State Radio Engineering University (59/1 Gagarina St., Ryazan 390005, Russia), ORCID: <http://orcid.org/0000-0002-0541-2878>, holopovev13@rambler.ru

All authors have read and approved the final manuscript.

