

ДОСЛІДЖЕННЯ МЕТОДІВ ВИЯВЛЕННЯ ОБ'ЄКТІВ НА ВІДЕОЗОБРАЖЕННЯХ

В. Я. Пуйда, А. О. Стоян

Національний університет "Львівська політехніка",
кафедра електронних обчислювальних машин

© Пуйда В. Я., Стоян А. О., 2020

Завдання виявлення об'єктів на відеозображеннях характерна для сучасних систем технічного зору (СТЗ), орієнтованих на різні функціональні застосування. Виявляти об'єкти можна як на статичних відеозображеннях, так і на виділених з відеопотоку кадрах. За своєю суттю виявлення об'єктів на відеозображенні, як правило, означає виявлення яскравісних чи кольорових неоднорідностей, які на подальших етапах можна трактувати як фізичні об'єкти. Крім цього, ще можна виконувати операції визначення координат, лінійних розмірів та інших характеристик цих неоднорідностей, які надалі використовувати для розв'язання інших задач в СТЗ, наприклад, для ідентифікації об'єктів.

Досліджено три алгоритми, які можна використати для виявлення об'єктів різної природи за різними підходами: виявлення кольорових неоднорідностей, визначення міжкадрової різниці, використання детектора особливих точок. Як вхідну інформацію використовують відеопотік, що вводиться з відеокамери або з файла типу "mp4". Моделювали алгоритми на універсальному комп'ютері та на апаратній платформі з відкритим кодом, побудованій на базі процесора Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC із робочою частотою 1,5 GHz. Програми моделювання підготовлено в середовищі Visual Studio 2019 з використанням бібліотек OpenCV4 для Windows 10 на універсальному ПК та Linux (OS Raspbian Buster) для платформи з відкритим кодом. Здійснено порівняльний аналіз вибраних методів. Отримані результати можна використати в наукових дослідженнях та для проектування реальних СТЗ різного функціонального призначення.

Ключові слова: виявлення об'єктів на відеозображенні, особливі точки, детектор ORB, комп'ютерний зір, виявлення рухомих об'єктів, HSV колірна модель.

Вступ

Виявлення об'єктів є однією з основних задач побудови СТЗ. Сьогодні не вдалося розробити універсальний алгоритм для виявлення об'єктів на різних типах відеозображень. Причина в різноманітності завдань, для вирішення яких проектується СТЗ, у різній природі об'єктів та в специфіці сцен, на відеозображенні яких необхідно виділити об'єкт. Ці причини вимагають розроблення спеціалізованих алгоритмів або адаптації відомих алгоритмів до конкретної задачі та конкретних умов. Особливість алгоритмів та задач, які має вирішувати конкретна СТЗ, накладає

відповідні вимоги до апаратного забезпечення, яке може базуватися на високопродуктивних універсальних ПК, міні чи мікрокомп'ютерах або на спеціалізованих апаратних засобах.

У роботі розглянуто такі методи:

- виявлення об'єктів за допомогою кольорових фільтрів;
- виявлення рухомих об'єктів за допомогою міжкадрової різниці;
- виявлення об'єктів за допомогою детектора особливих точок ORB.

Для дослідження цих методів розроблено програми моделювання відповідних алгоритмів для виконання на високопродуктивному персональному комп'ютері (ПК) з ОС **Windows 10** та на мікрокомп'ютері, спроектованому на базі процесора Broadcom BCM2711 з 4 ядерним Cortex-A72 та робочою частотою 1,5 ГГц з інстальною Linux ОС Raspbian Buster. Програми моделювання розроблено в середовищі Visual Studio 2019 із використанням відповідних бібліотек OpenCV4.

У результаті моделювання здійснено порівняльний аналіз алгоритмів, що дасть можливість вибрати конкретний метод для проектування відповідної СТЗ.

Аналіз публікацій

Задачі виявлення об'єктів виникали в різних сферах: від промислових технічних систем, наприклад, для визначення дефектів у деталях, до систем спеціального призначення, наприклад, у медицині, радіолокації, при побудові цифрових систем зв'язку тощо [1–9]. Розроблялися різноманітні підходи та методи оброблення сигналів [10–15], наприклад, кореляційні, узгодженої фільтрації.

Із розвитком обчислювальних засобів, появою високопродуктивних, малогабаритних комп'ютерних систем почалося активне проектування СТЗ для різноманітних застосувань. Це привело до інтенсивних досліджень в області адаптації відомих методів виявлення об'єктів та розроблення нових для розв'язання задачі виявлення об'єктів на відеозображеннях. З'явилися різні підходи, методи та реалізовані на їх основі алгоритми, які успішно використовуються в спеціалізованих СТЗ, наприклад, використання кольорових фільтрів [1], визначення міжкадрової різниці у відеопотоці [2], побудова різноманітних детекторів особливих точок, які дозволяють виявляти об'єкти на основі їх еталонних зображень. Розроблено ефективні методи виявлення особливих точок та на їх основі алгоритми для різних областей застосування, наприклад, ORB [3, 5], SURF [12–13], FAST [7]. Для формування дескрипторів використовують, наприклад, алгоритм BRIEF [4].

Постановка завдання

Провести дослідження алгоритмів на основі таких методів виявлення об'єктів в кадрі відеозображення, виділеного з відеопотоку:

- виявлення об'єктів за допомогою кольорових фільтрів;
- виявлення рухомих об'єктів за допомогою міжкадрової різниці;
- виявлення об'єктів за допомогою детектора особливих точок ORB.

Для дослідження алгоритмів підготувати програми з реалізацією на апаратних платформах різної продуктивності:

Вирішення завдання

Метод виявлення об'єктів за допомогою кольорових фільтрів ґрунтується на фільтруванні зображення за кольором: шукають об'єкт коли він істотно відрізняється за кольором від фону та освітлення.

Реалізують алгоритм такими етапами:

- 1) виділення RGB-кадру з відеопотоку;

2) конвертація RGB-кадру в HSV-кадр, яку здійснюють для комфортного сприйняття кольорів людиною, де HSV (Hue, Saturation, Value-Brightness) – колірна модель представлення зображення з трьома параметрами [8]:

а) Hue – колірний тон, (наприклад, червоний, зелений або синьо-блакитний) в діапазоні 0–360 (деколи приводиться в діапазоні 0–100);

б) Saturation – насиченість у діапазоні 0–100;

в) Value-Brightness – яскравість у діапазоні 0–100.

Конвертація RGB у HSV здійснюється так [9]:

якщо $0 \leq H < 360$, $0 \leq S \leq 1$ і $0 \leq V \leq 1$:

$$C = V \times S;$$

$$X = C \times (1 - |(H / 60^\circ) \bmod 2 - 1|);$$

$$m = V - C;$$

$$(R', G', B') = \begin{cases} (C, X, 0), & 0^\circ \leq H < 60^\circ \\ (X, C, 0), & 60^\circ \leq H < 120^\circ \\ (0, C, X), & 120^\circ \leq H < 180^\circ \\ (0, X, C), & 180^\circ \leq H < 240^\circ \\ (X, 0, C), & 240^\circ \leq H < 300^\circ \\ (C, 0, X), & 300^\circ \leq H < 360^\circ \end{cases}$$

$$(R, G, B) = ((R'+m) \times 255, (G'+m) \times 255, (B'+m) \times 255) \quad (1)$$

3) встановлення необхідного порогу кольорів на основі HSV моделі;

4) на основі сформованого діапазону кольорів формується бінарне зображення;

5) видалення шумів, наприклад, операціями ERODE та DILATE);

7) пошук об'єкта за його бінарним зображенням.

На рис. 1 показано скріншоти екранів вхідного кадру, бінарного зображення та виявленого об'єкта на відеозображенні за методом кольорових фільтрів при виконанні програми на універсальному ПК. Виділений об'єкт обведено колом.



Рис. 1. Скріншоти екранів вхідного кадру, бінарного зображення та виявленого об'єкта за методом кольорових фільтрів

Метод обчислення міжкадрової різниці (frame difference) є простішим методом виявлення рухомих об'єктів у відеопотоці і ефективніше працює для виявлення рухомих об'єктів на статичному фоні.

Реалізується алгоритм такими етапами:

- 1) відеокамера у певний момент часу фіксує послідовно два кадри;
- 2) ці кадри конвертуються в монохромне напівтонове зображення;
- 3) проводиться попіксельне обчислення різниці двох монохромних кадрів [5]:

$$d_t(x,y) = I_t(x,y) - I_{t-1}(x,y), \quad (2)$$

де $I_t(x,y)$ – інтегральне значення пікселя з координатою x, y отриманого кадру в часі t ; $I_{t-1}(x,y)$ – інтегральне значення пікселя з координатою x, y отриманого кадру в часі $t-1$

- 4) отримана різниця $d_t(x,y)$ порівнюється з порогом T і формується бінарна маска:

$$m_t(x,y) = \begin{cases} 0, & \text{якщо } d_t(x,y) < T \\ 1, & \text{в іншому випадку} \end{cases} \quad (3)$$

де $m_t(x,y)$ – значення t -го елемента маски; T – встановлений поріг (рівень чутливості);

- 5) отримана маска обробляється методом математичної морфології [16];
- б) виявлення об'єкта.

Метод обчислення міжкадрової різниці на практиці краще працює за відносно повільного переміщення об'єктів у полі зору відеокамери. На рис. 2 показано скріншоти екранів вхідного кадру, бінарного зображення та виявленого об'єкта на відеозображенні за методом міжкадрової різниці при виконанні програми на універсальному ПК. Виділений об'єкт обведено прямокутником.

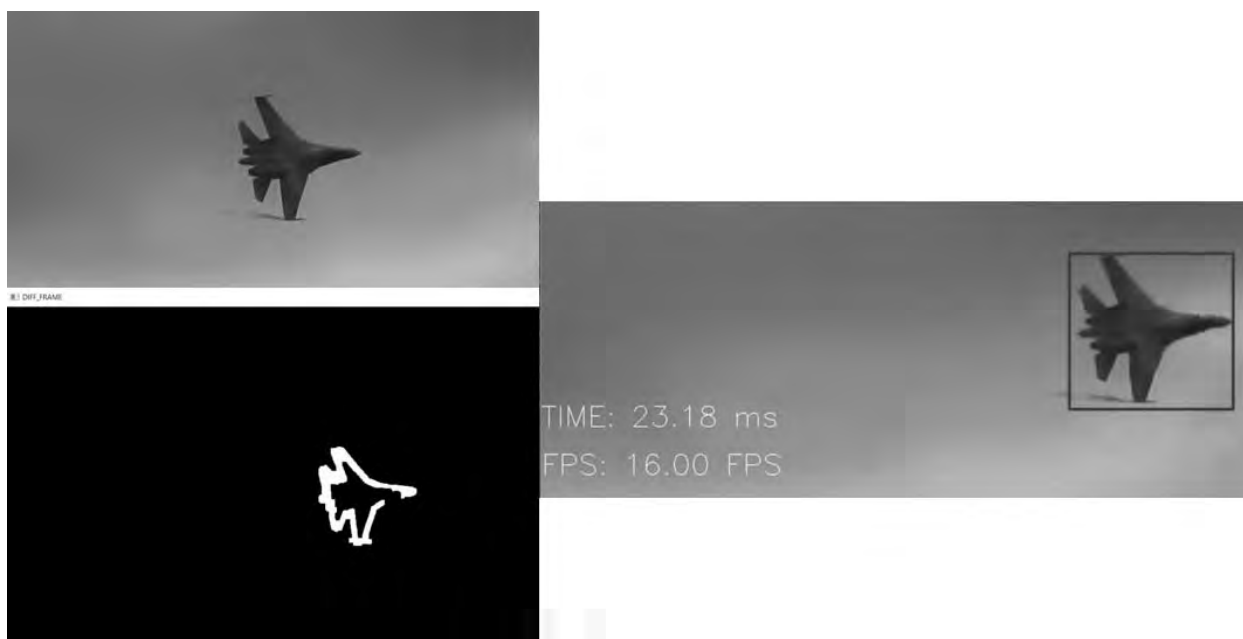


Рис. 2. Скріншоти екранів вхідного кадру, бінарного зображення та виявленого об'єкта за методом міжкадрової різниці

Метод виявлення конкретного об'єкта в кадрі відеозображення можна здійснювати за оригінальним зображенням об'єкта-еталону. Еталон порівнюється з кожним кадром із відеопотоку і область кадру, яка має відповідні ознаки подібності з еталоном, визначає наявність шуканого об'єкта. Існує багато методів виявлення об'єктів за їхніми еталонами. Одним з найпродуктивніших та надійних вважають метод виявлення за допомогою особливих точок. Ознакою особливості точки

може бути, наприклад, різкий перепад кольорів, яскравості околиці, кути, краї тощо. Виявлення та фіксування особливих точок реалізується з допомогою детектора особливих точок та формування дескриптора. В алгоритмі ORB для виявлення особливих точок використовують детектор кутів Харріса [6] та піраміду зображень [11]. Піраміда зображення – це багатомасштабне зображення одного кадру, яке складається з послідовностей зображень із різною роздільною здатністю. Кожен рівень піраміди містить зменшену версію зображення попереднього рівня, що забезпечує часткову інваріантність щодо зміни масштабу зображення.

На рис. 3 показано скріншоти екранів вхідного кадру, еталону та виявленого об'єкта на відеозображенні за методом пошуку особливих точок детектором ORB.



Рис. 3. Скріншоти екранів вхідного кадру, еталону та виявленого об'єкта за методом особливих точок детектора ORB

Моделювання алгоритмів проводили на універсальному ПК (ноутбук) із процесором Intel Core i7 2,4 ГГц, ОС Windows 10 та на мікрокомп'ютері з процесором Broadcom BCM2711 1,5 ГГц, Linux ОС Raspbian Buster, функціональну схему якого наведено на рис. 4. Програму для мікрокомп'ютера готували на ПК у середовищі Visual Studio 2019 у вигляді проекту під Linux та завантажували для виконання на мікрокомп'ютер через мережевий вузол IEEE 802.11. Результати виконання програми можна передавати в СТЗ через інтерфейс USB 3.0. У разі використання мікрокомп'ютера в складі реальної СТЗ виконавчий код програми можна записати на системну SD-карту.

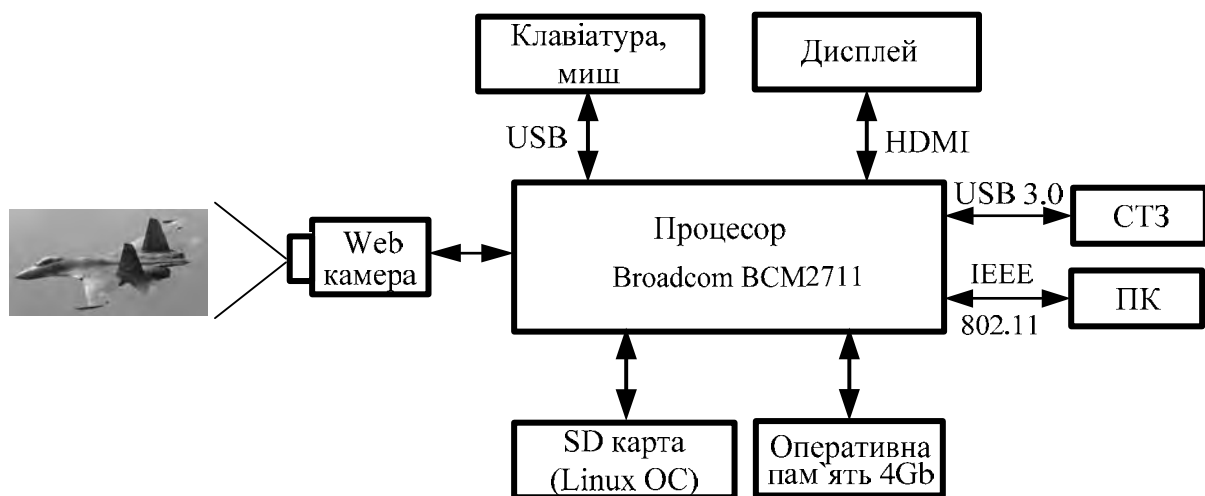


Рис. 4. Мікрокомп'ютер на базі процесора Broadcom BCM2711 з Linux ОС Raspbian Buster

Програми для універсального ПК та для мікрокомп'ютера реалізовано мовою програмування C++ у середовищі Visual Studio 2019 із використанням відповідних бібліотек OpenCV4.

Порівнювали методи на двох типах відеопотоків: політ літака в хмарному небі та транспортний потік на автомагістралі. У табл. 1,2 наведено середній час виконання програм під час реалізації різних методів та величину FPS, яка вказує на кількість кадрів на секунду, які камера встигає поновити при виконанні програми.

Таблиця 1

Порівняння методів виявлення об'єктів на відеозображенні літака

	FPS (кількість кадрів в секунду)	Час виконання на універсальному ПК, мс	Час виконання на Broadcom BCM2711, мс
Метод виявлення об'єктів за допомогою кольорових фільтрів	22,0	6,69	11,2
Метод виявлення рухомих об'єктів за допомогою міжкадрової різниці	16,0	23,18	32,4
Метод виявлення об'єктів за допомогою ORB-детектора особливих точок	33,85	41,24	67,1



Рис. 5. Скріншот кадру відеозображення транспортного потоку

Таблиця 2

Порівняння методів виявлення об'єктів на відеозображенні транспортного потоку

	FPS (кількість кадрів в секунду)	Час виконання на універсальному ПК, мс	Час виконання на Broadcom BCM2711, мс
Метод виявлення об'єктів за допомогою кольорових фільтрів	14,1	18,34	34,5
Метод виявлення рухомих об'єктів за допомогою міжкадрової різниці	16,3	9,15	12,1
Метод виявлення об'єктів за допомогою ORB-детектора особливих точок	9,6	32,13	71,4

Висновки

Здійснено програмне моделювання методів виявлення об'єктів на відеозображенні на основі використання кольорових фільтрів, визначення міжкадрової різниці у відеопотоці та побудови ORB-детектора особливих точок. Моделювання проведено на універсальному ПК (ноутбук) із процесором Intel Core i7 2,4 ГГц, ОС Windows 10 та на мікрокомп'ютері з процесором Broadcom BCM2711 1,5 ГГц з Linux ОС Raspbian Buster. Програми підготовлено в середовищі Visual Studio 2019 із використанням відповідних бібліотек OpenCV4. У процесі моделювання алгоритмів виявилося, що час виявлення об'єкта значною мірою залежить від фону та яскравісної характеристики об'єкта. Так, для одних параметрів найменший час виявлення об'єкта на відеозображенні виявився для міжкадрової різниці, а для інших параметрів для методу кольорових фільтрів. У всіх випадках найбільшим виявився час виявлення об'єкта для методу на основі ORB-детектора особливих точок.

Список літератури

1. A Fast HSV Image Color and Texture Detection and Image Conversion Algorithm / Monika Deswal1 , Neetu Sharma2 – *International Journal of Science and Research (IJSR)*.
2. Wenchao Huang, Yake Kang, Song Zheng. An improved frame difference method for moving target detection, *Chinese Automation Congress (CAC) 2017*, pp. 1537–1541, 2017.
3. An Improved ORB Algorithm of Extracting and Matching Features / Lei Yu1 , Zhixin Yu1 and Yan Gong2 - *International Journal of Signal Processing, Image Processing and Pattern Recognition Vol. 8, No. 5 (2015)*, pp. 117–126.
4. Calonder M., Lepetit V., Strecha C., Fua P. BRIEF: Binary Robust Independent Elementary Features // *11th European Conference on Computer Vision (ECCV)*, Heraklion, Crete. LNCS Springer, September 2010.
5. ORB: an efficient alternative to SIFT or SURF / Ethan Rublee, Vincent Rabaud, Kurt Konolige Gary Bradski Willow Garage, Menlo Park, California.
6. Harris, C., Stephens, M., 1988, A Combined Corner and Edge Detector, *Proceedings of 4th AlveyVision Conference*.
7. R. P. a. T. D. Edward Rosten. *Faster and better: a machine learning approach to corner detection*, 2008.
8. A Fast HSV Image Color and Texture Detection and Image Conversion Algorithm / Monika Deswal1, Neetu Sharma2. *International Journal of Science and Research (IJSR)*/
9. Max K. Agoston. *Computer Graphics and Geometric Modelling: Mathematics (v. 2) 2005th Edition*.
10. Jayaraman, S. *Digital Image Processing*. Tata McGraw Hill, 2009.
11. *Multiresolution Image Processing and Analysis* / A. Rosenfeld. Springer-Verlag, 1984.
12. Baya H., Essa A., Tuytelaarsb T., Van Gool L., *Speeded-Up Robust Features (SURF)*, *Computer Vision and Image Understanding* 110 (2008), no. 3, 346–359.
13. Volodymyr Puyda. SURF features extraction in a computer vision system. *advances in cyber-physical systems, Volume 1, Number 1 (2017)*.
14. Lozynsky V., *The matching filtration on the basis of differential PCM* // *Proceedings of the International Conference Metody i Technika Przetwarzania Sygnaiyw w Pomiarach Fizycznych MSM'2003*, Rzeszyw, Poland, 2003. S. 193–201.
15. Kharkevych O. O. *Spektry ta analiz. M.: Fizmathiz, 1962. 236 p. (in Russian)*.

ON METHODS OF OBJECT DETECTION IN VIDEO STREAMS**V. Puyda, A. Stoian**Lviv Polytechnic National University
Computer Engineering Department

© Puyda V., Stoian A., 2020

Detecting objects in a video stream is a typical problem in modern computer vision systems that are used in multiple areas. Object detection can be done on both static images and on frames of a video stream. Essentially, object detection means finding color and intensity non-uniformities which can be treated as physical objects. Beside that, the operations of finding coordinates, size and other characteristics of these non-uniformities that can be used to solve other computer vision related problems like object identification can be executed. In this paper, we study three algorithms which can be used to detect objects of different nature and are based on different approaches: detection of color non-uniformities, frame difference and feature detection. As the input data, we use a video stream which is obtained from a video camera or from an mp4 video file. Simulations and testing of the algorithms were done on a universal computer based on an open-source hardware, built on the Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC processor with frequency 1,5 GHz. The software was created in Visual Studio 2019 using OpenCV 4 on Windows 10 and on a universal computer operated under Linux (Raspbian Buster OS) for an open-source hardware. In the paper, the methods under consideration are compared. The results of the paper can be used in research and development of modern computer vision systems used for different purposes.

Keywords: object detection, feature points, keypoints, ORB detector, computer vision, motion detection, HSV model color.