

ИНФОРМАЦИОННО-КОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ INFORMATION-COMMUNICATION TECHNOLOGIES

УДК 004.021:004.932

DOI: 10.24151/1561-5405-2018-23-5-495-501

ПЛИС-система выделения ключевых точек при обработке изображений

А.И. Шариков, Е.М. Шарикова

*Национальный исследовательский университет «МИЭТ», г. Москва,
Россия*

tohan757@yandex.ru

В управляющих системах в качестве центрального вычислительного узла все чаще применяются ПЛИС-системы. Однако высокие комбинационные задержки логических схем на гибкой логике и низкая максимальная тактовая частота ПЛИС-систем ограничивают их применение. В работе предложен оригинальный алгоритм выделения признаков изображения – ключевых (реперных) точек, которые могут быть использованы для поиска объекта в кадре по заданному эталону. Ключевые точки определены для участков изображения, которые подвергаются аффинным преобразованиям, т.е. вращению и масштабированию. Применение ПЛИС-систем обусловлено необходимостью использования предложенных алгоритмов во встраиваемых системах с разрешением камер 640×480 пикселей при скорости 25 кадров в секунду. В работе проведен анализ скорости обработки кадра алгоритмами, реализованными на других платформах. Приведены количественные характеристики ресурсов на примере семейства ПЛИС Spartan-6 фирмы Xilinx, качество которых оценивалось тестированием. Разработанный алгоритм характеризуется устойчивостью, инвариантностью, стабильностью, и его аппаратная реализация имеет сравнимую скорость выполнения по отношению к самым быстрым алгоритмам подобного класса. За счет малого количества потребляемых ресурсов ПЛИС имеется возможность масштабирования и повышения производительности.

Ключевые слова: обработка изображений; ключевые точки; FPGA

Для цитирования: Шариков А.И., Шарикова Е.М. ПЛИС-система выделения ключевых точек при обработке изображений // Изв. вузов. Электроника. – 2018. – Т. 23. – № 5. – С. 495–501. DOI: 10.24151/1561-5405-2018-23-5-495-501

FPGA System of Image Key Points Detecting

A.I. Sharikov, E.M. Sharikova

National Research University of Electronic Technology, Moscow, Russia

tohan757@yandex.ru

Abstract: In the control systems the FPGA systems more often are used as a central computing unit. However, high combination delays of the logic circuits on flexible logics and low maximum tact frequency of the FPGA systems restricts their application. The application of the behavioral hardware description in Verilog HDL for video processing on the hardware platform has been shown. An original algorithm has been proposed for the image features (key points) identifying, which can be used for the object search in the frame by a specified standard. The key points have been determined for the image areas, subjected to affine transformations, i.e. to rotation and scaling. The application of the FPGA systems is stipulated by the necessity of using the proposed algorithms in the systems being built-in with the camera resolution 640×480 pixels at 25 frames per second. The analysis of the frame processing speed by the algorithms, realized on other platforms has been carried out. The quantitative characteristics of the resources have been presented on an example of the Xilinx Spartan-6 FPGA family, their quality has been estimated by testing. The developed algorithm is characterized by the resistance, stability, invariance and its hardware implementation has comparable speed of fulfillment with respect to the most rapid algorithms of a similar class. Due to a small quantity of consumed FPGA resources there is the possibility of scaling and the efficiency improving.

Keywords: image processing; key points; FPGA

For citation: Sharikov A.I., Sharikova E.M. FPGA system of image key points detecting. *Proc. Univ. Electronics*, 2018, vol. 23, no. 5, pp. 495–501. DOI: 10.24151/1561-5405-2018-23-5-495-501

Введение. В настоящее время видеобработка, в частности выделение ключевых точек, применяется в медицине, электронной и военной промышленности [1, 2]. В управляющих системах в качестве центрального вычислительного узла все чаще используются ПЛИС. По сравнению с процессорами общего назначения ПЛИС-системы имеют следующее преимущество: аппаратные решения, построенные на структурах логических элементов, способствуют увеличению скорости выполнения вычислений при возможности распараллеливания задачи. Однако высокие комбинационные задержки логических схем на гибкой логике и низкая максимальная тактовая частота ПЛИС-систем ограничивает их применение.

Постановка задачи. Задача поиска объекта на видеопоследовательности решается с помощью различных инструментов: нейронных сетей, корреляционных, контурных и дескрипторных алгоритмов [3, 4]. Последний основан на вычислении особенностей участков изображений, ключевых точек, каждая из которых имеет уникальный численный код, называемый дескриптором (набором дескрипторов). Алгоритм принятия решения о том, является ли точка ключевой, называется детектором [5, 6]. Использование аппаратной ПЛИС-системы предполагает разработку как алгоритма, так и схемотехнического описания с соблюдением требований, которые предъявляются к дескрипторным алгоритмам, а именно инвариантность к поворотам, масштабированию и изменению контраста [7].

Разработанный алгоритм, наряду с учетом особенностей дескрипторного подхода, должен иметь возможность реализации его на ПЛИС. В качестве целевой платформы используется микросхема архитектуры FPGA фирмы Xilinx семейства Spartan-6. Видеоинформация обрабатывается с помощью монитора и видеокамеры, подключенных к плате с установленной микросхемой. Алгоритм работает в два этапа (рис. 1).

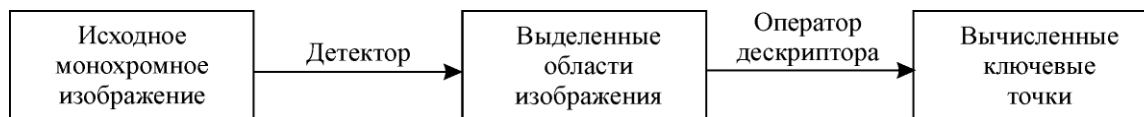


Рис.1. Алгоритм вычисления ключевых точек

Fig.1. Algorithm for calculating key points

Алгоритм детектора ключевых точек. Для каждого пикселя кадра проводится сравнение яркостей пикселей P_k окружности с пикселем, расположенным в центре. При отличии яркости крайних точек из P_k по отношению к центральному пикселю больше чем на t в большую или меньшую сторону признаку $S_i(x, y)$ присваивается число 1 или -1 соответственно. В дальнейшем вычисляется сумма всех признаков $G(x, y)$. В случае ее соответствия заданному интервалу точка принимается в качестве особой:

$$G(x, y) = \sum_{i=1}^{P_k} S_i(x, y),$$

$$S_i(x, y) = \begin{cases} -1, & \text{если } I_i(x, y) < I_p(x, y) - t_s, \\ 1, & \text{если } I_i(x, y) \geq I_p(x, y) + t_s, \\ 0 & \text{иначе,} \end{cases}$$

где $I_i(x, y) = P\left(x + R \cos\left(\frac{2\pi i}{P_k}\right), y + R \sin\left(\frac{2\pi i}{P_k}\right)\right)$; R – радиус анализируемой области;

$I_p(x, y) = P(x, y)$; $P(x, y)$ – яркость пикселя в точке с координатами (x, y) ; I_i – интенсивность на границе области; I_p – интенсивность центрального пикселя; t_s – порог детектирования. Если $G(x, y) < -t_s$ или $G(x, y) > t_s$, то точка является особой и принимается как признак объекта.

Вычисление дескриптора. Дескриптор $H(x, y)$ имеет длину, равную P_k бит. Каждый бит дескриптора является результатом сравнения интенсивности пикселя $I_p(x, y)$ с пикселем $I_i(x, y)$. Двоичный код дескриптора представляется десятичным числом. Для достижения инвариантности к вращению над полученным дескриптором проводится операция циклического сдвига до тех пор, пока не будет получено минимальное значение:

$$H(x, y) = \sum_{i=0}^{P_k-1} (\tau_i(x, y) \cdot 2^i),$$

$$\tau_i(x, y) = \begin{cases} 0, & \text{если } I_i(x, y) < I_p(x, y), \\ 1, & \text{если } I_i(x, y) \geq I_p(x, y), \end{cases}$$

$$H(x, y) \rightarrow H_0(x, y),$$

где (x, y) – координата пикселя; $H_0(x, y) = \min_i (H(x, y) \ll i)$; $H(x, y)$ – двоичное число, являющееся дескриптором.

Причина применения подобного подхода обусловлена простотой реализации.

Аппаратная реализация. Для создания ПЛИС-системы, представляющей собой модули, построенные на наборах программируемых логических элементов, задача разбивается на несколько независимых модулей: модуль ввода-вывода видеоинформации; ядро для выделения реперных точек; модуль отображения ключевых точек. В системе дополнительно к программируемой логике применяется микросхема динамической оперативной памяти DDR, предназначенная для разделения выводимых и обрабатываемых кадров. Для доступа к микросхеме ОЗУ используется модуль арбитража (блок «Контроллер памяти») в структуре ПЛИС-системы (рис. 2).

В табл.1 приведен относительный объем занимаемых ресурсов на ПЛИС семейства Spartan-6.

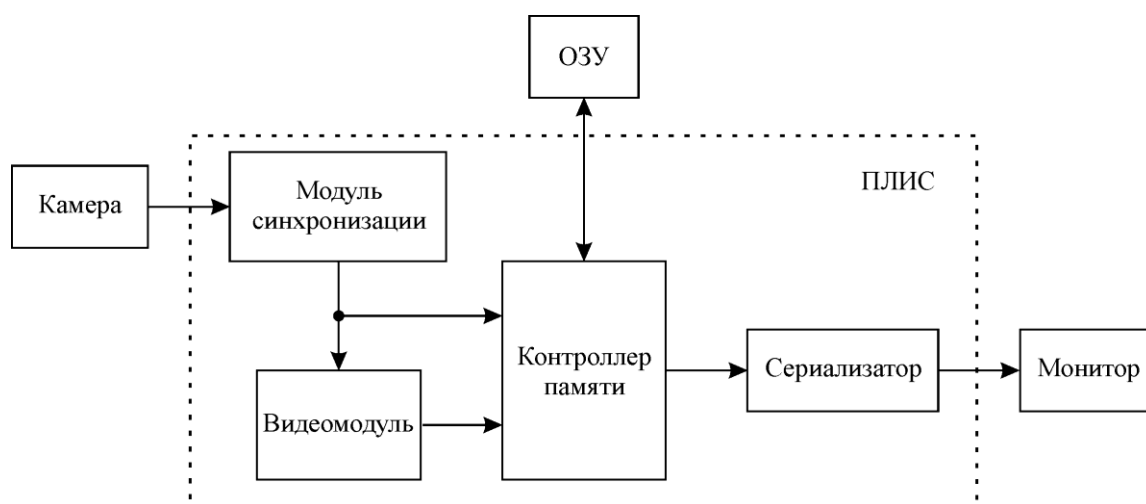


Рис.2. Структура ПЛИС-системы
Fig.2. Structure of the FPGA system

Таблица 1

Относительный объем, %, занимаемых ресурсов на ПЛИС семейства Spartan-6

Table 1

Relative volume, %, of occupied resources on FPGAs of the Spartan family-6

Тип микросхемы	Блоки RAM		Ячейки LUT (3677)	Регистры Flip-Flops (1176)
	Количество блоков 18 Kb BRAM (12)	Объем блочной памяти (108 кбит)		
XC6SLX4	100,00	50,00	153,21	24,50
XC6SLX16	37,50	18,75	40,35	6,45
XC6SLX25	23,08	11,54	24,46	3,91
XC6SLX45	10,34	5,17	13,47	2,15
XC6SLX75	6,98	3,49	7,88	1,26
XC6SLX100	4,48	2,24	5,81	0,93
XC6SLX150	4,48	2,24	3,99	0,64
XC6SLX25T	23,08	11,54	24,46	3,91
XC6SLX45T	10,34	5,17	13,47	2,15
XC6SLX75T	6,98	3,49	7,88	1,26
XC6SLX100T	4,48	2,24	5,81	0,93
XC6SLX150T	4,48	2,24	3,99	0,64

Примечание. В скобках указано количество ресурсов, полученных в результате синтеза алгоритма на FPGA.

Анализ данных таблицы позволяет сделать вывод о возможном повышении производительности за счет применения дополнительных ядер в структуре одной микросхемы и распараллеливания задачи на несколько потоков делением кадра на отдельные части, так как в рамках ПЛИС остаются незанятые ресурсы (LUT, Flip-Flops, BRAM).

Строка «FPGA» отражает скорость обработки разработанного алгоритма. Максимальная частота, равная 26 МГц, получена анализатором задержек фирмы Xilinx на разработанное описание. При этом для камер с разрешением 640×480 скорость обработки кадра составляет 11,6 мс. При реализации на платформе FPGA она не уступает существующему алгоритму ORB выделения ключевых точек (алгоритм ORB взят в качестве оценочного, так как является в настоящее время одним из быстроедействующих). При этом скорость работы ПЛИС-системы слабо зависит от тактовой частоты. Это связано с тем, что при использовании процессора в качестве вычислительного ядра все операции в нем осуществляются последовательно в отличие от ПЛИС, где время выполнения определяется комбинационной задержкой блока «Модуль-детектор» (рис. 3).

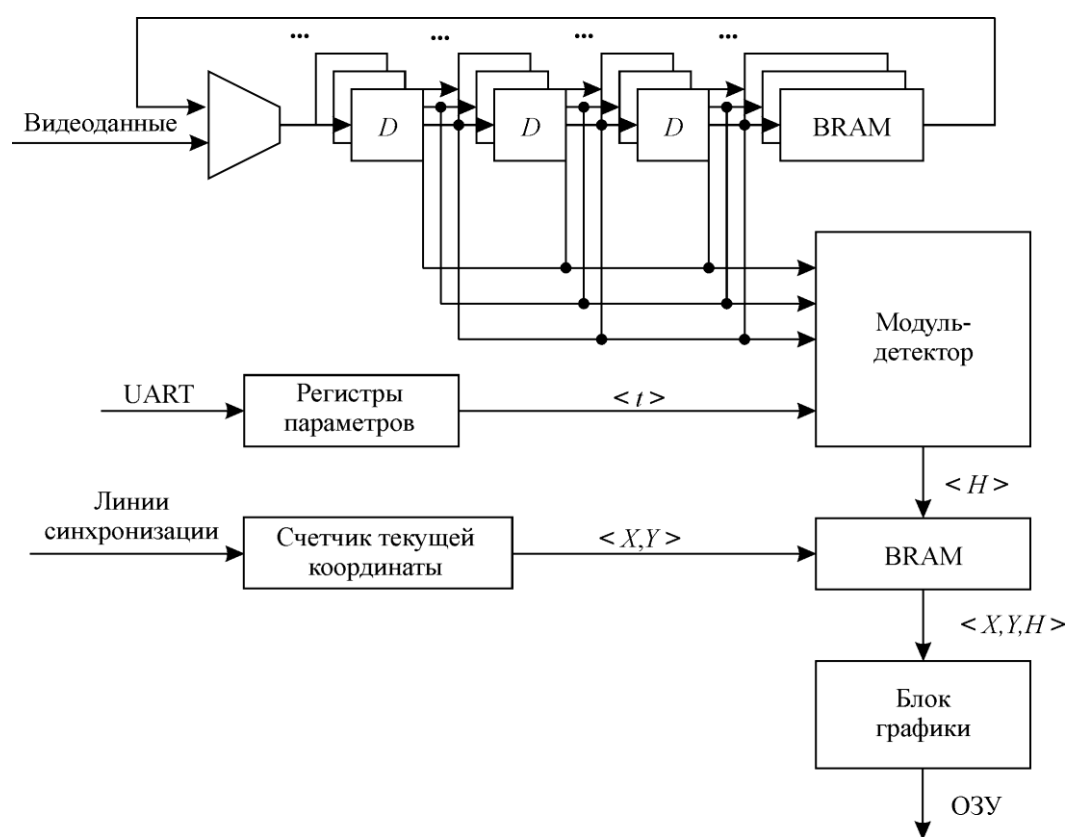


Рис.3. Структура блока «Видеомодуль»: D – D-триггеры; BRAM – блочная память; модуль-детектор – комбинационный блок алгоритма; UART – внешний интерфейс для управления

Fig.3. Structure of the module «Video module»: D – D-triggers; BRAM – block RAM; module-detector – combinational algorithm block; UART – external interface for control

Результаты тестирования. Для оценки эффективности разработанного алгоритма проведено моделирование на ПК. В качестве входной информации использовалось статическое изображение. Результаты работы алгоритма и оценка скорости детектирования ключевых точек представлены на рис. 4 и в табл.2 соответственно.

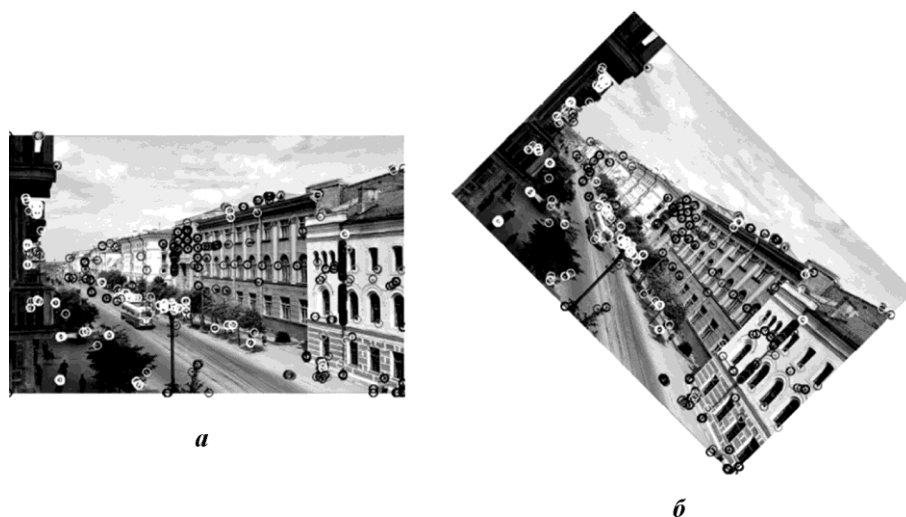


Рис.4. Пример работы алгоритма для исходного монохромного изображения:
a – исходное изображение; *б* – повернутое изображение
 Fig.4. An example of how the program works for the original monochrome image:
a – rotated image; *b* – original image

Таблица 2
Оценка времени обработки кадра с выделением
ключевых точек

Evaluation of the processing time of the frame
with the selection of key points

Table 2

Платформа	Частота, МГц	Время обработки, мс
ARM (OC Android) [5]	1000	66,6
Soc Zynq-7000 [1]	100	18
Intel i7 [5]	2800	15,3
FPGA	26	11,8

Примечание. Размеры кадра 640×480 пикселей, 8 бит/пиксель.

Закключение. Разработанный алгоритм характеризуется устойчивостью, инвариантностью и стабильностью. При этом его аппаратная реализация имеет сравнимую скорость выполнения по отношению к самым быстрым алгоритмам подобного класса, например, таким как ORB. За счет малого количества потребляемых ресурсов ПЛИС есть возможность масштабирования и повышения производительности.

Данная ПЛИС-система может быть использована в качестве базовой для построения систем обнаружения объектов в видеопотоке.

Литература

1. Lee K., Byun K. A hardware design of optimized orb algorithm with reduced hardware cost // Advanced Science and Technology Letters. – 2013. – No. 43. – P. 58–62.
2. Колдаев В.Д., Гагарина Л.Г., Теплова Я.О. Методологические аспекты сшивания кадров аэрофотоснимков // Информатизация и связь. – 2017. – №4. – С. 26–29.
3. Altwaijry H., Veit A., Belongie S. Learning to detect and match keypoints with deep architectures // British Machine Vision Conference (BMVC 2016). – 2016. – 124 p.
4. Сакович И.О., Белов Ю.С. Обзор основных методов контурного анализа для выделения контуров движущихся объектов // Инженерный журнал: наука и инновации. – 2014. – Вып. 12. – С. 1–8.

5. Rublee E., Rabaud V., Konolige K. ORB: An efficient alternative to SIFT or SURF // Computer Vision and 2011 IEEE International Conference. – 2011. – Nov. – P. 2564–2571.
6. Ramakrishna M., Shylaja S. Is ORB efficient over surf for object recognition? // International J. of Advanced Research in Science Engineering and Technology. – 2014. – Vol.3. – No. 8. – P. 2783–2788.
7. Tuytelaars T., Mikolajczyk K. Local invariant feature detectors: A survey // Foundations and trends in computer graphics and vision. – 2007. – No. 3(3). – P. 177–280.

Поступила 27.09.2017 г.; принята к публикации 19.06.2018 г.

Шариков Антон Игоревич – стажер-исследователь Научно-исследовательского института вычислительных средств и систем управления Национального исследовательского университета «МИЭТ» (Россия, 124498, г. Москва, г. Зеленоград, пл. Шокина, д. 1), tohan757@yandex.ru

Шарикова Елена Михайловна – инженер Научно-исследовательского института вычислительных средств и систем управления Национального исследовательского университета «МИЭТ» (Россия, 124498, г. Москва, г. Зеленоград, пл. Шокина, д. 1), elena.biserova.93@mail.ru

References

1. Lee K., Byun K. A hardware design of optimized orb algorithm with reduced hardware cost. *Advanced Science and Technology Letters*, 2013, 43, pp. 58–62.
2. Koldaev VD, Gagarina LG, Teplova Ya.O. Methodological aspects of stitching up aerial photographs. *Informatizatsiya i svyaz' = Informatization and communication*, 2017, no. 4, pp. 26–29. (In Russian).
3. Altwaijry H., Veit A., Belongie S. Learning to detect and match keypoints with deep architectures. *British Machine Vision Conference*, 2016.
4. Sakovich I.O., Belov Yu.S. Review of the contour analysis basic methods to distinguish moving object contours. *Inzhenernyy zhurnal: nauka i innovatsii = Engineering Journal: Science and Innovation*, 2014, iss. 12, pp. 1–8. (In Russian).
5. Rublee E., Rabaud V., Konolige K. ORB: An efficient alternative to SIFT or SURF. *Computer Vision and 2011 IEEE International Conference*, 2011, pp. 2564–2571.
6. Ramakrishna M., Shylaja S. Is ORB efficient Over Surf for Object Recognition? *International Journal of Advanced Research in Science Engineering and Technology*, 2014, vol.3, no. 8, pp. 2783–2788.
7. Tuytelaars T., Mikolajczyk K. Local invariant feature detectors: A survey. *Foundations and trends in computer graphics and vision*, 2007, no. 3(3), pp. 177–280.

Submitted 27.09.2017; Accepted 19.06.2018.

Information about the authors:

Anton I. Sharikov – Researcher of the Scientific Research Institute of Computing Means and Control Systems, National Research University of Electronic Technology (Russia, 124498, Moscow, Zelenograd, Shokin sq.,1), tohan757@yandex.ru

Elena M. Sharikova – Engineer of the Scientific Research Institute of Computing Means and Control Systems, National Research University of Electronic Technology (Russia, 124498, Moscow, Zelenograd, Shokin sq.,1), elena.biserova.93@mail.ru