

МРНТИ 50.47.02
УДК 65.011.56

<https://doi.org/10.51889/7032.2022.73.24.028>

Ж.Е. Сейтбатталов^{1*}, С.К. Атанов¹, Х. Молдамурат¹

¹ Евразийский Национальный университет им. Л.Н. Гумилева, г. Нур-Султан, Казахстан
*e-mail: sbtl.jeks@gmail.com

ИНТЕЛЛЕКТУАЛЬНАЯ СИСТЕМА РАСПОЗНАВАНИЯ ГОСУДАРСТВЕННЫХ НОМЕРОВ АВТОМОБИЛЕЙ НА ОСНОВЕ ГРАНИЧНЫХ ВЫЧИСЛЕНИЙ

Аннотация

Существующая система мониторинга транспортного движения аккумулирует большое количество данных и применяет алгоритмы для обработки полученных фото и видеоизображений через камеры. Эти данные используются для регулирования транспортных потоков, распознавания государственных номеров автомобилей и фиксации дорожно-транспортных происшествий. Собранные данные традиционно отправляются в центр обработки данных (ЦОД) для проведения углубленного анализа и после отправляются оператору. Таким образом, большое количество поступающих данных в ЦОД и центр мониторинга транспортного движения понижают пропускную способность сети, оказывая негативное воздействие на скорость принятия решения оператором. Для решения проблемы предлагается использовать парадигму граничных вычислений посредством размещения вычислительных узлов в близости от камер и проведения предобработки данных на граничных устройствах. В качестве граничного устройства, в статье рассматривается Raspberry Pi, а также алгоритмы компьютерного зрения для предварительной обработки данных на граничных узлах и процесс передачи данных в ЦОД с последующим оперативным принятием решения оператором. Полагаем, что применение парадигмы граничных вычислений будет способствовать оптимизации трафика сети и разгрузке вычислительных мощностей ЦОД.

Ключевые слова: интернет вещи, граничные вычисления, видео аналитика, цифровая обработка изображений, Raspberry pi, сетевые технологий.

Аңдатпа

Ж.Е. Сейтбатталов¹, С.К. Атанов¹, Х. Молдамурат¹

¹ Л.Н. Гумилёв атындағы Еуразия ұлттық университеті, Нұр-Сұлтан қ., Қазақстан

ШЕКАРАЛЫҚ ЕСЕПТЕР НЕГІЗІНДЕГІ АВТОМОБИЛЬДІҢ МЕМЛЕКЕТТІК НӨМЕРІН ТАНУДЫҢ ЗИЯЛДЫ ЖҮЙЕСІ

Қазіргі трафикті бақылау жүйесі деректердің үлкен көлемін жинақтайды және камералар арқылы алынған фото және бейне кескіндерді өңдеу алгоритмдерін қолданады. Бұл деректер көлік ағындарын реттеу, нөмірлерді тану және жол-көлік оқиғаларын тіркеу үшін пайдаланылады. Жиналған деректер дәстүрлі түрде өңдеу, терең талдау үшін деректер орталықтарына жіберіледі, содан кейін операторға жіберіледі. Осылайша, дата орталықтары мен трафикті бақылау орталығына түсетін деректердің үлкен көлемі желі өткізу қабілетін төмендетеді және оператордың шешім қабылдау жылдамдығына кері әсерін тигізеді. Бұл мәселені шешу үшін есептеуіш түйіндерді камералардың тікелей маңайында орналастыру және шеткі құрылғыларда деректерді алдын ала өңдеуді жүргізу арқылы шеттік есептеу парадигмасын пайдалану ұсынылады. Шеттік құрылғы ретінде мақалада Raspberry Pi, сондай-ақ шеткі түйіндердегі деректерді алдын ала өңдеуге арналған компьютерлік көру алгоритмдері және оператордың кейіннен операциялық шешімдер қабылдауымен деректерді деректер орталықтарына беру процесі қарастырылады. Біз шеттік есептеу парадигмасын пайдалану желілік трафикті оңтайландыруға және деректер орталығының есептеу қуатын түсіруге көмектеседі деп сенеміз.

Түйін сөздер: заттар интернеті, шеттік есептеулер, бейне аналитика, сандық бейнелеу, Raspberry pi, желілік технологиялар.

Abstract

INTELLIGENT RECOGNITION SYSTEM OF NUMBER PLATE BASED ON EDGE COMPUTING

Z.E. Seitbattalov¹, S.K. Atanov¹, Kh. Moldamurat¹

¹ L.N. Gumilyov Eurasian National University, Nur-Sultan, Kazakhstan

The present traffic monitoring system gathers a lot of data and uses algorithms to interpret images captured by cameras. These information are used to control traffic flow, identify license plates, and keep track of traffic accidents. Traditionally, the collected data are routed to data centers for processing and analysis before being delivered back to the

operator. As a result, a lot of data going into the data center and traffic monitoring center limits network capacity and slows down the operator's ability to make decisions quickly. We suggested using the edge computing paradigm to address this issue by positioning edge nodes close to the cameras and doing data preprocessing on edge devices. As an edge device, the article considers the Raspberry Pi, as well as computer vision algorithms for pre-processing data at edge nodes and the process of transferring data to data centers with subsequent operational decision making by the operator. We believe that the use of the edge computing paradigm will help optimize network traffic and offload the computing power of the data center.

Keywords: Internet of Things, edge computing, video analytic, Raspberry Pi, image processing, network technologies.

Введение

По мере увеличения количества Internet of Things (IoT) устройств в мире было сгенерировано огромное количество данных за последнее десятилетие [1]. Тенденция роста количества IoT устройств будет сохраняться и дальше. В соответствии с прогнозом от Statista к 2030 году количество IoT устройств достигнет более 25.4 миллиарда [2]. В связи с этим вырастет объем, генерируемых и передаваемых данных по интернет сети, что формирует предпосылку в необходимости быстрой обработки данных для систем чувствительных к временным задержкам. Из всех имеющихся типов данных, передаваемых по интернет сети, наиболее емкими по объему являются видео и фотоизображения.

Одной из подобных систем чувствительных к задержкам сети является система мониторинга транспортного движения. Подобные системы занимаются следующими трудоемкими задачами: расчетом скорости движения транспортного средства, фиксирование государственных номеров автомобилей и нарушений правил дорожного движения (ПДД), обнаружение дорожно-транспортных происшествий (ДТП) и плотного автомобильного движения, регулирование транспортного потока, принятие решения о вызове пожарной бригады и скорой помощи при ДТП [3-5]. Также надо отметить, что из перечисленных функции системы, процесс распознавания государственного номера средства передвижения является наиболее простым на первый взгляд, но все же может иметь длительный процесс распознавания и передачи данных по сети.

Целью этой статьи является реализация процесса распознавания государственного номера автомобиля на основе компьютерного зрения и способов повышения скорости передачи полученных данных в ЦОД и Центр мониторинга транспортного движения (ЦМТД) без искажения на базе граничных вычислений. На вычислительных узлах, размещённых в непосредственной близости от камер, проводится обработка изображений компьютерным зрением и граничными вычислениями, которые выделяют важную информацию из полученных видео и фотоизображений. Затем переводят их в удобный текстовый формат. Такой формат данных исключает возможность искажения информации в виду ее простоты и способствует высокой скорости передаче данных в сети.

В рассмотренных работах в качестве основных этапов распознавания государственного номера автомобиля выделяют следующие: локализация автомобильного номера (Number Plate Localization), сегментация символов (Character Segmentation) и оптическое распознавание символов (Optical Character Recognition) [4]. На первом этапе при неблагоприятных погодных условиях или слабой освещенности используются техники бинаризации изображения (binarization) или оттенки серого цвета (grayscale). После этого осуществляется локализация автомобильного номера и его сегментация с помощью Connected Component Analysis (CCA) и анализа соотношения (ratio analysis). И наконец, идет распознавание символов через Support Vector Classifier (линейный, полигональный, rbf), k Nearest Neighbor (KNN), Extra Tree Classifier, Logistic Regression (LR), Random forest (RF), Support Vector Classifier (SVC) и KNN [5]. Несмотря на высокую скорость обработки данных, реализация подобных систем на базе программируемых логических интегральных схемах (ПЛИС) обходится дорого.

В качестве альтернативного технического решения мы предлагаем использовать одноплатный компьютер Raspberry Pi, что снизит расходы на разработку системы и позволит легко отслеживать пакет отправляемых данных по сети. Для решения проблемы длительной передачи данных предлагается применить парадигму граничных вычислений. Ряд статей показал успешное осуществление перехода от традиционной облачной архитектуры к парадигме граничных вычислений с итоговым результатом повышения скорости передачи данных по сети [3, 6, 7, 8, 9]. Этот подход нашел свой успех применения в Smart House, Smart Traffic Monitoring System, Smart City, распознавании эмоций человека, мониторинга показателей состояния пациента, то есть в тех системах, где критически

важна скорость обработки и передачи данных. Однако стоит отметить, что при выполнении трудоемких задач и ситуации, когда интернет соединение стабильно, целесообразно использовать гибридную парадигму, позволяющую применять как облачные, так и граничные вычисления посредством выбора оптимальной сети через интеллектуальную систему разгрузки задач на базе нечеткой логики [10, 11]. Одноплатный компьютер Raspberry Pi легко может применяться в обеих парадигмах. Одним из преимуществ Raspberry Pi является наличие процессора средней скорости и обеспечение поддержки периферийного интерфейса и сети, что позволяет осуществлять сбор и анализ данных с IoT устройств [12]. Что касается протокольного решения на Raspberry Pi, существует протокол связи Message Queue Telemetry Transport (MQTT). Он считается одним из наиболее применяемых в IoT индустрии из-за низкого уровня энергопотребления [13].

Материалы и методы исследования

Как было отмечено в предыдущем разделе, в качестве граничного устройства будет использоваться одноплатный компьютер Raspberry Pi 4 Model B (4GB) с камерой OmniVision OV5647 для распознавания государственных номеров транспорта на базе алгоритмов компьютерного зрения (рисунок 1).

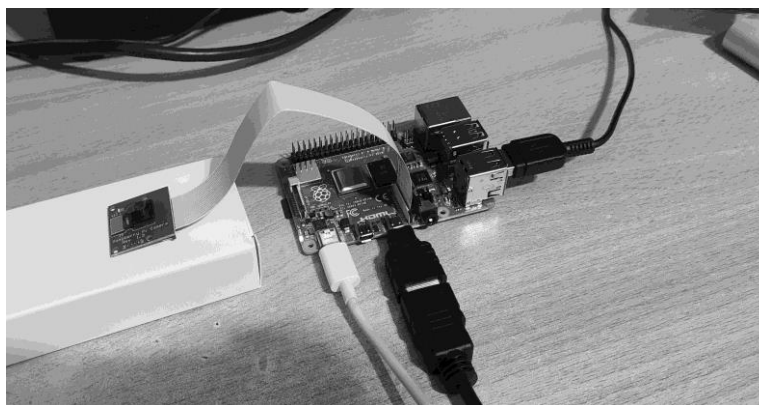


Рисунок 1. Граничный узел (Raspberry Pi и камера OmniVision OV5647)

Для организации сообщения между издателем (Raspberry Pi) и подписчиками (пользователями ЦМТД) будет использован протокол Message Queue Telemetry Transport (MQTT) на базе Node Red и MQTTBox. Схема организации взаимодействия по протоколу MQTT представлена на рисунке 2.

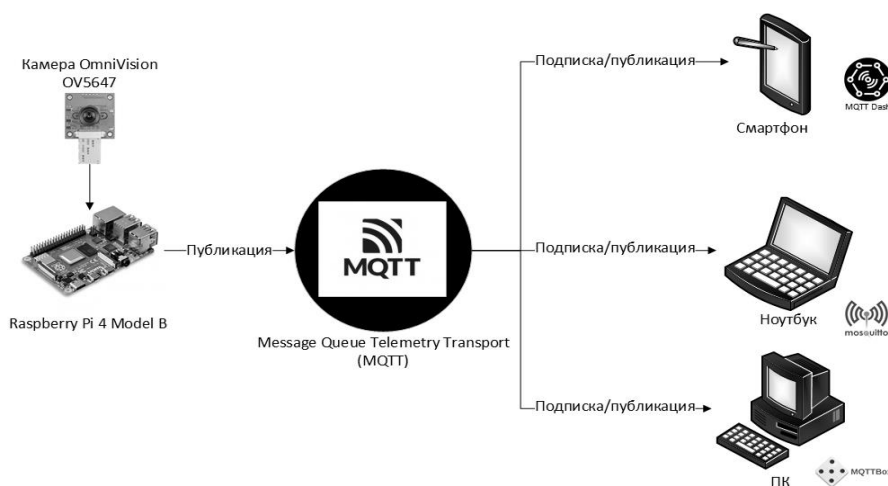


Рисунок 2. Схема взаимодействия MQTT брокера сообщений

Устанавливаем Node Red на одноплатный компьютер Raspberry Pi для организации потока данных и визуального программирования IoT устройства, а также проводим конфигурацию брокера сообщений MQTTBox на стороне дата-центра. Рекомендуемая операционная система для Raspberry Pi

– Raspbian OS, для сервера – любая совместимая операционная система с брокером сообщений на базе ядра Linux или Windows. В нашем случае была выбрана операционная система Windows с брокером сообщений MQTTBox. Схема IoT сети представлена на рисунке 3. Камера OmniVision OV 5647 передает фото и видеоизображения на Raspberry Pi. Одноплатный компьютер посредством алгоритмов компьютерного зрения определяет государственный номер автомобиля и передает его на сторону сервера в текстовом формате через MQTT брокер сообщений на базе Red Note. После оператор ЦМТД получает обработанные данные через MQTTBox и принимает решение о внесении государственного номера транспорта в базу данных номеров, например, при нарушении ПДД.

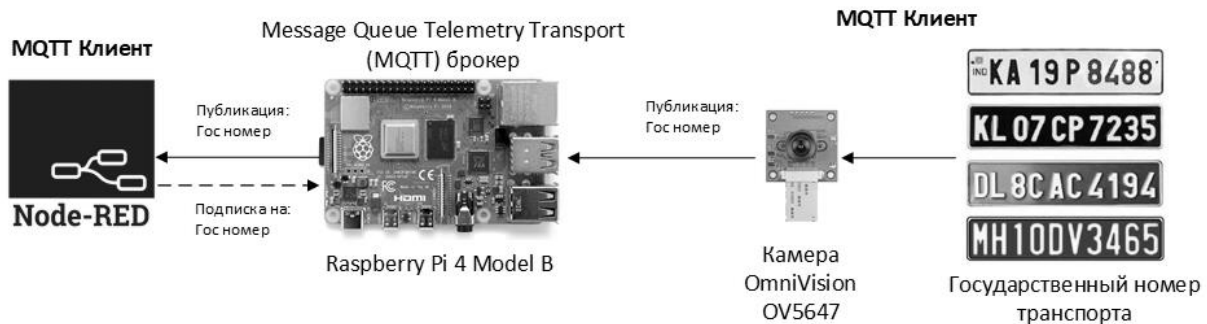


Рисунок 3. Схема IoT сети

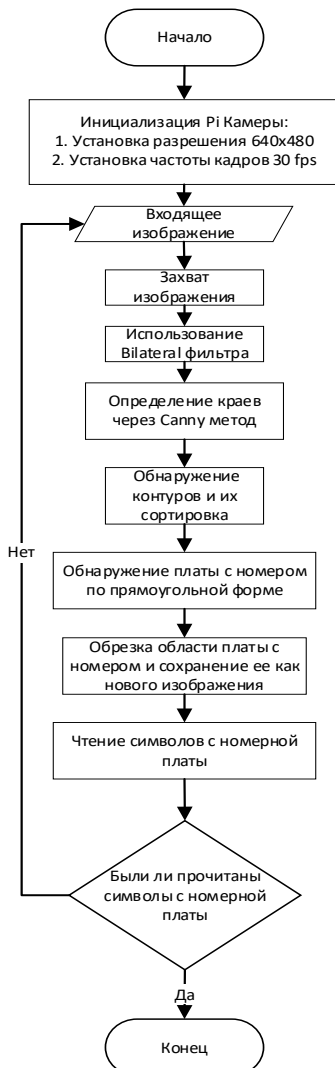


Рисунок 4. Блок-схема алгоритма компьютерного зрения

Для удобства процесса разработки алгоритма компьютерного зрения рекомендуется настроить и развернуть VNC Viewer на стороне разработчика, а также подключить протоколы SSH, VNC на Raspberry Pi.

Для написания алгоритма на языке программирования Python были использованы редактор Nano и библиотеки OpenCV, Tesseract, PYTTSX3, Imutils, PIL import Image, picamera и Tesseract OCR с целью локализации автомобильного номера, сегментации символов и оптического распознавания символов [14].

На рисунке 4 представлена блок схема алгоритма компьютерного зрения для распознавания государственного номера транспорта.

Для начала была выполнена инициализация камеры, установка ее разрешения (640x480) и частоты на 30 кадров в секунду. После была использована функция capture_continuous, чтобы начать захват кадров с камеры Raspberry Pi. После захвата кадра была применена функция двустороннего фильтра для удаления нежелательных деталей из захваченного изображения.

После удаления ненужных деталей был использован метод Canny Edge, чтобы выполнить обнаружение граней табло с государственным номером.

Далее был выполнен поиск контуров на полученных изображениях и сортировка обнаруженных контуров от большего к меньшему. Так как Raspberry Pi может найти несколько контуров, в связи с этим нам нужно отфильтровать контур номерного знака, выполнив поиск контура в виде прямоугольной формы. После этого была выполнена обрезка всех областей, кроме номерного знака.

Далее было сохранено полученное изображение как новое.

На последнем этапе было выполнено считывание символов с изображения номерного знака с помощью библиотеки Tesseract и сохранены распознанные символы в переменную. На рисунке 5 приведен пример вывода государственного номера в текстовом варианте (вывод в терминале) и в виде обрезанного изображения.



Рисунок 5. Процесс опознавания государственного номера транспорта с помощью компьютерного зрения (Computer Vision)

Для написания алгоритма опознавания автомобильных номеров и организации потока сообщений из меню узла Node Red были использованы блоки aedes broker, inject, exec, function, mqtt out, mqtt in и debug. Схема потока представлена на рисунке 6.

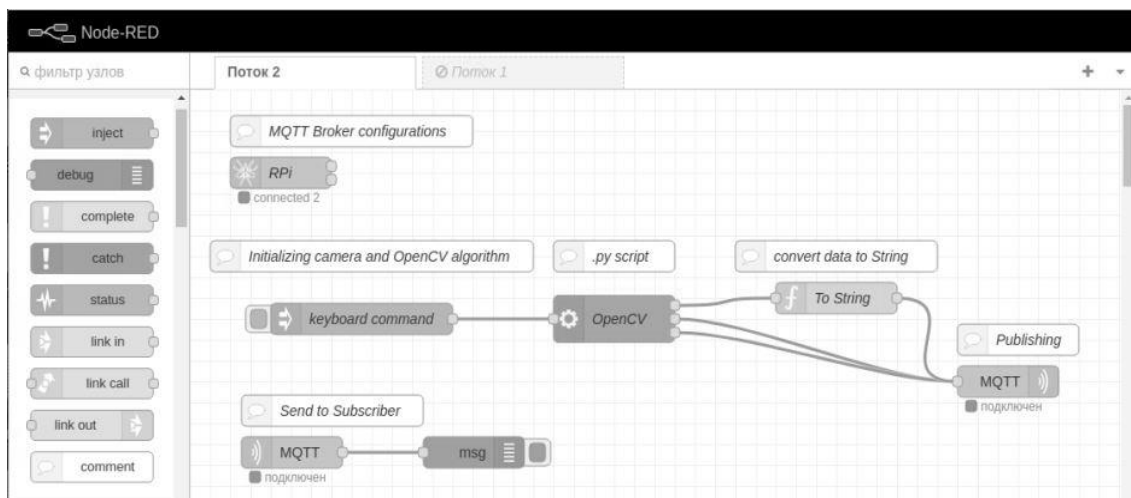


Рисунок 6. Flow в Node Red

Узел aedes broker (RPi) позволяет выполнить настройки брокера сообщений для Raspberry Pi и указать его порт. Узел Inject (keyboard command) инициализирует работу потока, камеры и запускает алгоритм блока OpenCV (компьютерного зрения), реализованный на языке программирования Python для локализации государственного номера, сегментации символов и оптического распознавания символов. Полученные результаты отправляются в узел function (To string) для перевода считанного текста в читаемый вид для человека. Далее узлы mqtt out и mqtt in (MQTT) осуществляют публикацию данных на адрес подписчиков (пользователи ЦМТД). Узел msg позволяет вывести результат.

Результаты исследования

На рисунке 7 приведены результаты обработки фотоизображений в окнах Stopped и терминала в виде текстового формата. Снятое изображение камерой OmniVision OV 5647 представлено в окне Frame.



Рисунок 7. Результат обработки фотоизображений с государственным номером транспорта

По результатам в виде текстового формата можно отметить некоторую погрешность в идентификации государственных номеров, что связано с небольшой базой данных для обучения компьютерного зрения.

В консоли Node Red (правая часть окна), во вкладке «Отладка» на рисунке 8 представлены результаты распознавания автомобильных номеров через Node Red. Результат передачи данных оператору и время исполнения алгоритма компьютерного зрения (после текста run time) указаны в окне брокера сообщений MQTTBox. Среднее время на исполнения алгоритма составило 4 секунды. У представленных результатов также имеется погрешность в распознавании автомобильных номеров.



Рисунок 8. Результат обработки фотоизображений в Node Red

Графическая диаграмма сетевого трафика при передаче данных в брокер сообщений представлена на рисунке 9. Максимальное значение сетевого трафика составило 4.78 килобит/секунду.

Таким образом, можно отметить, что процесс передачи данных по парадигме граничных вычислений способствует оптимизации сетевого трафика, так как при облачной парадигме объем передаваемых фото и видеоданных измеряются в Мегабит/секунду и Гигабит/секунду.

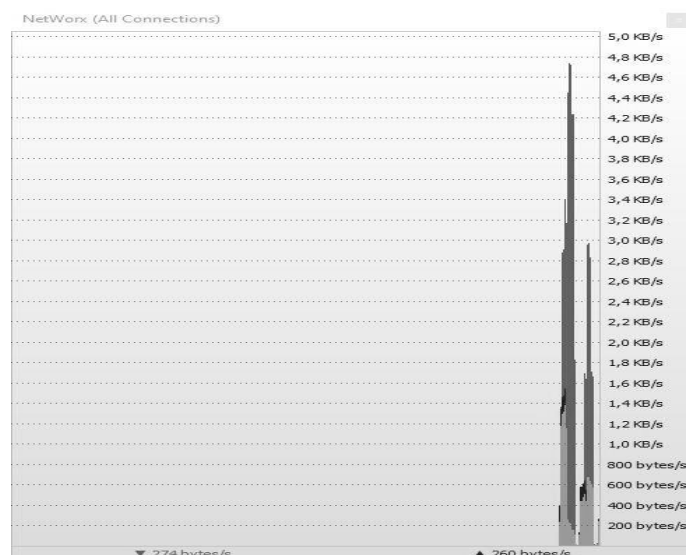


Рисунок 9. Графическая диаграмма сетевого трафика при передаче данных в брокер сообщений

В таблице 1 представлены различные методы обработки изображений, платформы, разрешение и время выполнения в миллисекундах [4, 15].

Таблица 1. Сравнение по времени выполнения различных алгоритмов обработки изображений в центральном процессорном устройстве.

Метод обработки изображения	Платформа	Разрешение	Время, мс
High Definition Number Plate Localization	Zynq-7000 SoC	960×720 (HD)	16.17
Standard Definition Number Plate Localization	FPGA Virtex-4	640×480 (SD)	6.7
Oily style NPR	CPU: Q9450 with 42.56 GFLOPS GPU: NVIDIA G92 (GeForce 9800 GTX) with 128 Cores and 512 MB Video Memory, GTX 280 for Speed UP Test	512×512	7172
Corners and edge detectio	OS: Ubuntu 11.04 CPU: Dual Core 6600, 2.40 GHz, Mem: 2 GB GPU: GeForce GTX 280, 240 CUDA cores, Memory: 1GB GPU: Tesla C1060, 240 CUDA cores, Memory: 4GB	2048×2048	4006
Content authentication	CPU Intel Xeon 5520 (2.26GHz) RAM 12GB DDR3 (1333MHz) GPU Architecture Tesla C1060 OS Centos 5.3 (64 bit) V2.3 CUDA	1024×1024	28877.66
Linear feature extraction	CPU: Q9450 with 42.56 GFLOPS GPU: NVIDIA G92 (GeForce 9800 GTX) with 128 Cores and 512 MB Video Memory, GTX 280 for Speed UP Test	1800×1400	500.958685
Inverse sinusoidal contrast transformation	AMD Phenom II Quad-core to 3.2 GHz, 12 GB of RAM, Operating System: 64-bit Linux Fedora 14 GPU: GeForce 430 GT video card with 96 cores and 1 GB of RAM DDR3 is used	1024×1024	151.256079

Таким образом, скорость выполнения различных методов обработки изображений на ПЛИС в действительности выше, чем на Raspberry Pi. Однако стоит отметить, что скорость исполнения алгоритма на одноплатном компьютере Raspberry Pi незначительно уступает вычислительным мощностям персонального компьютера при обработке фотоизображений центральным процессорным устройством.

Заклучение

В данной работе мы предложили применение парадигмы граничных вычислений на базе одноплатного компьютера Raspberry Pi 4 Model B с камерой OmniVision OV 5647 для проведения предварительной обработки данных, что способствовало оптимизации трафика сети и разгрузке вычислительных мощностей центра обработки данных. Примененный метод граничных вычислений с Raspberry Pi и передачей данных в текстовом формате также способствовал снижению затрат на техническое решение в отличие от программируемых логических интегральных схем и персональных компьютеров. Кроме того, стоит отметить, что предложенный метод позволяет увеличить количество подключаемых IoT устройств к сети из-за минимального воздействия на сетевой трафик, а также позволяет сохранить конфиденциальность большей части данных. Таким образом, с целью повышения скорости и качества распознавания государственных транспортных номеров с помощью алгоритма компьютерного зрения предлагается увеличить базу данных для обучения. Помимо этого, также добавить функциональную возможность переключения режима работы граничного устройства между облачными и граничными парадигмами. Однако, также учитывая вышеизложенную проблему, представляется, что перспективным может являться направление создание гибридных методов для распознавания государственных номеров, использующих преимущества и нивелирующих недостатки имеющихся на сегодня различных подходов.

References:

- 15 Marjani, M., F. Nasaruddin, A. Gani, A. Karim, I. A. T. Hashem, A. Siddiq, and I. Yaqoob. "Big Iot Data Analytics: Architecture, Opportunities, and Open Research Challenges." *Article. Ieee Access* 5 (2017): 5247-61. <https://doi.org/10.1109/access.2017.2689040>.
- 16 Sulich, A., M. Rutkowska, A. Krawczyk-Jeziarska, J. Jezierski, and T. Zema. "Cybersecurity and Sustainable Development." *Proceedings Paper. Knowledge-Based and Intelligent Information & Engineering Systems (Kse 2021)* 192 (2021): 20-28. <https://doi.org/10.1016/j.procs.2021.08.003>.
- 17 Liu, G. X., H. Shi, A. Kiani, A. Khreishah, J. Lee, N. Ansari, C. J. Liu, and M. M. Yousef. "Smart Traffic Monitoring System Using Computer Vision and Edge Computing." *Article; Early Access. Ieee Transactions on Intelligent Transportation Systems* (2021): 12. <https://doi.org/10.1109/tits.2021.3109481>.
- 18 Al-Zawqari A., Hommos O., Al-Qahtani A., Farhat A. A. H., Bensaali F., Zhai X. J., and Amira A. "HD Number Plate Localization and Character Segmentation on the Zynq Heterogeneous Soc." *Article. Journal of Real-Time Image Processing* 16, no. 6 (Dec 2019): 2351-65. <https://doi.org/10.1007/s11554-017-0747-7>.
- 19 Babbar, S., S. Kesarwani, N. Dewan, K. Shangle, and S. Patel. "A New Approach for Vehicle Number Plate Detection." *Paper presented at the 11th International Conference on Contemporary Computing (IC3), Jaypee Inst Informat Technol, Noida, INDIA, Aug 02-04 2018*.
- 20 Nikouei, S. Y., R. H. Xu, and Y. Chen. "Smart Surveillance Video Stream Processing at the Edge for Real-Time Human Objects Tracking." *Fog and Edge Computing: Principles and Paradigms* (2019): 319-46.
- 21 Yang, J. N., T. T. Qian, F. Zhang, and S. U. Khan. "Real-Time Facial Expression Recognition Based on Edge Computing." *Article. Ieee Access* 9 (2021): 76178-90. <https://doi.org/10.1109/access.2021.3082641>.
- 22 Yar, H., A. S. Imran, Z. A. Khan, M. Sajjad, and Z. Kastrati. "Towards Smart Home Automation Using Iot-Enabled Edge-Computing Paradigm." *Article. Sensors* 21, no. 14 (Jul 2021): 23 4932. <https://doi.org/10.3390/s21144932>.
- 23 Brimzhanova, S. S., Atanov, S. K., Moldamurat, K., Koblekov, K. S., and L. G. Gagarina. "Cross-Platform Compilation of Programming Language Golang for Raspberry Pi." *Proceedings of the 5th International Conference on Engineering and MIS, Astana, Kazakhstan, Association for Computing Machinery, 2019*.
- 24 Atanov, S. K. and Seitbattalov Z. Y. and Moldabayeva Z. S. "Development an Intelligent Task Offloading System for Edge-Cloud Computing Paradigm." *Proceedings - 2021 16th International Conference on Electronics Computer and Computation (ICECCO)* (2021).
- 25 Seitbattalov, Z. Y. and Atanov, S. K. and Moldabayeva, Z. S. "An Intelligent Decision Support System for Aircraft Landing Based on the Runway Surface." *Proceedings - 2021 IEEE International Conference on Smart Information Systems and Technologies (IEEE SIST)* (2021).
- 26 Mahmud, R., and A. N. Toosi. "Con-Pi: A Distributed Container-Based Edge and Fog Computing Framework." *Ieee Internet of Things Journal* 9, no. 6 (Mar 2022): 4125-38. <https://doi.org/10.1109/jiot.2021.3103053>.
- 27 Lima, D. B. C., R. Lima, D. D. Medeiros, R. I. S. Pereira, C. P. de Souza, and O. Baiocchi. "A Performance Evaluation of Raspberry Pi Zero W Based Gateway Running Mqtt Broker for Iot." *2019 Ieee 10th Annual Information Technology, Electronics and Mobile Communication Conference (Iemcon)* (2019): 76-81.
- 28 Arslan, Y., and H. Canbolat. "Performance of Deep Neural Networks in Audio Surveillance." *Proceedings - 2018 6th International Conference on Control Engineering & Information Technology (Ceit)* (2018): 5.
- 29 S. Saxena, S. Sharma, and Sharma N. "Parallel Image Processing Techniques, Benefits and Limitations." *Research Journal of Applied Sciences, Engineering and Technology* 12 (01 2016): 223-38. <https://doi.org/10.19026/rjaset.12.2324>.