

СИСТЕМА ВІДСТЕЖЕННЯ РУХОМИХ ОБ'ЄКТІВ З ВИКОРИСТАННЯМ МІКРОКОМП'ЮТЕРА ТА БІБЛІОТЕКИ OPENCV

Могильний С.Б., Миніч М.А.

*Інститут телекомунікаційних систем КПІ ім. Ігоря Сікорського, Україна
E-mail: isearch@ukr.net, marinka43753@gmail.com*

Monitoring system for moving objects using Raspberry PI and library OpenCV

The system under consideration allows you to process data on a client device and only transfer useful information to the user. An analysis of the algorithms belonging to the OpenCV library used to track objects using the Raspberry PI.

Розглянута система дозволяє обробляти дані на клієнтському пристрої і лише передавати корисну інформацію користувачеві. Аналіз алгоритмів, що належать бібліотеці OpenCV для відстеження об'єктів за допомогою Raspberry PI.

Сьогоднішній попит на використання пристроїв, пов'язаних між собою мережею, спровокував появу такого поняття як Інтернет речей (IoT – Internet of Things). Можливість віддалено керувати пристроями приваблює, адже така технологія дає змогу раціонально використовувати свій час та контролювати процеси. Наприклад, в розумному будинку керувати кліматом, побутовими пристроями, освітленням та безпекою приміщень.

Уся система IoT потребує передачі даних, як невеликих об'ємів (температура, вологість, загазованість), так і достатньо великих, таких як зображення у системах відео спостереження. Передача відеопотоку або кадрів вимагає різкого збільшення пропускної здатності каналів передачі даних. Хоча часто більшість кадрів не містить корисну інформацію для кінцевого користувача. Тому, щоб уникнути надмірного завантаження каналів передачі даних, пропонується виконувати попередню обробку зображення на клієнтському пристрої. В даному випадку, клієнтський пристрій – це система з вбудованим мікрокомп'ютером. До мікрокомп'ютера приєднана відеокамера, що захоплює (формує) зображення в той момент, коли об'єкт знаходиться близько до центру кадру і, повертаючись, надалі відстежує місцеположення рухливого об'єкту, тримаючи його в центрі кадру. Такий підхід дозволяє отримувати зображення з максимальною корисною інформацією про об'єкт і відповідно, зменшується кількість кадрів для передачі через каналу зв'язку.

Табл. 1 Порівняльна характеристика алгоритмів стеження в бібліотеці OpenCv

<i>Назва алгоритму</i>	<i>Опис</i>	<i>Переваги</i>	<i>Недоліки</i>
Boosting	На основі онлайн версії AdaBoost. Класифікатор навчається на позитивних та негативних прикладах, проте не враховує суміжні області, а вважає їх фоном.	Відстеження у реальному часі.	Продуктивність середня, не надійний в моменти втрати об'єкту.
MIL	Поточне місце розташування об'єкта та зона навколо нього сприймаються, як колекція позитивних прикладів, тому є вірогідність, що принаймні одне зображення з колекції буде відцентровано.	Відновлюється під час часткового перекривання.	Не надійний при повідомленні помилки відстеження, не відновлюється після повного перекривання.
KCF	Поєднано властивості MIL та Boosting, використовується перетворення Фур'є, щоб підраховувати відгук за одну згортку.	Точність і швидкість краще ніж у MIL та Boosting	Не відновлюється під час повного перекривання.
TLD	Трекер слідує за об'єктом від кадру до кадру. Детектор локалізує всі зовнішності об'єкта, і у разі необхідності коригує трекер.	Працює зі змінами в масштабі та за перекриванням на багатьох кадрах.	Не працює у випадку повного обертання поза площиною.
MEDIANFLOW	Відстежує об'єкт як в прямому, так і в зворотньому напрямках і часі та вимірює розбіжності між цими двома траєкторіями.	Звітує про відсутність відстеження. Переважно для об'єктів, що не перекриваються.	Непридатний для відстеження тривалого руху
GOTURN	Базується на нейромережі	Надійний при зміні освітлення та деформації об'єкта	Додатково необхідно завантажувати модель Caffe та proto.txt
MOSSE	Використовує адаптивну кореляцію для відстеження об'єктів, яка виробляє стабільні кореляційні фільтри при ініціалізації за допомогою одного кадру.	Надійних при змінах в освітленні, масштабі, позі та нежорстких деформаціях. Виявляє перекриття.	В масштабах продуктивності поступається трекерам deep learning.
CSRT	Використання карти просторової надійності для налаштування підтримки фільтра на частину вибраної області кадру для відстеження	Точність у відстеженні, поліпшене відстеження непрямокутних областей.	Порівняно з іншими трекерами працює з меншою кількістю кадрів в секунду (25fps)

Одним із важливих етапів створення системи відстеження є вибір алгоритму для стеження за об'єктом в кадрі. Бібліотека з відкритим кодом OpenCV (Open Source Computer Vision) спеціально розроблена для обробки зображень, комп'ютерного зору та алгоритмів загального призначення. Для задачі утримування об'єкта при його частковому перекриванні та стеженні за об'єктом у реальному часі доцільно використовувати саме цю бібліотеку. Так як існує не один алгоритм, а їх набір, необхідно провести порівняльний аналіз та вибрати найбільш відповідний вимогам поставленої задачі.

Мета алгоритму стеження – знайти об'єкт в поточному кадрі, якщо відстеження об'єкту було успішним у попередніх кадрах. Оскільки, об'єкт відстежувався до поточного кадру, то відомі параметри моделі руху. Модель руху – це відомості про розташування та швидкість об'єкта у попередніх кадрах. Якщо нічого іншого про об'єкт невідомо, то на основі моделі руху можна передбачити його нове розташування.

Бібліотека OpenCV містить вісім алгоритмів, що використовуються для стеження за об'єктами: BOOSTING, MIL, KCF, TLD, MEDIANFLOW, GOTURN, MOSSE і CSRT.

Беручи до уваги наведені дані, можна зробити висновок, що ідеального алгоритму сьогодні не існує, тому потрібно чимось жертвувати. Для даної задачі підходить алгоритм TLD, так як він підтримує часткове перекривання об'єктів та здатний вносити корегування до трекера в залежності від умов та масштабу.

Література

1. Zdenek Kalal, Krystian Mikolajczyk, Jiri Matas: Tracking-Learning-Detection.
2. Satya Mallick: Object Tracking using OpenCV.
3. Z. Kalal, et. al, "P-N Learning: Bootstrapping Binary Classifiers by Structural Constraints,"CVPR, 2010.