

ДЕТЕКЦІЯ РУХУ У СТИСНУТОМУ ВІДЕОПОТОЦІ В РЕАЛЬНОМУ ЧАСІ

У статті описано швидкий, ефективний та простий метод визначення областей змін на зображенні та детекції руху у стиснутому відеопотоці. Запропонований метод працює безпосередньо зі стиснутим відео у форматі H.264.

Ключові слова: детекція руху, H.264, вектори руху.

Вступ

Детекція руху у відеопотоці є одним з найважливіших завдань у розробці систем відеоспостереження.

Для детекції руху широко використовуються методи віднімання фону “Background subtraction”, також відомі як методи детекції переднього фону “Foreground Detection” в просторовій області. Лише бібліотека bgslibrary на сьогодні пропонує 29 алгоритмів віднімання фону. В результаті застосування цих методів, об’єкти, які рухаються, будуть відсегментовані на вихідному бінарному зображенні. Для застосування більшості з цих алгоритмів необхідна також попередня та постобробка відеопотоку з метою покращення якості відсегментованих об’єктів.

Наприклад, алгоритм PBAS з бібліотеки bgslibrary, на думку Stefano Tommesani [5] та за результатами дослідження алгоритмів 2012 р. [2], за усіма категоріями є найкращим. Переваги PBAS алгоритму: відсутність помилкової класифікації, швидке усунення артефактів, відмінна сегментація об’єктів із заповненням, добра адаптація на раптову зміну фону, автоматична адаптація до шуму. Але є один головний недолік – це висока обчислювальна вартість. Отже, існує необхідність в альтернативному підході для аналізу відео.

Альтернативним підходом є розробка та реалізація алгоритмів обробки зображень і відео, які працюють безпосередньо у стиснутій області. Більшість цифрових зображень і відеопотоків сьогодні зберігаються і передаються в стислому вигляді. Перевага передачі відео в стислому форматі очевидна. Різні сучасні методи стиснення: MPEG-1, MPEG-2, MPEG-4, H.261, H.263 і H.264/MPEG-4, з’явилися протягом останніх двох десятиліть. Еволюція сучасних стандартів кодування допомагає краще стискати відеодані, водночас на розпакування відео потрібно більше ресурсів. Алгоритми обробки зображень і відео, які працю-

ють безпосередньо у стиснутій області, дозволяють уникнути зайві обчислювальні витрати на декодування та подальшу обробку зображення в просторовій області.

У цій статті ми розглянемо питання щодо детекції руху у відеопотоці в реальному часі у стиснутій області. Алгоритм компенсації руху, який застосовується при стиску відеоданих у форматі H.264, дозволяє просто реалізувати метод детекції руху у відеопотоці. Ефективність запропонованого методу ілюструється в рамках програми відеоспостереження.

Знаходження відмінностей на зображенні у відеопотоці

Знаходження відмінностей є важливою операцією обробки зображень, яка ідентифікує зміни пікселів, пов’язаних з рухом / зміною об’єкта в заданому фоні (до фону ми відносимо усі об’єкти, що не рухаються). Як результат цієї операції, отримуємо пікселі або області, які можна класифікувати як змінені або без змін на основі заданого критерію.

Під час стиску відеоданих у форматі H.264 застосовується алгоритм компенсації руху, який дозволяє точно знаходити відмінності на зображенні у відеопотоці. Формат H.264 є новим і ефективним стандартом кодування відео.

Кілька методів знаходження змін у стиснутій області в MPEG-x і H.264/AVC для зображень і відеопотоків описано в [1; 3; 4; 6; 7]. Одна з ключових переваг методів визначення відмінностей полягає у простоті їхньої реалізації. Проте вони мають недоліки — це висока чутливість до шуму та освітлення.

Основні компоненти кодованого відео

Для аналізу відео в стисненій області необхідно отримати основні компоненти з кодованого

відео. Основними компонентами є вектори руху. За стандартом відеокадри кодуються в одиницях 16 на 16 (16x16) макроблоків. Відповідно кожний макроблок може бути поділений на блоки 16x8, 8x16, 8x8, 4x8, 8x4 та 4x4 пікселів (рис. 1).

користанням компенсації, з посиланням на інший I- або P-кадр. В-кадри, або «двонаправлені» кадри, містять теж intra- і inter-кодовані макроблоки з посиланням на один кадр або на два найближчих кадри I-, P або B-кадри (рис. 2).

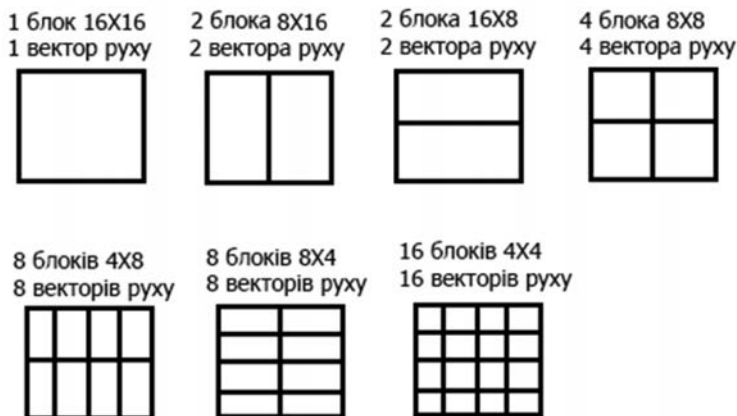


Рис. 1. Поділ на блоки та наявність векторів руху

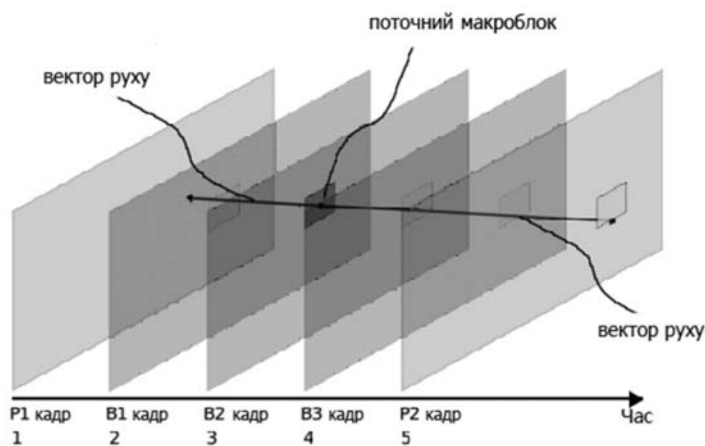


Рис. 2. Застосування компенсації руху до макроблоку В-кадру

Блок може бути передбачений іншим блоком з попередніх фреймів, якщо різниця між блоками є мінімальною. Різниця між блоками описується компенсацією руху. Блоки не змінюються, а зсуваються в положення передбаченого блоку. Цей зсув зображено вектором руху (рис. 2). Алгоритм компенсації руху дозволяє точно виділяти області руху.

Кадри відеопотоку відрізняються за типами залежно від способу кодування, зберігання інформації та наявності або відсутності залежностей цього кадру від попередніх і наступних.

Залежно від типу кадру існують також і різні типи макроблоків. I-кадри відеопотоку, або ключові кадри, можуть складатися тільки з intra-кодованих макроблоків – це незалежно стислі макроблоки, без використання компенсації руху. Тоді як P-кадри, «різницеві» кадри можуть містити intra- і inter-кодовані макроблоки, з ви-

На підставі вилученої інформації про вектори руху макроблоків у P- і B-кадрах, ми можемо визначити чи відбулися зміни та знайти область рухомого об'єкта.

Метод детекції руху у стиснутому відеопотоці

Нашим завданням є детекція змін на зображенні за наявністю векторів руху. Кожний inter-кодований блок з P- і B-кадрів має вектор руху, цей блок будемо називати блоком руху, оскільки саме в цьому блоці відбулися зміни. Багато «помилкових» блоків руху з'являється, якщо враховувати всі присутні вектори руху, навіть коли немає руху та об'єкт не присутній (рис. 3).

Для вирішення цієї проблеми застосовують фільтрацію. Фільтрація векторів руху є дуже важливим завданням при аналізі руху в області стис-



Рис. 3. Приклад розташування усіх блоків руху на зображенні (позначені квадратами)



Рис. 4. Приклад розташування «помилкових» блоків руху, що представлені векторами в блоках 16×16 на зображенні, після фільтрації

нення, оскільки від цього залежить надійність результату. Найчастіше використовують порогове значення. Якщо відстань вектора менше заданого порогового значення, вектор руху не враховують. Додатково застосовують медіанний фільтр разом з усереднюючим. У [7] запропоновано деякий модифікований фільтр Mean-Accumulated-Thresholded (MAT) filter. У [4] використовують каскадний фільтр, який складається з двох послідовних фільтрів Гауса та медіанного.

Досліджено, що при застосуванні модифікованого фільтра [7], каскадного фільтра [4] та окремо при застосуванні гаусового, медіанного,

усереднюючого фільтра та порогового значення, залишаються «помилкові» блоки та губляться дрібні об'єкти, що повільно рухаються. Це пов'язано з тим, що об'єкти рухаються з різною швидкістю. Якщо рух повільний, відстань векторів буде незначною, а через недоліки освітлення та передачу даних відстань векторів може бути значною.

Блоки руху з ненульовими векторами руху знаходяться у тій області зображення, де знаходиться об'єкт, що рухається. Чим більше значення вектору руху, тим ближче розташований об'єкт або тим швидше рухається.



Рис. 5. Приклад розташування «істинних» блоків руху, що представлено векторами в блоках 8×8 , 16×8 та 8×16 на зображенні, після фільтрації



Рис. 6. Приклад розташування «істинних» блоків руху, що представлено векторами в блоках 8×8 , 16×8 та 8×16 на зображенні, після фільтрації

Для знаходження «істинних» блоків руху ми знаходимо лише довжини векторів руху в блоках 8×8 , 16×8 та 8×16 , оскільки виявили, що «помилкові» блоки частіше знаходяться в неподілених макроблоках 16×16 (рис. 4). А «істинні» блоки руху представлено векторами в блоках 8×8 , 16×8 та 8×16 . Приклади подано на рис. 5 і 6, де «істинні» блоки руху на зображенні виділено квадратами.

Додатково ми використовуємо фільтрацію. Вектор руху повинен мати більше двох ненульових сусідніх векторів і мати відстань більше двох. Це впливає з розміру досліджуваного об'єкта.

Після фільтрації можемо проаналізувати зображення за таким параметром, як сума довжин усіх векторів руху. Якщо сума довжин усіх векторів більше відповідного порогового значення h ,

можна стверджувати, що на зображенні присутні рухомі об'єкти, відповідно можна знайти області рухомих об'єктів на зображенні. Порогове значення h задається залежно від необхідності не враховувати дрібний рух, наприклад рух птахів.

Висновки

У цій статті запропоновано простий, швидкий та ефективний метод детекції руху на зображенні у стиснутому відеопотоці в реальному часі на підставі аналізу та фільтрації векторів руху. Метод працює безпосередньо зі стиснутим відео у форматі H.264 та ефективно використовується в рамках програмного комплексу відеоспостереження.

Список літератури

1. An Effective and Fast Scene Change Detection Algorithm for MPEG Compressed Videos / [Z. Li, J. Jiang, G. Xiao, H. Fang] // ICIAR 2006, LNCS 4141, 2006. – P. 206–214.
2. Change detection benchmark web site [Electronic resource]. – Mode of access: <http://wordpress-jodoin.dmi.usherb.ca>. – Title from the screen.
3. Low Complexity Change Detection Algorithm Operating in the Compressed Domain / J. Bracamonte, M. Ansorge, F. Pellandini, P. Farine // Proceedings of the 8th COST 276 Workshop on Information and Knowledge Management for Integrated Media Communication. – 2005. – Vol. 276 (8). – P. 7–12.
4. Rajkumar R. Survey on Motion Vector Filtering and Object Segmentation Methods in compressed Domain / R. Rajkumar, D. SaiKrishna, A. S. Jayanth // International Journal of Advancements in Technology. – April 2011. – Vol. 2 (2).
5. Tommesani Stefano. Comparing background subtraction algorithms [Electronic resource] / Stefano Tommesani – Mode of access: <http://www.tommesani.com/index.php/video/comparing-background-subtraction-algorithms.html>. – Title from the screen.
6. Scene change detection techniques for video database systems / H. Jiang, A. Helal, A. Elmagarmid, A. Joshi // Multimedia System. – June 1998. – P. 186–195.
7. Sorwar Golam. A Novel Filter for Block – Based Object Motion Estimation / Golam Sorwar, Manzur Murshed, Laurence Dooley // Digital Image Computing Techniques and Applications. – Melbourne, Australia, 2002.

O. Buchko, R. Savchenko, D. Yakovenko

REAL TIME MOTION DETECTION IN COMPRESSED VIDEO STREAM

This article presents a fast, efficient and simple method for image change and motion detection in the compressed video stream. The proposed method works directly on H.264 compressed video format.

Keywords: motion detection, H.264, motion vectors.

Матеріал надійшов 05.05.2014

УДК 519.85

Горборуков В. В., Франчук О. В.

ЗАДАЧА ПЛАНУВАННЯ ТЕХНІЧНОГО ОБСЛУГОВУВАННЯ СКЛАДНИХ СИСТЕМ

У статті розглянуто задачу планування технічного обслуговування складних систем, яка формалізується в класі моделей дискретного програмування з обмеженнями комбінаторного типу. Для розв'язку задачі запропоновано алгоритм, що базується на ідеології методу динамічного програмування.

Ключові слова: технічна система, математична модель, оптимізація, дискретне програмування, динамічне програмування.

При експлуатації складних систем (територіально-розподілених систем, мереж зв'язку [11], інформаційно-обчислювальних мереж [12] та ін.) для продовження термінів їхньої працездатності, враховуючи неминучі процеси «старіння» систем та зносу їхніх підсистем, застосовуються

гнучкі стратегії технічного обслуговування (ТО) [1; 2; 8]. При використанні гнучких стратегій проведення та обсяги ТО визначаються фактичним технічним станом елементів, що обслуговуються. За рахунок цього зменшується надмірність операцій ТО, знижуються експлуа-