

АПАРАТНО-ПРОГРАМНА РЕАЛІЗАЦІЯ ІНТЕЛЕКТУАЛЬНОЇ ВІДЕОКАМЕРИ

¹Інститут кібернетики ім. В. М. Глушкова НАН України, Київ

На сьогодні відеокамери є розповсюдженими приладами, які використовуються для широкого спектра задач реального часу — від використання у оптичних маніпуляторах до ідентифікації людини. Однак, наразі не існує універсальних відеокамер, які могли б ефективно використовуватися у всьому спектрі задач. Це пов'язано з великою кількістю параметрів відеопотоку (висота та ширина, якість, контраст кадру) та вимогам як до апаратного (швидкодія, стійкість до зовнішніх впливів) так і до програмного (кодування відеопотоку, підтримка протоколів даних) забезпечення таких камер. У роботі проаналізовано модульний підхід до створення інтелектуальної відеокамери з певних модулів. За результатами проведеного аналізу існуючих інтелектуальних відеокамер, їх переваг та недоліків запропоновано класифікувати реалізації відеокамер на три випадки. Запропоновано аналізувати поведінку інтелектуальної відеокамери як пристрій, який складається з декількох частин — «Сенсор» (пристрій, що передає обробнику даних неспотворену відеоінформацію), «Обробник даних» (будь-який пристрій, який виконує обробку відеоінформації для її аналізу та інтерпретації) та «Транслятор результату» (пристрій, який передає результат для подальшої реакції), що дозволить швидко створювати спеціалізовані камери на базі існуючої. Сформульовані вимоги до апаратної складової варіантів практичної реалізації відеокамери та розрахована гранична швидкість передачі інформації від «Сенсора» до «Обробника даних». Запропонована програмна архітектура, яка необхідна для функціонування камери. Створені UML-діаграми класів для ключових програмних модулів, які описують приймання даних від «Сенсора» до «Обробника даних», обробку даних у «Обробнику» та передачу інформації для подальшого аналізу від «Обробника даних» до «Транслятора результату».

Ключові слова: інтелектуальна відеокамера, системи реального часу, програмна архітектура.

Вступ

На сьогодні відеокамери є розповсюдженими приладами, які використовуються для широкого спектра задач реального часу, однак, не існує універсальних відеокамер, які могли б ефективно використовуватися всюди. Це пов'язано з великою кількістю параметрів відеопотоку та вимогам апаратного і програмного забезпечення камер. Використання великої кількості камер приводить до проблем передачі та обробки великої кількості відеоінформації, якою займаються складні відеосистеми реального часу (наприклад, у [1]). Головною складовою таких систем є спеціалізовані інтелектуальні відеокамери, які можуть в той чи інший спосіб зменшувати обсяг переданої інформації за рахунок аналізу відеопотоку. Однак зазвичай для зменшення об'єму відеокамера використовує лише перетворення «чистого» відеопотоку у відеопоток зі спотвореннями (так зване «кодування» відеопослідовності). Це може стати недоліком у деяких випадках (наприклад, для виділення та супроводження автомобіля, що рухається, дроном з відеокамерою) та призведе до неприйняттого результату. У цьому випадку, для «кодування» відеоінформації, можливе використання спеціалізованих алгоритмів для аналізу відеоданих, виділення тільки необхідної інформації і передачі її для подальшої обробки, як це запропоновано у [2]. Разом з тим, використання типових реалізацій для обробки отриманої відеоінформації значно спростить розробку нових камер. Для ще більшої швидкодії, після виділення на сенсорі камери, можливе подальше кодування відеопотоку. Перспективним наразі виглядає розробка спеціалізованих відеокамер модульного типу, які б дозволяли швидко модифікацію камери для пристосування її під задачу з конкретними вимогами до неї (наприклад, мале живлення камери, висока роздільна здатність камери, висока швидкодія тощо).

Метою роботи є розробка підходів до побудови модульної інтелектуальної відеокамери, яку можна швидко налаштувати для конкретних задач.

Існуючі основні варіанти реалізації інтелектуальної відеокамери

Сенсор та мікроконтролер (або DSP)

Переваги такого варіанта — низьке споживання енергії та низька ціна готового приладу. Дійсно, якщо є необхідність створити прилад масового використання, який би використовувався в складних фізичних умовах, реагував би на декілька типів випадків (наприклад, сигнал про заваду на шляху, наближення об'єкта та ін.) та був недорогим — такий варіант буде найкращим вибором. Недоліком такого варіанта є порівняно великий час розробки, оскільки дешеві мікроконтролери часто мають невелику кількість оперативної пам'яті та пам'яті програм і потребують спеціалізованого програмного забезпечення. Такий варіант описаний у [3].

Сенсор (камера) та мікропроцесор

На відміну від попереднього варіанта потребує більше енергії, дорожчий. Однак такий варіант використовує операційну систему реального часу (наприклад, FreeRTOS, Linux-подібну та ін.), що дозволяє швидко компонувати камеру з різних функціональних блоків. Зазвичай процесор має декілька центральних процесорних блоків (CPU або «ядер»), що дозволяє одночасно приймати, обробляти та передавати дані для подальшої реакції, причому велика кількість оперативної пам'яті дозволяє використовувати складну алгоритмічну обробку отриманих даних. Прикладом такої камери може бути сенсор та одноплатний мікрокомп'ютер (наприклад, Raspberry Pi). Подібна конфігурація описана у [4].

Сенсор (камера) та ПЛІС

Використання програмованої логічної інтегральної схеми дозволяє приймати дані з сенсора камери, обробляти їх на спеціалізованих апаратних засобах і передавати дані для подальшої реакції на швидкостях, які дозволяють на порядок підвищити отримання результату. Також перевагою є створення тільки необхідних апаратних засобів, що значно зменшить енергоспоживання цього варіанта. Однак для створення необхідної конфігурації ПЛІС потрібно багато часу у порівнянні з попередніми варіантами. Для пришвидшення розробки провідні компанії-виробники ПЛІС (Altera та Xilinx) випускають спеціалізовані апаратно-програмні набори [5], [6]. Подібні набори випускає також компанія NVideo [7].

Вищезазначені варіанти не виключають одне одного — існують проекти, які дозволяють використовувати так звані «Linux hard-coded cores» у ПЛІС, що дозволяє значно зменшити час швидкості розробки прототипу камери [8], або використання сенсора, DSP та ПЛІС [9].

Для подальшого розгляду вибрано варіант «Сенсор та мікропроцесор», оскільки використання готових мікрокомп'ютерів типу Raspberry Pi або Orange Pi з операційною системою Linux та готового сенсора значно зменшує час створення експериментального зразка.

Узагальнена модель інтелектуальної відеокамери

Наразі у світі широко використовуються інтелектуальні відеокамери (ІВ) для різних типів задач. Однак, не дивлячись на широке поширення ІВ все ще залишаються проблеми обробки, передачі та інтерпретації результату, які не дозволяють створити універсальну відеокамеру. Існує певна плутанина з терміном «інтелектуальної відеокамери» (описано у [10], [11]) під інтелектуальною відеокамерою у цій статті будемо розуміти апаратно-програмний комплекс, який дозволяє приймати інформацію у візуальному діапазоні, обробляти отриману інформацію для виділення певних ознак у реальному часі та створення відеоданих, на яких відображається результат виділення ознак.

На рис. 1 показана блок-схема інтелектуальної камери.

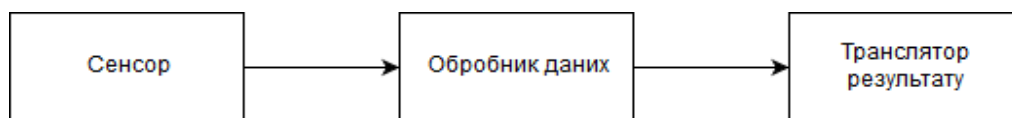


Рис. 1. Блок-схема інтелектуальної камери

Інтелектуальну камеру можна розділити на три функціональні блоки — «Сенсор», «Обробник даних» та «Транслятор результату». Таке розділення дозволяє замінювати тільки частину камери для підвищення ефективності розв'язання поставленої задачі.

Сенсор — пристрій, що передає обробнику даних неспотворену відеоінформацію для подальшої обробки. В залежності від задач як сенсор можуть бути задіяні як найпростіші сенсори типу HDCS2020 [3], так і складна мультисенсорна система, яка імітує око комахи [12].

Обробник даних — будь-який пристрій, який виконує обробку відеоінформації для її аналізу та інтерпретації. Це може бути DSP-процесор, мікроконтролер, мікропроцесор або ПЛІС.

Транслятор результату — пристрій (можливе фізичне суміщення з Обробником даних), який передає результат для подальшої реакції. У найпростішому випадку це може бути світлодіод; можливе використання графічного дисплея (фізичним суміщенням такого варіанта може бути канал передачі даних до нього).

У статті розглянуто варіант підключення тільки одного сенсора до обробника даних та транслятора результату.

Апаратні вимоги до реалізації інтелектуальної камери

Під апаратними вимогами для компонентів відеокамери слід розуміти такі вимоги:

1. Достатність оперативної пам'яті та обчислювальної потужності в «Обробнику даних»;
2. Підтримка зв'язку функціонального блока один з одним (електрична сумісність, підтримка протоколів та кодування) із застосуванням можливостей пришвидшення обробки в «Обробнику даних» (прямий доступ до пам'яті, векторна обробка даних, використання паралельних та/або гетерогенних наявних модулів «Обробника даних» та ін.);

Існує багато типів сенсорів, які різняться технологією виготовлення (CCD та CMOS), інтерфейсом передачі даних (паралельним чи послідовним), роздільною здатністю тощо. Для апаратної реалізації інтелектуальної відеокамери як сенсор використано так званий scientific CMOS-сенсор (сенсор виконаний за технологією КМОП, призначений для наукових цілей; в подальшому sCMOS-сенсор) WH300. Цей сенсор має роздільну здатність 0,3 МПк (640×480), паралельним 10-бітним інтерфейсом з максимальною швидкістю передачі даних до 40 МГц та керуванням через SPI-протокол. Як «Обробник даних» та «Транслятор результату» застосовується мікрокомп'ютер Orange Pi One з процесором Allwinner H3 (ARM Cortex-A7 Quad-core) та ОЗУ 512MiB DDR3. Підключення за допомогою 10-бітного інтерфейсу сенсора та одночасне використання DMA в «Обробнику результату» неможливе, тому сенсор підключений до перетворювача інтерфейсу parallel-to-SPI MAX 7317, який буде працювати зі швидкістю передачі даних 25 МГц та передавати дані для подальшої обробки до DMA. Час, який знадобиться для передачі одного кадру з камери до мікропроцесора можна достатньо точно розрахувати за допомогою формули

$$t = \frac{1}{f} \cdot \left(\frac{n}{b} \right),$$

де t — час передачі одного кадру, с; f — швидкість передачі кадру, Гц; b — розрядність однієї передачі, біт; n — загальна кількість інформації, біт.

Взявши $f = 25$ МГц; $b = 10$ та $n = 640 \cdot 480 = 307200$, з формули отримаємо $t \approx 1,23 \cdot 10^{-3}$ с, що дозволяє «Обробнику результату» приймати з сенсора до 810 кадрів в секунду. Для пришвидшення отримання результату можна використовувати частину кадру.

Особливості програмної реалізації камери

В залежності від вибору реалізації відеокамери буде відрізнятися і програмна (використання мікроконтролера або мікропроцесора) або апаратна (використання ПЛІС) складова архітектури. Однак, незалежно від реалізації, така камера повинна підтримувати виконання певних функцій (наприклад передачу зображення в певному форматі, маніпуляції з геометричними розмірами зображення та ін.). Таким чином, можливо створити модель, яка б описувала архітектурні особливості функціональних блоків незалежно від їх реалізації. Для створення моделі будемо використовувати патерни (patterns) програмування [13].

Оскільки сенсор є завершеною програмно-апаратною сутністю — немає практичної необхідності створювати свої апаратно-програмні реалізації архітектури, тому в подальшому розглядається тільки реалізація архітектури у мікрокомп'ютері. Вона виглядає таким чином:

- а) для прийняття та передачі інформації з сенсора використовується патерн «Adapter». Використання цього патерну пов'язано з неможливістю впливати на інтерфейси підключення сенсора та кількість підтримуваних форматів. З програмної точки зору цей патерн буде виглядати як на рис. 2.

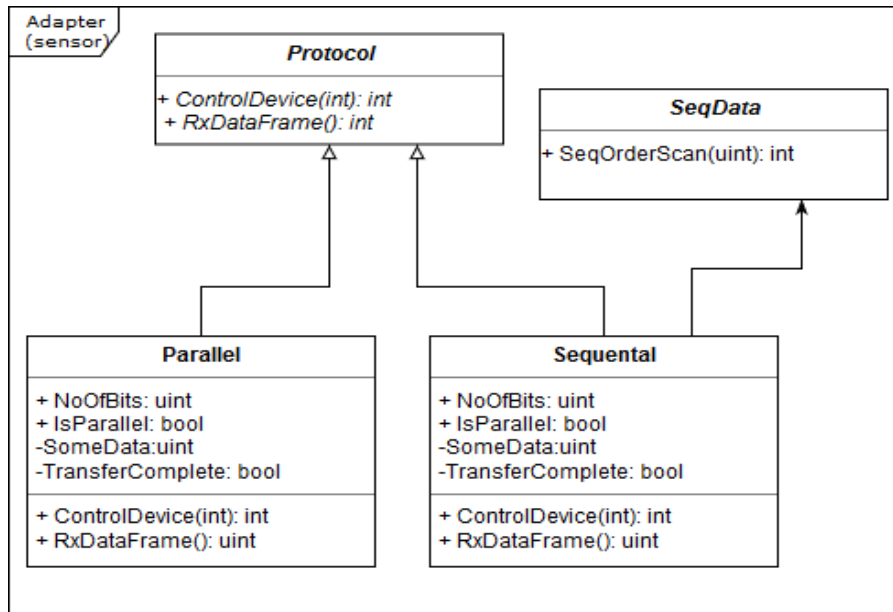


Рис. 2. Спрощена діаграма класів для прийняття та передачі інформації з сенсора

б) для створення сутностей та подальшої обробки отриманих даних типу «піксель», «рядок», «стовпчик», «вікно» та ін. та для їх організації використовуються патерн «Composite», необхідний для створення великої кількості однотипних функцій. Необхідний для створення відеопотоку з кадрів різного розміру (безпосереднє створення відбувається патерном «Abstract Factory», структура якого тривіальна і не розглядається у статті). Реалізація може бути такою як на рис. 3.

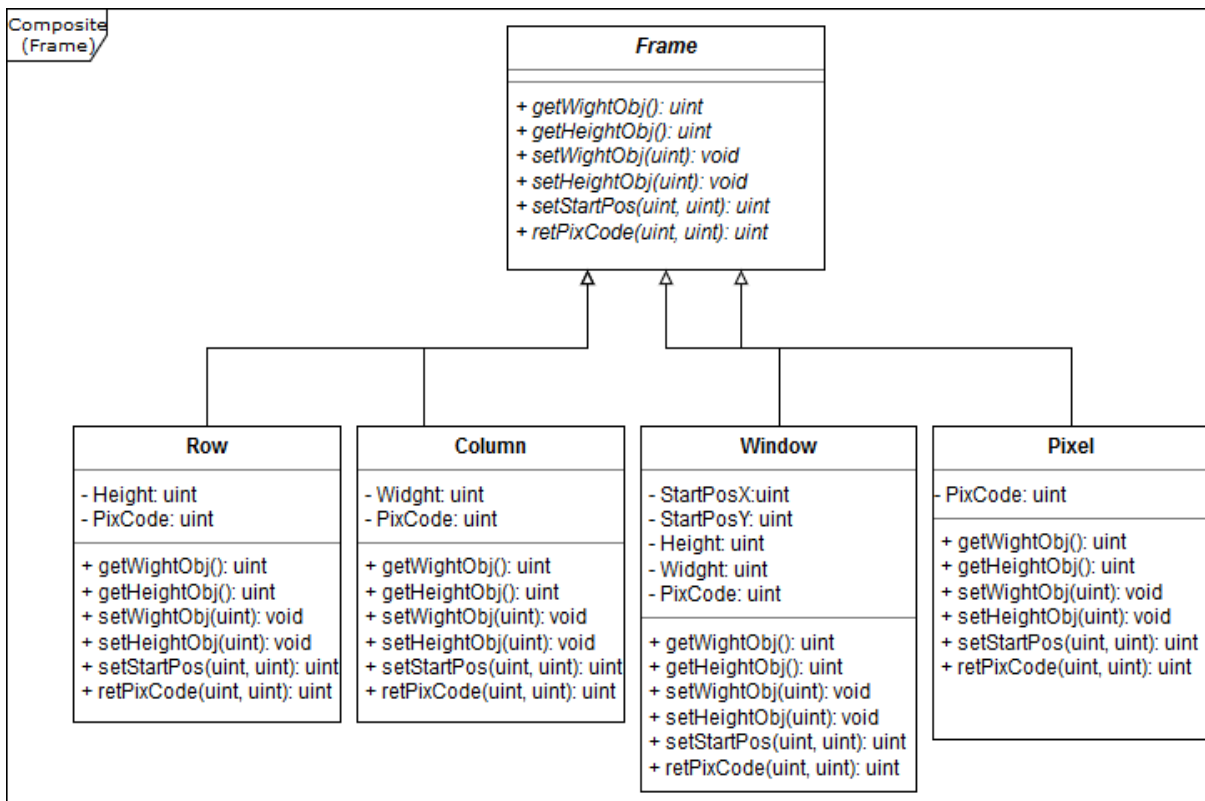


Рис. 3. Спрощена діаграма класів для структурування переданої інформації

Патерн «Flyweight» використовується для створення та використання однотипних сутностей, які застосовуються для виділення певних ознак у зображенні (наприклад, так званих «масок»). Необхідний для обробки отриманої інформації, який вже наявний і створений патерном «Composite».

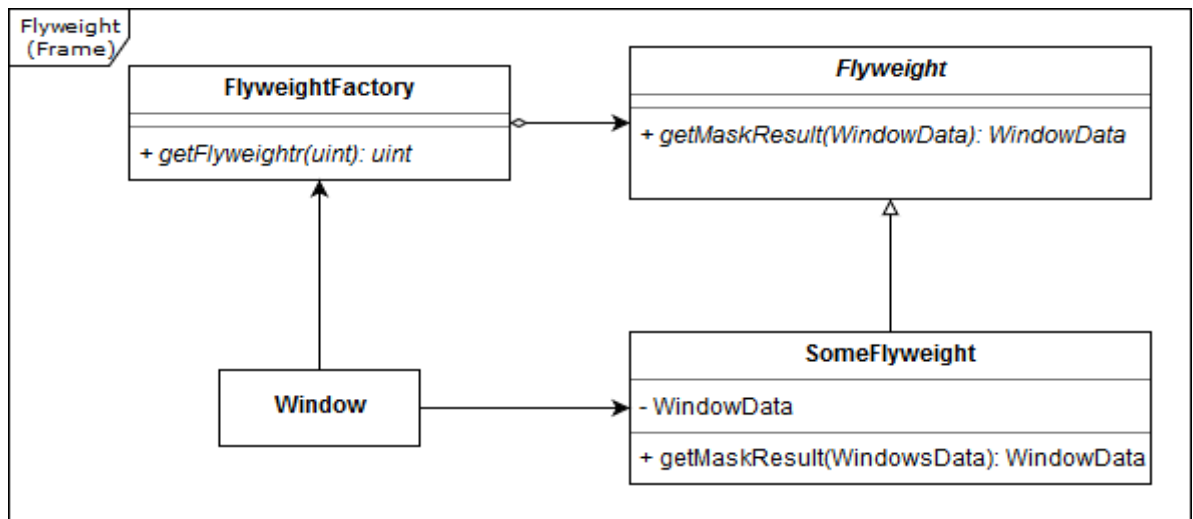


Рис. 4. Спрощена діаграма класів для обробки отриманої інформації

в) для видачі результату використовується патерн «Bridge», оскільки необхідно створити гнучкий інтерфейс виводу, який би швидко реконфігурувався для підключення до наявних інтерфейсів виводу даних. Такий патерн можна реалізувати так, як показано на рис. 5.

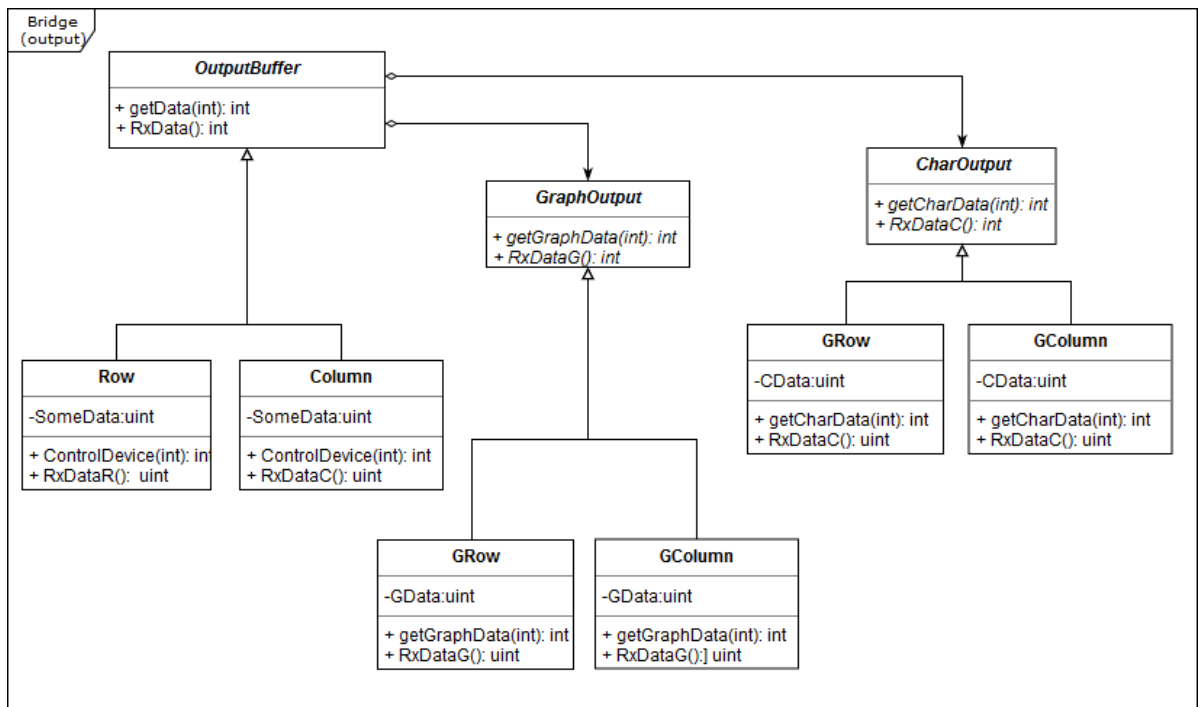


Рис. 5. Спрощена діаграма класів для передавання обробленої інформації

Висновки

1. Використання модульності в розробці нових типів інтелектуальних камер дозволить скоротити час розробки нової камери.
2. Запропонований модульний підхід конкретизації задачі дозволяє швидко вибрати і розробити нові варіанти інтелектуальних відеокамер для розв'язання конкретних задач.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ:

- [1] Roberto Arroyo, Javier Yebe, Luis M. Bergasa, Iván G. Daza, and Javier Almazán, "Expert video-surveillance system for real-time detection of suspicious behaviors in shopping malls," *Expert Systems with Applications*, vol. 42, pp. 7991-8005, 2015.

- [2] Vitaliy Boyun, "Directions of Development of Intelligent Real Time Video Systems," *Application and Theory of Computer Technology*, [S. l], v. 2, no. 3, pp. 48-66, apr. 2017. ISSN 2514-1694.
- [3] В. П. Боюн, Ю. А. Сабельников, «Интеллектуальная камера,» *Электронные компоненты и системы (ЭКиС)*, № 2, К., 2002, с. 17-23
- [4] Charles Bell, *Beginning Sensor Networks with Arduino and Raspberry Pi*, Apress, 2013, ISBN: 9781430258247.
- [5] *VEEK-MT2S*, [Online] Available: <https://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&CategoryNo=167&No=1098> .
- [6] Xilinx Zynq, *UltraScale+ MPSoC ZCU104 Evaluation Kit*, [Online], Available: <https://www.xilinx.com/products/boards-and-kits/zcu104.html> .
- [7] *NVIDIA Jetson AGX Xavier*, [Online], Available: <https://www.nvidia.com/en-us/autonomous-machines/jetson-agx-xavier/> .
- [8] John Williams, "Embedded Linux on FPGAs for fun and profit," *CELF Embedded Linux Conference 2009*. 2009 [Online], Available: https://elinux.org/images/5/54/ELC2009_Embedded_Linux_on_FPGAs_for_fun_and_profit.pdf .
- [9] Matteo Tomasi, "FPGA-DSP co-processing for feature tracking in smart video sensors," *Journal of Real-Time Image Processing*, no. 11, pp. 1-17, 2014.
- [10] А. Н. Головин, «Интеллектуальные видеокамеры: состояние, определение и классификация,» *Управляющие системы и машины (УСиМ)*, № 1, с. 46-54, 2013.
- [11] Yu Shi Eveleigh, and Serge Lichman, *Smart Cameras: A Review*, National ICT Australia Ltd, 2007.
- [12] Vladan Popovic, Kerem Seyid, Ömer Cogal, Abdulkadir Akin, and Yusuf Leblebici, "Design and Implementation of Real-Time Multi-Sensor Vision Systems," *Springer International Publishing AG 2017*, ISBN 978-3-319-59056-1. <https://doi.org/10.1007/978-3-319-59057-8>.
- [13] Э. Гамма, Р. Хелм, Р. Джонсон, и Дж. Влссидес, *Приемы объектно-ориентированного проектирования. Паттерны проектирования*. СПб., Россия: Питер, 2001, 368 с.

Рекомендована кафедрою програмного забезпечення ВНТУ

Стаття надійшла до редакції 10.10.2016

Багацький Олексій Валентинович — канд. техн. наук, старший науковий співробітник, e-mail: bagatskyu.o.v@gmail.com .

Інститут кібернетики ім. В. М. Глушкова НАН України, Київ

O. V. Bahatskyi¹

Software Realization of Intellectual Video Camera

¹V. M. Glushkov Institute of Cybernetics NASU, Kyiv

At present video cameras are widely distributed in the world and are used for a wide range of real-time tasks - from using optical manipulators to identifying a person. However, at the moment there are no universal cameras that could be effectively used in the entire spectrum of tasks. This is due to the large number of video stream parameters (height and width, brightness, frame contrast) and requirements for both hardware (speed, resistance to external influences) and software (video stream coding, supporting coding) for providing such cameras. This paper analyzes the modular approach to creating an intelligent video camera consisting of several modules. According to the results of the analysis of existing intelligent cameras, their advantages and disadvantages, it was proposed to consider the implementation of existing cameras for three cases. It is proposed to analyze the behavior of an intelligent video camera as a composite device, which consists of the following parts — "Sensor" (a device that transmits undistorted video information to the data handler for further processing), "Data processor" (any device that performs video processing for its further analysis and interpretation) and "Result Translator" (a device that transmits the result for further processing), which allows you to quickly create specialized cameras based on the essence. Requirements were formulated for the hardware component of the options for the practical implementation of the video camera and calculated the maximum speed of information transfer from the "Sensor" to the "Data processor". Also, software architecture is proposed for the functioning of the camera. UML-class diagrams for critical program modules were created that describe data reception from the "Sensor" to the "Data processor", data processing in "Data processor", and transfer the result for further processing from the "Data processor" to the "Result Translator".

Keywords: intelligent video camera, real-time systems, software architecture.

Bahatskyi Oleksii V. — Cand. Sc. (Eng.), Senior Researcher, e-mail: bagatskyu.o.v@gmail.com

Аппаратно-программная реализация интеллектуальной видеокамеры

¹Институт кибернетики им. В. М. Глушкова НАН Украины, Киев

В настоящее время видеокамеры широко распространены и используются для решения широкого спектра задач реального времени — от использования в оптических манипуляторах до идентификации человека. Однако, на данный момент не существует универсальных видеокамер, которые могли бы эффективно использоваться во всех задачах. Это связано с большим количеством параметров видеопотока (высота и ширина, яркость, контраст кадра) и требованиям как к аппаратному (быстродействие, устойчивость к внешним воздействиям), так и к программному (кодирование видеопотока, поддержка протоколов данных) обеспечению таких камер. В работе проанализирован модульный подход к созданию интеллектуальной видеокамеры, состоящей из нескольких модулей. По результатам проведенного анализа существующих интеллектуальных видеокамер, их преимуществ и недостатков предложено классифицировать реализации видеокамер на три случая. Предложено анализировать поведение интеллектуальной видеокамеры как устройство, которое состоит из нескольких частей — «Сенсор» (устройство, передающее обработчику данные неискаженной видеоинформации), «Обработчик данных» (любое устройство, которое выполняет обработку видеоинформации для ее дальнейшего анализа и интерпретации) и «Транслятор результата» (устройство, которое передает результат для дальнейшей реакции), что позволит быстро создавать специализированные камеры на базе существующей. Сформулированы требования к аппаратной составляющей вариантов практической реализации видеокамеры и рассчитана предельная скорость передачи информации от «Сенсора» до «Обработчика данных». Предложена программная архитектура, которая необходима для функционирования камеры. Созданы UML-диаграммы классов для ключевых программных модулей, которые описывают прием данных от «Сенсора» до «Обработчик данных», обработку данных в «Обработчике данных» и передачу информации для дальнейшего анализа от «Обработчик данных» к «Транслятору результата».

Ключевые слова: интеллектуальная видеокамера, системы реального времени, программная архитектура.

Багацкий Алексей Валентинович — канд. техн. наук, старший научный сотрудник, e-mail: bagatskyu.o.v@gmail.com