

Разработка и исследование реконфигурируемых быстродействующих алгоритмов распознавания изображений для оценки дорожной ситуации на базе специализированных мобильных устройств с параллельно-конвейерной архитектурой*

С.В. Шидловский, Д.В. Шашев, А.С. Бондарчук, В.В. Шатравин, М.В. Окунский

В статье обобщаются результаты исследования применения перестраиваемых вычислительных сред для задач обработки изображений, получаемых на борту беспилотных транспортных систем. Был предложен новый подход к разработке специализированных быстродействующих алгоритмов распознавания изображений дорожной сцены, предназначенных для аппаратного выполнения на вычислителях с параллельно-конвейерной архитектурой по модели перестраиваемых вычислительных сред.

Рассмотрены архитектуры и алгоритмическое обеспечение таких сред для задач оценки дорожной ситуации. Новый сформированный класс алгоритмов распознавания изображений основан на вычислении дескрипторов LBP, HOG и классификатора SVM. Показано, что вычисления дескрипторов HOG в слое перестраиваемой вычислительной среды выполняются за один машинный такт для всего изображения, а время выполнения (при реализации на FPGA Intel Arria 10) не превышает 20 нс.

Ключевые слова: высокопроизводительные вычислительные системы, параллельные вычисления, перестраиваемые вычислительные среды, беспилотные транспортные средства.

**Работа выполнена при финансовой поддержке РФФИ (проект №19-29-06078).*

Введение

Развитие автономных беспилотных транспортных систем (БТС) и их повсеместное использование устанавливает актуальную научную проблему, заключающуюся в распознавании изображений непосредственно на борту БТС. Ключевой аспект проблемы – постоянно растущий объем видеоданных, а также вычислительная сложность алгоритмов, что приводит к необходимости использования дорогих массивных вычислительных устройств, использование которых в составе мобильных устройств фактически невозможно по ряду экономических и технических причин. Необходим подход, который позволит отойти от стандартных принципов адаптации существующих методов, а также от постоянного наращивания используемых вычислительных ресурсов.

Современное состояние исследований рассматриваемой научной проблематики заключается в тенденции построения реконфигурируемых многопроцессорных вычислителей для решения задач обработки, классификации и распознавания изображений [1–11]. Суть этой тенденции заключается в том, что архитектура вычислительной системы адаптируется под структуру решаемой задачи. Общеизвестно, что максимальное быстродействие обеспечивают вычислительные системы с аппаратной реализацией алгоритма. На данный



ШИДЛОВСКИЙ

Станислав Викторович
Национальный исследовательский
Томский государственный университет



ШАШЕВ

Дмитрий Вадимович
Национальный исследовательский
Томский государственный университет



БОНДАРЧУК

Антон Сергеевич
Национальный исследовательский
Томский государственный университет



ШАТРАВИН

Владислав Владимирович
Национальный исследовательский
Томский государственный университет



ОКУНСКИЙ

Михаил Викторович
Национальный исследовательский
Томский государственный университет

момент наиболее распространено использование гибридных вычислительных устройств на базе FPGA и GPU. В многочисленных научных трудах как зарубежных, так и отечественных ученых, а также коммерческих работах крупных компаний, таких как Google, NVidia и пр., рассматриваются преимущества использования подобных вычислительных систем, при которых увеличение ускорения обработки изображений может достигать более 200 раз.

Однородные вычислительные устройства, частным случаем которых является перестраиваемые вычислительные среды (ПВС), уже продемонстрировали свою эффективность в задачах обработки данных. В основе тензорного процессора Google TPU лежат систолические матрицы [12], что обеспечивает существенный прирост скорости выполнения матричного умножения. А одним из наиболее многообещающих направлений сейчас являются так называемые «системы на пластине» (system-on-wafer), представляющие собой СБИС, занимающие полную площадь кремниевой пластины [13, 14]. Такие вычислители включают в себя триллионы транзисторов, сотни тысяч вычислительных ядер, а их вычислительные блоки чередуются с элементами памяти. В качестве примера можно привести устройство Wafer-Scale Engine 2 компании Cerebras (рис. 1), представляющее собой кремниевую пластину с площадью 46 тыс. мм² при техпроцессе 7 нм и насчитывающее 2.6 трлн транзисторов или 850 тыс. вычислительных

ядер, что позволяет реализовать даже крупные языковые модели (LLM) [13–15]. В качестве другого примера можно привести систему Tesla Dojo, также представляющую собой двумерную решетку из однородных вычислителей на цельной кремниевой пластине и предназначенную для решения задач искусственного интеллекта [16].

Целью данной работы является проведения исследований по адаптации алгоритмов распознавания изображений дорожной сцены (в частности, распознавания автомобилей и пешеходов), использующих дескрипторы LBP и HOG, а также классификатор SVM, под аппаратную реализацию на перестраиваемых вычислительных средах. В результате чего необходимо разработать методику формирования алгоритмов в виде автоматных отображений, описывающих принцип работы вычислительной среды, а также сформировать новый класс алгоритмов распознавания изображений.

Перестраиваемые вычислительные среды

ПВС является дискретная математическая модель высокопроизводительной вычислительной системы, состоящей из одинаковых и одинаково соединенных друг с другом простейших универсальных элементов, далее называемых элементарными вычислителями (ЭВ). ПВС представляет собой решетку с двумя или более количеством осей, в узлах которой расположены ЭВ, способные выполнить некоторый набор операций. Для указания того, какую операцию или функцию передачи сигналов требуется выполнить в необходимый момент времени, на каждый элементарный вычислитель ПВС подается соответствующий сигнал настройки, при этом передача сигналов в ПВС может осуществляться от любого ЭВ к любому другому ЭВ [17].

Размер решетки ПВС, предназначенной для обработки изображения, совпадает с размером данного изображения в пикселях. Таким образом, значения пикселей обрабатываемого изображения подаются на соответствующие им элементарные вычислители ПВС. С выходов элементарных вычислителей поступает необходимый результат цифровой обработки пикселей изображения.

Перестраиваемые вычислительные среды позволяют эффективно решать задачи цифровой обработки изображений. Эффективность достигается за счет возможности программно настроить алгоритмы в процессе функционирования вычислительной системы и/или перед началом ее работы. Принципы построения ПВС позволяют достигнуть высоких показателей быстродействия, надежности и адаптивности вычислительной системы. Высокие качественные показатели реализации алгоритмов обработки изображений на ПВС обеспечиваются за счет распараллеливания процессов обработки пикселей изображения, взаимозаменяемости ЭВ, их идентичности и одинаковых связей между ними, а также за счет способности изменять свою структуру для выполнения необходимых преобразований.



Рис. 1. Вычислитель Wafer-Scale Engine 2 M.

Вычисление LBP дескриптора

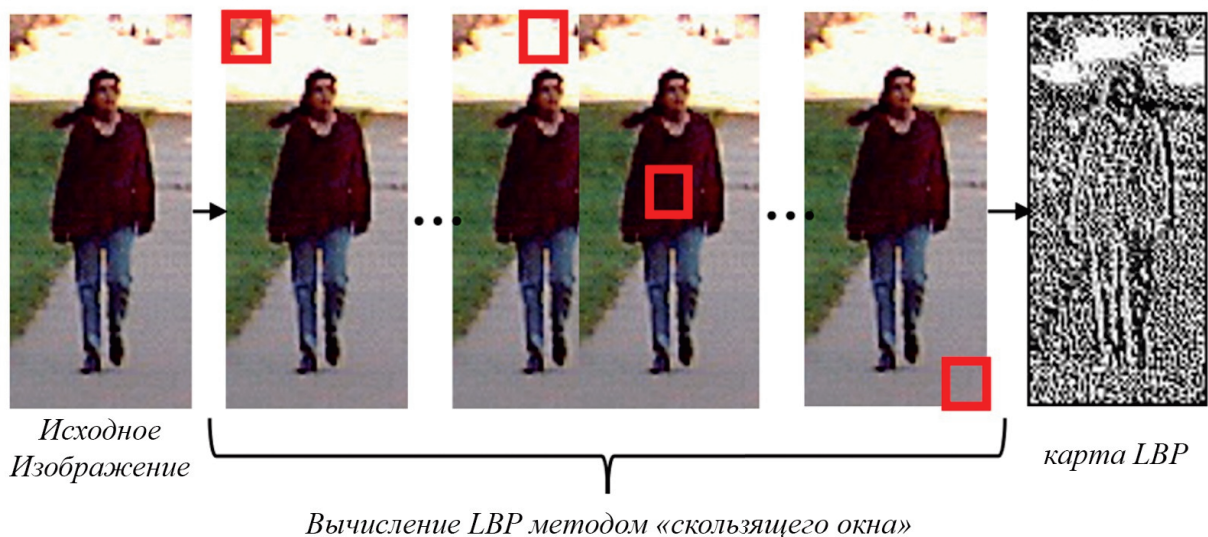
Так, для вычисления признаков LBP синтезирован ЭВ, описываемый следующей системой формул:

$$\begin{cases}
 f_1 = x(\bar{z}_2\bar{z}_3\bar{z}_4 \vee \bar{z}_1\bar{z}_2\bar{z}_3 \vee z_2z_3z_4\bar{z}_1), \\
 f_2 = x(\bar{z}_1\bar{z}_2 \vee z_3z_4\bar{z}_1 \vee \bar{z}_2\bar{z}_3\bar{z}_4), \\
 f_3 = x(z_4\bar{z}_1\bar{z}_2 \vee z_3\bar{z}_1\bar{z}_2 \vee z_1\bar{z}_2\bar{z}_3\bar{z}_4), \\
 f_4 = x(z_4\bar{z}_1\bar{z}_2 \vee z_3\bar{z}_1\bar{z}_2 \vee z_2\bar{z}_1\bar{z}_3 \vee z_1\bar{z}_2\bar{z}_3\bar{z}_4), \\
 f_5 = x(z_3z_4\bar{z}_1\bar{z}_2 \vee z_2\bar{z}_1\bar{z}_3 \vee z_1\bar{z}_2\bar{z}_3\bar{z}_4), \\
 f_6 = x(z_3z_4\bar{z}_1 \vee z_2\bar{z}_1 \vee z_1\bar{z}_2\bar{z}_3\bar{z}_4), \\
 f_7 = x(z_2z_4\bar{z}_1 \vee z_2z_3\bar{z}_1 \vee z_1\bar{z}_2\bar{z}_3\bar{z}_4), \\
 f_8 = x(\bar{z}_2\bar{z}_3\bar{z}_4 \vee z_4\bar{z}_1\bar{z}_3 \vee z_2z_3\bar{z}_1), \\
 f = \left(\sum_{n=1}^8 B(y_n, x) 2^{n-1} \right) z_1\bar{z}_2\bar{z}_3\bar{z}_4.
 \end{cases} \quad (1)$$

Здесь математическая модель ЭВ включает в себя информацию о количестве выходов (набор f_n), соединяющих элементарный вычислитель с соседними ЭВ, количестве информационных входов (x и набор y_n), а также о выходах (f) самой ПВС, получаемую путем соединения множества ЭВ между собой. Кроме того, в модель входит информация о настройке каждого ЭВ (набор z_k ($k=1, 2, \dots, 4$)), с помощью которого происходит реконфигурация модели.

В модели ПВС все пиксели изображения обрабатываются параллельно и одновременно для вычисления локального бинарного шаблона каждого пикселя. Полученные коды LBP в дальнейшем могут быть

Классический LBP



LBP для ПВС

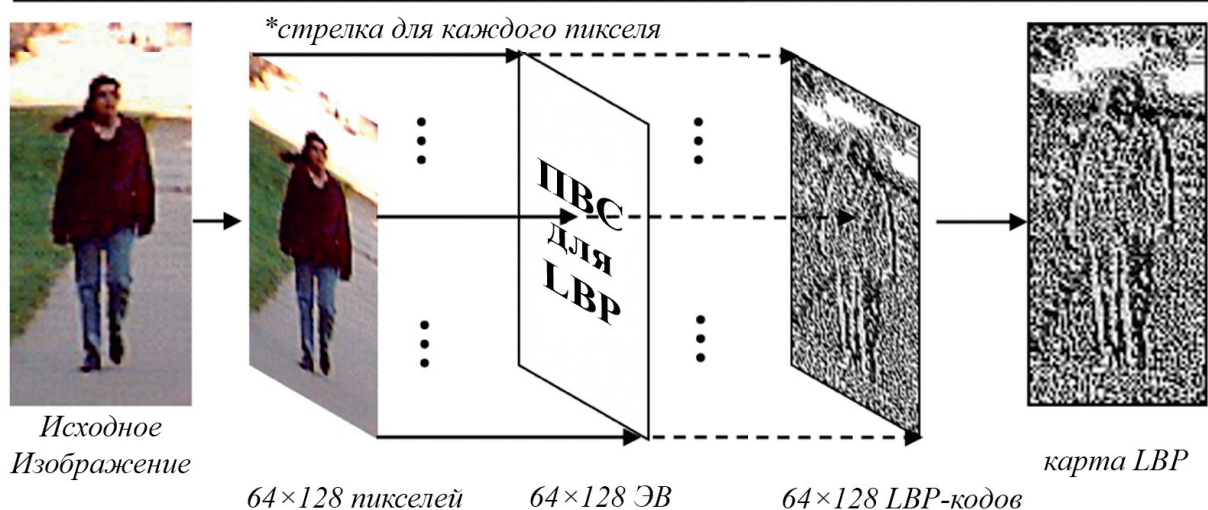


Рис. 2. Классический и предложенный способы нахождения дескриптора LBP.

использованы для классификации изображений. Например, результирующие изображения, полученные с помощью данного метода извлечения признаков, можно использовать в качестве данных для моделей машинного обучения.

На рис. 2 представлены принципы нахождения дескриптора LBP классическим способом (используя «скользящее окно») и предлагаемым способом – когда каждый пиксель изображения подается на свой, отдельный, элемент ПВС [18]. В результате всё изображение обрабатывается за один машинный такт работы ПВС. Длительность этого такта определяется в зависимости от того, какие функциональные возможности заложены в элементарный вычислитель на этапе проектирования.

В работе также были рассмотрены модификация алгоритма HOG для выделения признаков объекта интереса на изображении. Значения градиентов вычисляются в горизонтальном и вертикальном направлениях с помощью одномерных дифференцирующих масок: $[-1, 0, 1]$ и $[-1, 0, 1]^T$. Затем рассчитываются направление и величина градиента для каждого пикселя в изображении.

Вычисление HOG дескриптора

ЭВ, реализующий классическое представление алгоритма HOG, распараллеленного на вычис-

лительную архитектуру ПВС, можно описать следующей системой формул:

$$\begin{cases} f_1 = x(z_3 z_4 \bar{z}_1 \vee z_2 \bar{z}_1 \vee z_1 \bar{z}_2 \bar{z}_3 \bar{z}_4), \\ f_2 = x(\bar{z}_2 \bar{z}_3 \bar{z}_4 \vee \bar{z}_1 \bar{z}_2 \vee z_3 z_4 \bar{z}_1), \\ f_3 = x(z_3 \bar{z}_1 \bar{z}_2 \vee z_2 \bar{z}_1 \bar{z}_3 \vee z_4 \bar{z}_1 \bar{z}_2 \vee z_1 \bar{z}_2 \bar{z}_3 \bar{z}_4), \\ f_4 = x(\bar{z}_2 \bar{z}_3 \bar{z}_4 \vee z_4 \bar{z}_1 \bar{z}_3 \vee z_2 z_3 \bar{z}_1), \\ m = z_1 \bar{z}_2 \bar{z}_3 \bar{z}_4 \sqrt{(y_1 - y_2)^2 + (y_3 - y_4)^2}, \\ \theta = \arctg\left(\frac{y_1 - y_2}{y_3 - y_4}\right) z_1 \bar{z}_2 \bar{z}_3 \bar{z}_4. \end{cases} \quad (2)$$

Здесь f_i ($i=1, 2, \dots, 4$) – выходы ЭВ, предназначенные для соединения с соседними ЭВ в ПВС; m – выход ЭВ, характеризующий величину изменения яркости (длину вычисленного вектор) изображения; θ – выход ЭВ, характеризующий направление изменения яркости; x – вход ЭВ, на который подается значение обрабатываемого пикселя изображения; y_i ($i=1, 2, \dots, 4$) – вход ЭВ, на который подаются значения соответствующих f_i соседних ЭВ, несущих информацию об окружающих пикселях изображения; z_k ($k=1, 2, \dots, 4$) – входы ЭВ, с помощью которых происходит реконфигурация модели.

Для извлечения признаков бинарного изображения разработан алгоритм нахождения градиента по аналогии с алгоритмом HOG. Бинарный градиент состоит из величины m и направления ϕ изменения яркости соседних пикселей изображения с 0 на 1 или наоборот. Направление градиента может принимать три значения: 180, 225 и 270°. Если изменение яркости пикселей (между текущим пикселем и пикселями – соседями) отсутствует, то $m=0$, что означает отсутствие градиента. На рис. 3 представлены четыре возможных варианта значений градиента для рассматриваемого пикселя x по отношению к соседним пикселям y_1 и y_2 .

Значение направления градиента было закодировано двузначным двоичным числом $\phi = \phi_1 \phi_2$ согласно таблице 1.

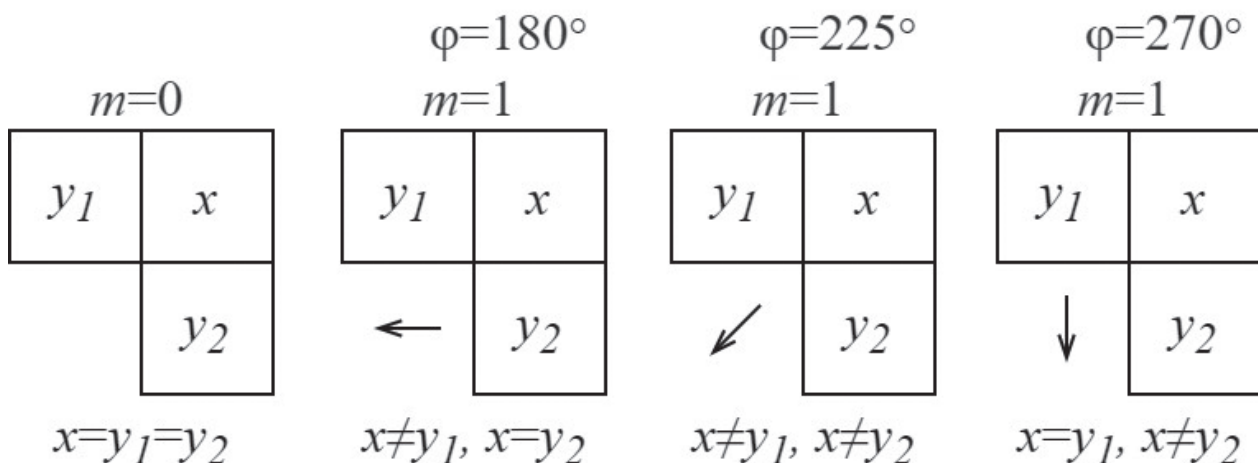


Рис. 3. Варианты значений бинарного градиента.

Таблица 1. Кодирование значений направления бинарного градиента

Φ	«Градиент отсутствует»	180°	225°	270°
Φ_1	0	0	1	1
Φ_2	0	1	0	1

Вектор признаков, состоящий из значений m , Φ_1 и Φ_2 бинарного градиента пикселей изображения, далее может использоваться для классификации объектов с помощью метода опорных векторов (SVM).

На рис. 4 представлен результат вычисления направления бинарного градиента.

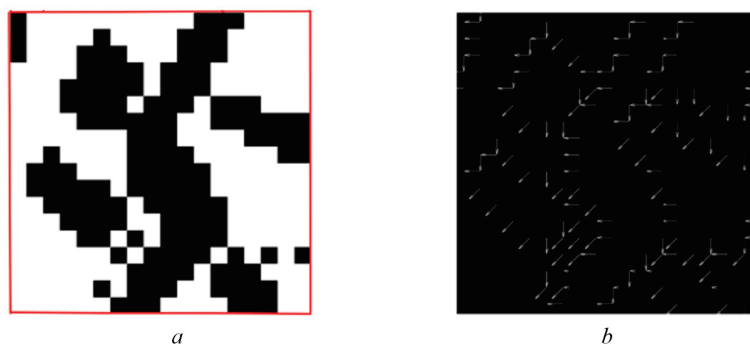


Рис. 4. Визуализация бинарного градиента на изображении: *a* – исходное бинарное изображение, *b* – визуализация бинарного градиента в каждом пикселе изображения

Классификатор SVM

Система уравнений, реализующая классификацию SVM на базе ЭВ, может быть представлена следующим образом [19]:

$$\begin{cases}
 f_1 = x(\bar{z}_1\bar{z}_2\bar{z}_3 \vee z_2z_4\bar{z}_1 \vee z_2z_3\bar{z}_1 \vee \bar{z}_2\bar{z}_3\bar{z}_4), \\
 f_2 = x(z_3z_4\bar{z}_1 \vee z_2\bar{z}_1\bar{z}_3 \vee z_2\bar{z}_1\bar{z}_4 \vee z_1\bar{z}_2\bar{z}_3\bar{z}_4), \\
 W = (w_m(x \vee (x \cdot \bar{y}_2 \vee \bar{x} \cdot y_2)) + w_{\Phi_2}(x \cdot (x \cdot \bar{y}_2 \vee \bar{x} \cdot y_2) \vee \\
 \vee \bar{x} \cdot (x \cdot \bar{y}_2 \vee \bar{x} \cdot y_2)))(\bar{z}_1\bar{z}_2\bar{z}_3\bar{z}_4 + z_2z_3z_4\bar{z}_1) + \\
 + w_{\Phi_1}(x \cdot \bar{y}_2 \vee \bar{x} \cdot y_2)(\bar{z}_1\bar{z}_2\bar{z}_3 + z_3\bar{z}_1\bar{z}_2 + z_3z_4\bar{z}_1 + \bar{z}_2\bar{z}_3\bar{z}_4) + \\
 + (w_m((x \cdot \bar{y}_1 \vee \bar{x} \cdot y_1) \vee (x \cdot \bar{y}_2 \vee \bar{x} \cdot y_2)) + \\
 + w_{\Phi_2}((x \cdot \bar{y}_1 \vee \bar{x} \cdot y_1) \cdot (x \cdot \bar{y}_2 \vee \bar{x} \cdot y_2) \vee \\
 \vee (x \cdot \bar{y}_1 \vee \bar{x} \cdot y_1) \cdot (x \cdot \bar{y}_2 \vee \bar{x} \cdot y_2)))(z_4\bar{z}_1\bar{z}_2 + z_3\bar{z}_1\bar{z}_2 + z_1\bar{z}_2\bar{z}_3\bar{z}_4) + \\
 + (w_m((x \cdot \bar{y}_1 \vee \bar{x} \cdot y_1) \vee x) + w_{\Phi_2}((x \cdot \bar{y}_1 \vee \bar{x} \cdot y_1) \cdot \bar{x} \vee \\
 \vee (x \cdot \bar{y}_1 \vee \bar{x} \cdot y_1) \cdot x))z_2\bar{z}_1\bar{z}_3 + x(w_{\Phi_1}(z_2\bar{z}_1\bar{z}_3 + z_2\bar{z}_1\bar{z}_4) + w_m z_2z_3\bar{z}_1\bar{z}_4).
 \end{cases} \quad (3)$$

Здесь f_i ($i=1, 2$) – выходы ЭВ для соединения с соседними элементами; W – выход ЭВ, характеризующий «вес» обрабатываемого пикселя в алгоритме SVM; x – вход ЭВ, на который подается значение обрабатываемого пикселя; y_i ($i=1, 2$) – входы ЭВ, для связи с соседними элементами; w_m – весовые значение для составляющей m ; w_{Φ_i} ($i=1, 2$) – весовые значения для кодированного направления яркости; z_k ($k=1, 2, \dots, 4$) – входы ЭВ, с помощью которых происходит реконфигурация модели.

Рассмотрим процесс распознавания объекта интереса на изображении с помощью модели перестраиваемой вычислительной среды на примере выделения характеристик бинарного градиента. Каждый пиксель исходного полутонового изображения подается на «свой» персональный ЭВ ПВС. Для реализации алгоритма распознавания используется три слоя ПВС.

На первом слое происходит формирование и выбор бинарного изображения из входного 8-битового представления полутонового изображения. На втором слое ПВС осуществляется вычисление величины градиента и его направления с привлечением аппарата булевого дифференцирования и синтезированной логики работы ЭВ. В качестве дополнительного информационного входа каждого ЭВ ПВС используется соответствующая компонента вектора весов, формируемого на основе классификатора SVM. Для формирования такого вектора необходимо, во-первых, выбрать или сформировать базу данных изображений с объектами интереса (для случая распознавания пешеходов была выбрана база изображений от Massachusetts Institute of Technology (MIT), the Center for Biological and Computational Learning (CBCL)). Результат визуализации весов для компонентов m и Φ_i датасета CBCL представлен на рис. 5. Во-вторых, необходимо осуществить поэтапную обработку каждого изображения из базы данных путем вычисления для каждого пикселя каждого изображения из базы данных значений бинарного градиента и его направления. В-третьих, сформировать на основе полученных данных векторы признаков с сохранением их пространственного расположения на изображении и применить выбранный классификатор. В результате этого получим пространственное расположение предобученного вектора весов, соответствующее значению компоненты которого мы вносим в качестве дополнительной информации в соответствующий ему по пространственному расположению ЭВ ПВС.

В процессе распознавания компоненты вектора весов перемножаются

с полученными характеристиками бинарного градиента пикселя обрабатываемого изображения. В третьем слое полученные результаты складываются между собой в рамках области, соответствующей размеру изображения, на котором осуществлялось обучение (операция свертки). В зависимости от установленного процента распознавания можно судить о наличии объекта интереса (пешехода) в области.

Экспериментальные исследования

Реализованные модели ПВС с привлечением языка System Verilog на аппаратуре FPGA Intel Arria 10 показали, что длительность вычисления дескрипторов HOG в слое ПВС не превышает 20 нс. Ближайшая по тематике работа [20], описывающая более классический подход к обработке изображения, показывает результат в 87 мкс.

Реализованная модель интегрирована в систему технического зрения (СТЗ) оценки дорожной ситуации БТС (рис. 6). Она обеспечивает объезд препятствий и реагирование БТС на присутствие пешеходов. СТЗ вместе с системой управления БТС функционирует под управлением Robot Operating System (ROS). Комбинированная схема соединения компонентов БТС представлена на рис. 7.

На рис. 8 отражена последовательность операций, осуществляемых системой технического зрения макета БТС при реализации быстродействующего алгоритма определения присутствия человека на кадре, полученном с камеры.



Рис. 5. Пространственное представление предобученного вектора весов, формируемого классификатором SVM.



Рис. 6. Макет беспилотной транспортной системы.

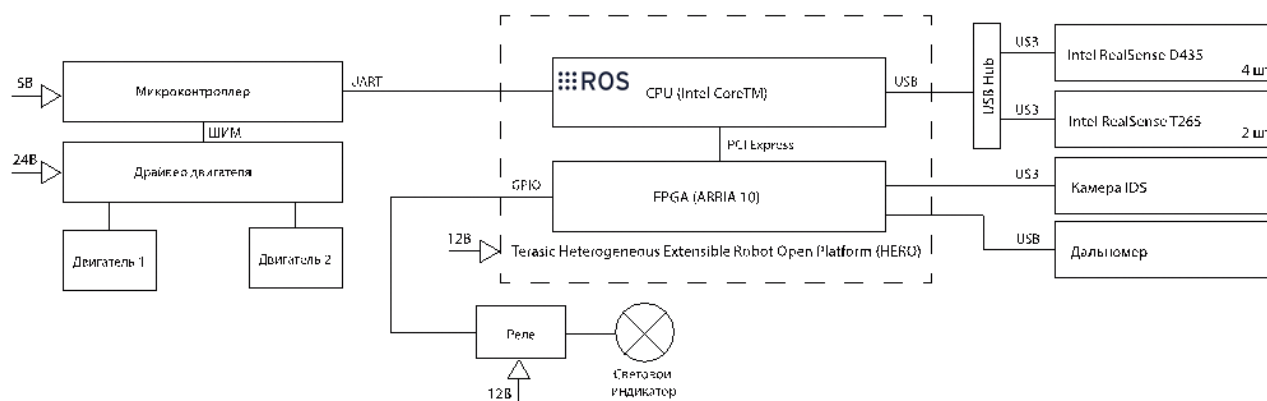


Рис. 7. Комбинированная схема комплекса.



Рис. 8. Принцип работы алгоритма распознавания человека на изображении.

Выводы

Работа открывает новое направление на стыке областей обработки изображений и вычислительных систем. Рассмотренная концепция заключается в том, что каждый пиксель изображения обрабатывается собственным специализированным вычислителем. Количество таких вычислителей сопоставимо с размером обрабатываемого изображения, а уровень описания закладываемых в них алгоритмов сопоставим с уровнем технологий, обеспечивающих взаимосвязь каждого активного чувствительного элемента (Active Pixel Sensor) видеосенсора с собственным ЭВ. Такой подход, с одной стороны, демонстрирует колоссальное увеличение производительности при обработке изображений, с другой стороны, увеличение требований к количеству транзисторов, расположенных на цельной кремневой пластине. Мировой опыт, описанный в работе, показы-

вает возможность реализации таких вычислителей, а предлагаемый подход в использовании однотипных ЭВ (принцип однородности) позволяет обходить неисправные участки на кремневой пластине, вызванные технологическим несовершенством.

Таким образом, просматривается возможность создания специализированных камер с низкими массогабаритными характеристиками, обеспечивающих скоростную обработку изображений. Интегрирование подобных камер на борт БТС позволит повысить автономность таких средств и снизить их массогабаритные характеристики.

Литература

1. K. Aizawa, K. Sakaue, Y. Suenaga
Image Processing Technologies. Algorithms, Sensors and Applications, USA, NY, New-York, Marcel Dekker, 2004, 282 pp.
2. D.G. Bailey
Design for Embedded Image Processing on FPGAs, SG, Singapore, John Wiley & Sons, 2011, 482 pp.
3. M. Fularz, M. Kraft, A. Schmidt, A. Kasiński
Int. J. Adv. Robot. Syst., 2015, 12, 1. DOI: 10.5772/61434.
4. P. Garcia-Risueno, P.E. Ibanez
Int. J. Mod. Phys., 2012, 23(7), 1230001.
DOI: 10.1142/S0129183112300011.
5. S. Saxena, N. Sharma, S. Sharma
Int. J. Adv. Res. Comp. Commun. Engin., 2013, 2(4), 1896.
DOI: 10.17148/IJARCCCE.2013.2420.
6. И.А. Каляев, И.И. Левин, Е.А. Семерников, В.И. Шмойлов
Реконфигурируемые мультимедийные вычислительные структуры, РФ, Ростов-на-Дону, ЮНЦ РАН, 2008, 393 с.
7. М.С. Кочерга, В.И. Шмойлов
Вестник ЮНЦ РАН, 2008, 4(2), 18.
8. А. Коновальчик
Современные технологии автоматизации, 2013, №3, 70.
9. Y. Jiang, S. Liu, H. Zhang, X. Kong
Optics Communications, 2014, 332, 359.
DOI: 10.1016/J.OPTCOM.2014.07.038.
10. C. Shi, J. Yang, Y. Han, Z. Cao, Q. Qin, L. Liu, N.-J. Wu, Z. Wang
B Proc. 2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014, 57, 128.
DOI: 10.1109/ISSCC.2014.6757367.
11. В.М. Хачумов, А.Ю. Попков, И.В. Соченков
В Сб. тез. докл. Третьего национального суперкомпьютерного форума (НСКФ-2014). РФ, Переславль-Залесский, ИПС имени А.К. Айламазяна РАН, 2014, с. 13–17.
12. N. Jouppi, C. Young, N. Patil, D. Patterson, G. Agrawal, R. Bajwa, S. Bates, S. Bhatia, N. Boden, A. Borchers, R. Boyle, P. Cantin, C. Chao, C. Clark, J. Coriell, M. Daley et al.
B 2017 ACM/IEEE 44th Annual International Symposium on Computer Architecture (ISCA), 2017, 1.
DOI: 10.1145/3079856.3080246.
13. S. Lie
IEEE Micro, 2023, 43(3), 18. DOI: 10.1109/MM.2023.3256384.
14. S. Chun, T. Kuo, H. Tsai, C. Liu, C. Wang, J. Hsieh, T. Lin, T. Ku, D.C. Yu
B 2020 IEEE 70th Electronic Components and Technology Conference (ECTC), USA, FL, Orlando, 2020, pp. 1–6.
DOI: 10.1109/ECTC32862.2020.00013.
15. N. Dey, G. Gosal, H. Khachane, W. Marshall, R. Pathria, M. Tom, J. Hestness
arXiv:2304.03208, 2023, 1. DOI: 10.48550/arXiv.2304.03208.
16. E. Talpes, D.D. Sarma, D.F. Williams, S. Arora, T. Kunjan, B. Floering, A. Jalote, C. Hsiong, C. Poorna, V. Samant, J. Sicilia, A.K. Nivarti, R. Ramachandran, T. Fischer, B. Herzberg, B. McGee et al.
IEEE Micro, 2023, 43(3), 31. DOI: 10.1109/MM.2023.3258906.
17. Д.В. Шашев, А.С. Бондарчук
Вестник ТГУ. Управление, вычислительная техника и информатика, 2023, 65, 116. DOI: 10.17223/19988605/65/12.
18. D.V. Shashev, A.A. Taganov, M. Mondal, M.V. Okunsky
J. Phys.: Conf. Ser., 2020, 1611, 012070.
DOI: 10.1088/1742-6596/1611/1/012070.
19. A.S. Bondarchuk, D.V. Shashev, S.V. Shidlovskiy
B Proc. Distributed Computer and Communication Networks: 24th International Conference, DCCN 2021 (RF, Moscow, 20–24 September, 2021), Ser. Communications in Computer and Information Science (CCIS), Eds V.M. Vishnevskiy, K.E. Samouylov and D.V. Kozyrev, FRG, Cham, Springer, 2022, 1552, pp. 408–419. DOI: 10.1007/978-3-030-97110-6_32.

20. N. Attarmoghaddam, K.F. Li
IEEE Transactions on Circuits and Systems II: Express Briefs,

2022, 69(4), 2306.
DOI: 10.1109/TCSII.2022.3148228.

English

Development and Research of Reconfigurable High-Speed Image Recognition Algorithms for Traffic Situation Assessment Based on Specialized Mobile Devices with Parallel-Pipeline Architecture*

Stanislav V. Shidlovskiy
Tomsk State University
36 Lenin Ave., Tomsk, 634050, Russia
shidlovskiysv@mail.ru

Dmitry V. Shashev
Tomsk State University
36 Lenin Ave., Tomsk, 634050, Russia
dshashev@mail.tsu.ru

Anton S. Bondarchuk
Tomsk State University
36 Lenin Ave., Tomsk, 634050, Russia
bondarchuk.a.c@gmail.com

Vladislav V. Shatravin
Tomsk State University
36 Lenin Ave., Tomsk, 634050, Russia
shatravin@stud.tsu.ru

Mihail V. Okunskiy
Tomsk State University
36 Lenin Ave., Tomsk, 634050, Russia
iamleftbrain@gmail.com

Abstract

The article summarizes the results of studying the use of reconfigurable computing environments for image processing tasks obtained on board unmanned transport systems. A new approach was proposed to the development of specialized, high-speed algorithms for recognizing images of a road scene, designed for hardware execution on computers with a parallel-pipeline architecture based on the model of reconfigurable computing environments.

The architecture and algorithmic tooling of such environments for traffic situation assessment problems are considered. A new class of image recognition algorithms based on the calculation of LBP, HOG descriptors and the SVM classifier has been formed. It is shown that calculations of HOG descriptors in the layer of a reconfigurable computing environment are performed in one machine cycle for the entire image, and the execution time, when implemented on an Intel Arria 10 FPGA, does not exceed 20 ns.

Keywords: high-performance computing systems, parallel computing, reconfigurable computing environments, driverless vehicles.

*The work was financially supported by RFBR (project 19-29-06078).

Images & Tables

Table 1. Encoding binary gradient direction values

φ	«No gradient»	180°	225°	270°
φ_1	0	0	1	1
φ_2	0	1	0	1



Fig. 1. Wafer-Scale Engine 2 M.

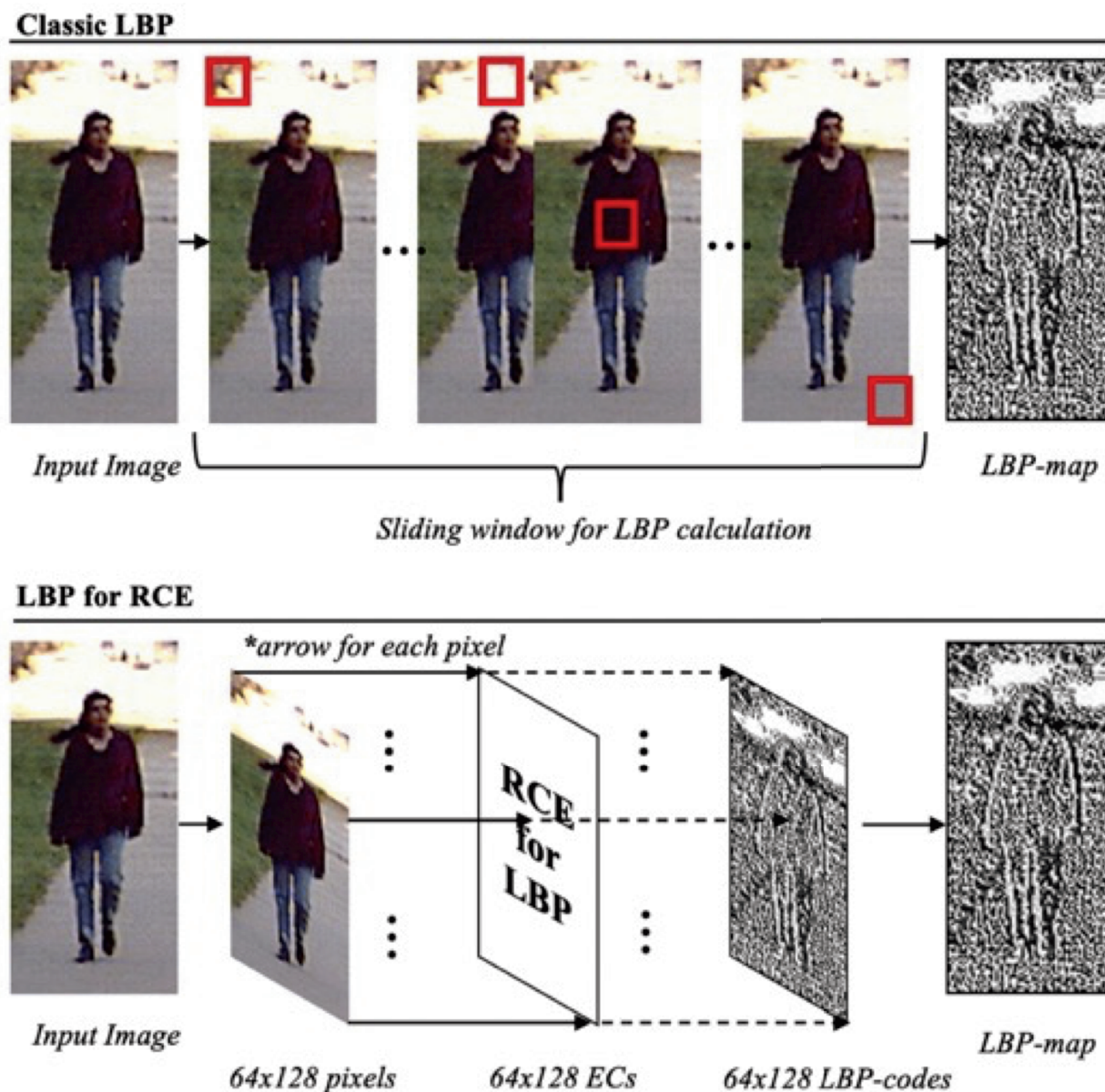


Fig. 2. Classic and proposed methods for finding the LBP descriptor.

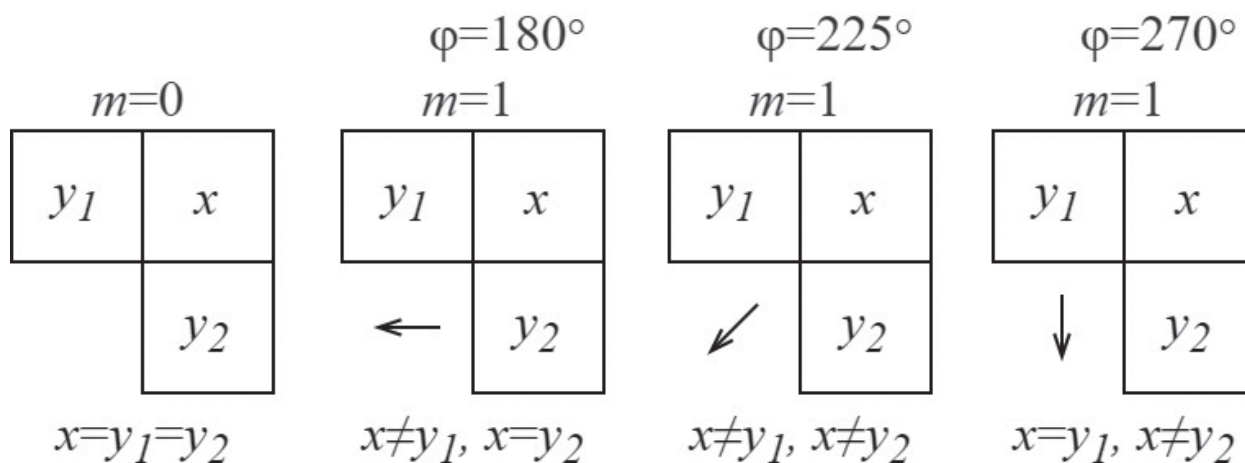


Fig. 3. Binary Gradient Value Options.



a



b

Fig. 4. Visualization of a binary gradient in an image: a – original binary image, b – visualization of a binary gradient in each pixel of the image.



Gradient weight values



Weight values for θ_1 direction



Weight values for θ_0 direction

Fig. 5. Spatial representation of the pre-trained vector of weights generated by the SVM classifier.



Fig. 6. Model of an unmanned transport system.

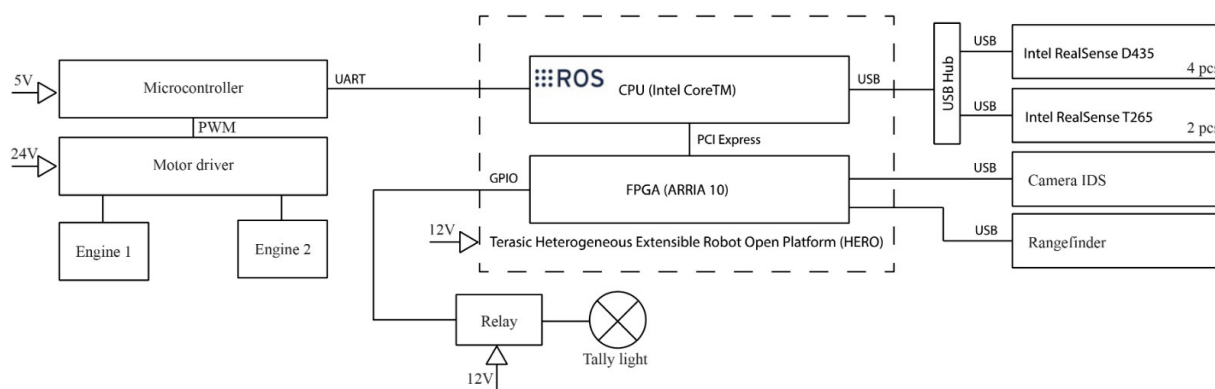


Fig. 7. Combined scheme of the complex.



Fig. 8. The principle of operation of the algorithm for recognizing a person in an image.

References



1. K. Aizawa, K. Sakaue, Y. Suenaga
Image Processing Technologies. Algorithms, Sensors and Applications, USA, NY, New-York, Marcel Dekker, 2004, 282 pp.
2. D.G. Bailey
Design for Embedded Image Processing on FPGAs, SG, Singapore, John Wiley & Sons, 2011, 482 pp.
3. M. Fularz, M. Kraft, A. Schmidt, A. Kasiński
Int. J. Adv. Robot. Syst., 2015, **12**, 1. DOI: 10.5772/61434.
4. P. Garcia-Risueno, P.E. Ibanez
Int. J. Mod. Phys., 2012, **23**(7), 1230001.
DOI: 10.1142/S0129183112300011.
5. S. Saxena, N. Sharma, S. Sharma
Int. J. Adv. Res. Comp. Commun. Engin., 2013, **2**(4), 1896.
DOI: 10.17148/IJARCCCE.2013.2420.
6. I.A. Kalyaev, I.I. Levin, E.A. Semernikov, V.I. Shmoilov
Reconfigurable Multi-Pipeline Computing Structures [Rekonfiguriruyemye multikonveyernye vychislitelnye struktury], RF, Rostov-on-Don, SSC RAS, 2008, 393 pp. (in Russian).
7. M.S. Kocherga, V.I. Shmoilov
Vestnik UNTs RAN [Bulletin of South Scientific Centre of RAS], 2008, **4**(2), 18. (in Russian).
8. A. Konovalchik
Modern Automation Technologies [Sovremennye tekhnologii avtomatizatsii], 2013, **3**, 70. (in Russian).
9. Y. Jiang, S. Liu, H. Zhang, X. Kong
Optics Communications, 2014, **332**, 359.
DOI: 10.1016/J.OPTCOM.2014.07.038.
10. C. Shi, J. Yang, Y. Han, Z. Cao, Q. Qin, L. Liu, N.-J. Wu, Z. Wang
In Proc. 2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014, **57**, 128.
DOI: 10.1109/ISSCC.2014.6757367.
11. V.M. Khachumov, A.Yu. Popkov, I.V. Sochenkov
In Proc. National Supercomputing Forum (NSCF-2020) [Sbornik tezisev dokladov Tretego Nacionalnogo Superkompyuternogo Forumu (NSKF-2014)]. RF, Pereslavl-Zalessky, Program Systems Institute of the RAS, 2014, pp. 13–17 (in Russian).
12. N. Jouppi, C. Young, N. Patil, D. Patterson, G. Agrawal, R. Bajwa, S. Bates, S. Bhatia, N. Boden, A. Borchers, R. Boyle, P. Cantin, C. Chao, C. Clark, J. Coriell, M. Daley et al.
In 2017 ACM/IEEE 44th Annual International Symposium on Computer Architecture (ISCA), 2017, 1.
DOI: 10.1145/3079856.3080246.
13. S. Lie
IEEE Micro, 2023, **43**(3), 18. DOI: 10.1109/MM.2023.3256384.
14. S. Chun, T. Kuo, H. Tsai, C. Liu, C. Wang, J. Hsieh, T. Lin, T. Ku, D.C. Yu
In 2020 IEEE 70th Electronic Components and Technology Conference (ECTC), USA, FL, Orlando, 2020, pp. 1–6.
DOI: 10.1109/ECTC32862.2020.00013.
15. N. Dey, G. Gosal, H. Khachane, W. Marshall, R. Pathria, M. Tom, J. Hestness
arXiv:2304.03208, 2023, 1. DOI: 10.48550/arXiv.2304.03208.
16. E. Talpes, D.D. Sarma, D.F. Williams, S. Arora, T. Kunjan, B. Floering, A. Jalote, C. Hsiong, C. Poorna, V. Samant, J. Sicilia, A.K. Nivarti, R. Ramachandran, T. Fischer, B. Herzberg, B. McGee et al.
IEEE Micro, 2023, **43**(3), 31. DOI: 10.1109/MM.2023.3258906.
17. D.V. Shashev, A.S. Bondarchuk
Tomsk State University Journal of Control and Computer Science [Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika], 2023, **65**, 116 (in Russian). DOI: 10.17223/19988605/65/12.
18. D.V. Shashev, A.A. Taganov, M. Mondal, M.V. Okunsky
J. Phys.: Conf. Ser., 2020, **1611**, 012070.
DOI: 10.1088/1742-6596/1611/1/012070.
19. A.S. Bondarchuk, D.V. Shashev, S.V. Shidlovskiy
In Proc. Distributed Computer and Communication Networks: 24th International Conference, DCCN 2021 (RF, Moscow, 20–24 September, 2021), Ser. Communications in Computer and Information Science (CCIS), Eds V.M. Vishnevskiy, K.E. Samouylov and D.V. Kozyrev, FRG, Cham, Springer, 2022, **1552**, pp. 408–419.
DOI: 10.1007/978-3-030-97110-6_32.
20. N. Attarmoghaddam, K.F. Li
IEEE Transactions on Circuits and Systems II: Express Briefs, 2022, **69**(4), 2306. DOI: 10.1109/TCSII.2022.3148228.