

УДК 004.932.2

Ю.В. Ладиженський, А.О. Серeda
Донецький національний технічний університет, м.Донецьк
кафедра прикладної математики і інформатики
E-mail: ly@cs.dgtu.donetsk.ua, aas11@bk.ru

ВІДСТЕЖУВАННЯ ОБ'ЄКТІВ У ВІДЕОПОТОЦІ НА ОСНОВІ ВІДСТЕЖУВАННЯ ПЕРЕМІЩЕННЯ ФРАГМЕНТІВ ОБ'ЄКТІВ

Abstract

Ladyzhenskyy Y., Sereda A. Object tracking in video stream based on a tracking of object fragments. The new method and algorithm for object tracing was proposed. An object is considered as a set of its moving fragments. It allows a reliable tracking of partially occluded objects. The software realization of proposed algorithm with real-time performance was created.

Keywords: object tracking, object fragment, partial occlusion, trajectory, background subtraction.

Анотація

Ладиженський Ю.В., Серeda А.О. Відстежування об'єктів у відеопотоці на основі відстежування переміщення фрагментів об'єктів. Запропоновано новий метод і алгоритм відстежування об'єктів. Об'єкт розглядається як множина рухомих фрагментів. Траєкторія об'єкту будується по траєкторіям фрагментів. Це дозволяє надійно відстежувати частково перекриті об'єкти. Створено програмну реалізацію алгоритму для відстежування об'єктів у реальному часі.

Ключові слова: відстежування об'єктів, фрагмент об'єкту, часткове перекриття, траєкторія, віднімання фону.

Аннотация

Ладыженский Ю.В., Серeda А.А. Отслеживание объектов в видеопотоке на основе отслеживания перемещения фрагментов объектов. Предложены новый метод и алгоритм отслеживания объектов. Объект рассматривается как множество движущихся фрагментов. Траектория объекта строится по траекториям фрагментов. Это позволяет надежно отслеживать частично перекрытые объекты. Создана программная реализация алгоритма для отслеживания объектов в реальном времени.

Ключевые слова: отслеживание объектов, фрагмент объекта, частичное перекрытие, траектория, вычитание фона.

Вступ

Необхідність відстежування об'єктів у відеопотоці виникає в багатьох областях, наприклад, у системах безпеки, системах контролю теологічних процесів, при організації людино-машинного інтерфейсу, при створенні систем автоматизованого аналізу спортивних змагань. Під відстежуванням об'єктів у відеопотоці розуміється отримання списку об'єктів і їх координат в кожному кадрі.

Основними проблемами при відстежуванні руху множини об'єктів у відеопотоці є надійне відстежування об'єктів, частково або повністю перекритих іншими об'єктами; об'єктів, форма яких значно змінюється, а також забезпечення високої швидкодії. Можна виділити два підходи до вирішення проблеми відстежування частково перекритих об'єктів.

1) Не виконувати відстежування об'єкту безпосередньо під час перекриття, а запам'ятати можливі відповідності між об'єктами у відеопотоці до і після проміжку часу, де відбувалося перекриття. Відповідність між об'єктами до і після перекриття може бути знайдена шляхом порівняння зображень об'єктів, характеристик їх руху, або за допомогою оператора системи.

2) Виконувати відстежування переміщення об'єктів при їх частковому перекритті. Оскільки ціле зображення об'єкту неможливо відстежити, необхідно відстежувати переміщення окремих частин зображення об'єкту. Цей підхід має перевагу, оскільки може забезпечити меншу кількість помилок при відстежуванні. При цьому не виключається використання додатково методів, використовуваних в першому підході.

Можна використовувати різні визначення мінімальної частини об'єкту, яка підлягає відстежуванню. Цей вибір має вирішальне значення для побудови алгоритму відстежування. Деякі з існуючих підходів:

1) Відстежування однорідних регіонів. У [1] виконується одночасне відстежування і цілих об'єктів, і регіонів, з яких вони складаються. Однорідні регіони отримуються за допомогою кластеризації зображення об'єкту, яка має високу обчислювальну складність.

2) Відстежування переміщень точкових особливостей. У роботі [2] виконується відстежування точкових особливостей для об'єктів, зображення яких перетинаються, а вибір точкових особливостей, що підлягають відстежуванню, здійснюється перед моментом перетину зображень об'єктів.

3) Відстежування контурів. У роботі [3] виявляються випадки перекриття об'єктів і робиться спроба відтворити невидимі частини контурів об'єктів.

4) Відстежування частин зображення цілого об'єкту, не перекритих іншими об'єктами. У роботі [4] виконується відстежування переміщення зображення всього об'єкту. Для цього використовується модель, що постійно оновлюється. У разі перетину об'єктів створюється маска, що показує, які частини об'єкту перекриті іншими об'єктами. Відстежуються переміщення тільки тих частин об'єкту, які не є перекритими.

У якості відстежуваних частин зображень об'єктів можуть бути використані фрагменти зображення довільної форми.

Розроблені алгоритми відстежування переміщення фрагментів зображення, стійкі до викривлень фрагментів [5], які можуть забезпечити відстежування з малим числом помилок і збоїв. При кодуванні відео, зокрема, за стандартами MPEG-2 і MPEG-4, використовується відстежування переміщення прямокутних фрагментів зображення. Розроблені швидкі алгоритми і спеціалізовані процесори, призначені для відстежування переміщень прямокутних фрагментів [6], які можуть бути модифіковані і використані при відстежуванні переміщення об'єктів, що може забезпечити високу швидкодію.

У цій статті запропоновано новий метод і алгоритм відстежування об'єктів, який базується виключно на відстежуванні переміщення їх фрагментів.

Загальний опис методу

Кожен рухомий об'єкт розглядається як сукупність рухомих фрагментів. Кожен фрагмент є областю кадру довільного розміру і форми, що містить частину об'єкту або цілий об'єкт. Відстежування переміщень об'єктів може бути зведене до двох основних кроків:

1) Відстежування переміщення всіх рухомих фрагментів у відеопотоці.

2) Побудова траєкторій руху об'єктів на основі результатів відстежування переміщення їх фрагментів. Групи близько розташованих фрагментів, які рухаються узгоджено, об'єднуються в об'єкти.

Запропонований підхід дозволяє:

1) Виконувати відстежування об'єктів, які частково перекриті іншими об'єктами. Для цього достатньо успішно відстежити переміщення хоча б одного фрагмента об'єкту. По траєкторії цього фрагмента можливо відтворити траєкторію всього об'єкту.

2) Виконувати відстежування множини об'єктів, які протягом всього часу свого існування є частково перекритими і утворюють одну "пляму". Таку групу об'єктів може бути сегментовано на окремі об'єкти за траєкторіями руху їх частин.

3) Виконувати відстежування об'єктів, що сильно змінюють свою форму. Для цього достатньо, щоб у об'єкту існувала видима частина, яка міняє свою форму відносно слабо і може бути відстежена.

- 4) Створити ефективну реалізацію на багатопроцесорних обчислювальних системах.
- 5) Створити ефективну апаратну реалізацію, яка може забезпечити високу швидкодію.

Знаходження об'єктів і їх координат на основі траєкторій фрагментів можна виконати один раз після завершення відстежування фрагментів у всьому відеозапису, або виконувати в кожному кадрі після відстежування фрагментів в цьому кадрі. У першому випадку, маючи більше інформації, можна допустити менше помилок. Проте, у багатьох випадках необхідне отримання координат об'єктів у реальному часі. У кожен момент часу в пам'яті зберігаються усі об'єкти, фрагменти і їх координати за останні T_{window} секунд. З кожним новим кадром відбувається отримання нових координат і, якщо це необхідно, уточнення одержаних раніше. Таким чином, запропонований підхід дозволяє отримувати координати як у реальному часі, так і після закінчення відстежування.

Дії, що відбуваються при аналізі одного кадру (загальний крок алгоритму), представлені на рис. 1. Овалами показані дані, прямокутниками — виконувані дії, стрілками — потоки даних.

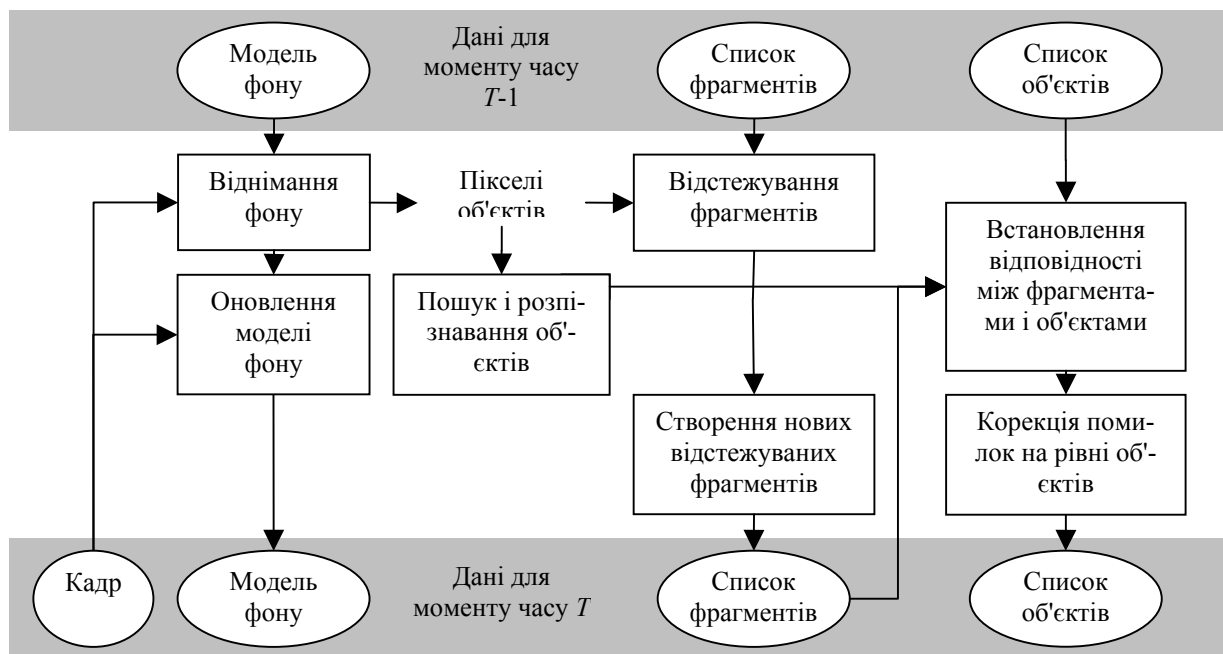


Рисунок 1 — Загальний крок алгоритму відстежування об'єктів

Віднімання фону і моделювання фону

Віднімання фону, побудова і оновлення моделі фону є поширеним підходом до виділення рухомих об'єктів у відеопотоці [1]. Це дозволяє виділити в кожному кадрі пікселі, які відрізняються від статичного фону і належать рухомих об'єктам.

Пошук і розпізнавання об'єктів

Групуючи фрагменти, які рухаються узгоджено, запропонований алгоритм може виділяти об'єкти довільного виду. Тому пошук і розпізнавання об'єктів в кадрах не є обов'язковим. Для утрупання фрагментів, що мають схожі траєкторії, необхідно, щоб вони рухалися протягом деякого часу. Через це об'єкти, що щойно з'явилися, не можуть бути виділені відразу. Пошук і розпізнавання об'єкту, що щойно з'явився, дозволить об'єднати всі його фрагменти відразу після появи об'єкту.

Визначення фрагмента і об'єкту

Відстежуваний фрагмент f_k характеризується наступними властивостями:

- 1) $Width_k$, $Height_k$ — ширина і висота прямокутника, що містить фрагмент.

2) I_k — матриця розміром $Height_k \times Width_k$, що містить пікселі фрагмента. Значення пікселів, що не відносяться до фрагмента, можуть бути довільними.

3) M_k — матриця розміром $Height_k \times Width_k$, яка показує, які пікселі з I_k і з якою впевненістю відносяться до даного фрагменту, $m_{kij} \in [0,1]$.

4) $[T_k^{\min}, T_k^{\max}]$ — проміжок часу, в якому відомі координати даного фрагмента, тобто його час існування.

5) $X_k(t), Y_k(t)$ — координати лівого верхнього кута фрагмента у момент часу $t \in [T_k^{\min}, T_k^{\max}]$.

6) Id_k — дорівнює унікальному ідентифікатору об'єкту, до якого відноситься даний фрагмент, або особливому значенню *None*, якщо фрагмент не відноситься до жодного об'єкту.

Різні фрагменти можуть перетинатися. Більшість пікселів об'єктів відносяться як мінімум до одного фрагмента.

В процесі відстежування відбувається оновлення матриць I і M .

Представлення за допомогою матриць I і M аналогічно використовуваному в [4], але множина фрагментів дозволяє краще відстежувати об'єкти із змінною формою, ніж використовується в [4] одна модель для всього об'єкту.

Об'єкт o_k характеризується наступними властивостями:

1) $[OT_k^{\min}, OT_k^{\max}]$ — проміжок часу існування об'єкту.

2) $OX_k(t), OY_k(t)$ — координати центру об'єкту у момент часу $t \in [OT_k^{\min}, OT_k^{\max}]$.

3) OId_k — унікальний ідентифікатор даного об'єкту.

Позначимо через $Fragments(o)$ множину всіх фрагментів, що відносяться до об'єкту o . Значущістю об'єкту назвемо сумарний час існування всіх його фрагментів. Нехай $CommonTime(f_i, f_j) = \max(0, \min(T_i^{\max}, T_j^{\max}) - \max(T_i^{\min}, T_j^{\min}))$ Позначимо через $Scale(f, t)$ відношення розміру зображення об'єкту в пікселях до розміру того ж самого об'єкту в метрах у точці кадру, в якій знаходиться фрагмент f у момент часу t . Аналогічно визначимо $CommonTime$ і $Scale$ для об'єктів.

Відстежування переміщення фрагментів

Для відстежування переміщення фрагментів можна застосовувати швидкі алгоритми пошуку прямокутних фрагментів, розроблені для кодування відео [6]. Відстежування переміщень фрагментів при відстежуванні об'єктів має дві особливості:

1) Фрагмент може мати довільну форму.

2) Необхідно оцінювати надійність відстежування (ймовірність того, що відстеження переміщення даного фрагменту відбулося вірно).

Нехай V — область пошуку фрагмента f_k у кадрі A .

Розглянемо клас алгоритмів, які в процесі відстежування обчислюють різницю між фрагментом f_k і областю кадру A за формулою $D(x, y) = \sum_{dx=0}^{Width_k-1} \sum_{dy=0}^{Height_k-1} F(a_{x+dx, y+dy}, i_{k, dx, dy})$ і

визначають координати фрагменту кадру A , найбільш схожого на фрагмент f_k як $(x_{\min}, y_{\min}) = \arg \min_{(x, y) \in V} D(x, y)$, де F — деяка функція (наприклад, модуль різниці яскравостей).

Такі алгоритми можуть бути легко модифіковані для відстежування фрагментів довіль-

ної форми: $D(x, y) = \sum_{dx=0}^{Width_k-1} \sum_{dy=0}^{Height_k-1} F(a_{x+dx, y+dy}, i_{k, dx, dy}) m_{k, dx, dy}$.

Для визначення надійності відстежування фрагмента використовуються наступні критерії:

1) Вибраний фрагмент повинен бути достатньо схожий на фрагмент f_k :

$$D(x_{\min}, y_{\min}) \leq D_{\max}.$$

2) Усі інші фрагменти з області пошуку V повинні бути достатньо несхожі на шуканий фрагмент f_k : $D_{\min} = \min_{(x,y) \in V, |(x,y), (x_{\min}, y_{\min})| \geq r} D(x, y)$,

$$D_{\min} \geq sD(x_{\min}, y_{\min}) \quad (1)$$

де D_{\max} , r і s — константи.

Назвемо коефіцієнтом упевненості у відстеженні величину

$$K_{conf} = \max \left(0, \min \left(1, k_1 \left(\frac{D_{\max}}{D(x_{\min}, y_{\min})} - 1 \right), k_2 \left(\frac{D_{\min}}{sD(x_{\min}, y_{\min})} - 1 \right) \right) \right), \quad (2)$$

де $k_1, k_2 > 0$ — константи. Фрагмент вважається відстеженим, якщо $K_{conf} > 0$.

Для перевірки умови (1) необхідно виконати порівняння всіх фрагментів з області пошуку V з фрагментом f_k . При цьому відбуваються практично ті ж самі дії, що і при пошуку фрагмента методом повного перебору всіх його можливих положень, що має велику часову складність. Доцільно виконувати перевірку (1) один раз в декілька кадрів.

Передбачення найбільш вірогідного положення фрагмента і області, в якій він може знаходитися здійснюється на підставі відомих координат фрагмента або об'єкту, до якого він належить, а також заданих максимальної швидкості та прискорення.

Відстежування фрагмента припиняється, якщо його не вдалося знайти в $T_{fragment}$ послідовних кадрів. Якщо фрагмент був втрачений, а потім знайдений, його невідомі координати інтерполюються по відомих.

Створення нових відстежуваних фрагментів

Один раз в декілька кадрів виконується пошук всіх пікселів, які не відносяться ні до фону, ні до наявних фрагментів. Створюються нові відстежувані фрагменти, що покривають всі достатньо великі групи знайдених пікселів. Визначення оптимального положення, розмірів і форми створюваних фрагментів вимагає подальших досліджень.

Встановлення відповідності між фрагментами і об'єктами

Алгоритм встановлення відповідності між фрагментами і об'єктами складається з наступних кроків, які дозволяють обробляти всі ситуації, пов'язані з виникненням, зникненням, злиттям і розділенням видимих об'єктів:

1) Відділення фрагментів від об'єктів (включаючи розділення об'єктів), якщо їх рух не узгоджений.

2) Віднесення фрагментів до існуючих об'єктів, з якими їх рух є узгодженим.

3) Злиття об'єктів, рух яких є узгодженим.

4) Видалення об'єктів, у яких не залишилося відстежуваних фрагментів.

5) Створення нових об'єктів із фрагментів, що не відносяться до існуючих об'єктів.

Критерії узгодженості руху

Для перевірки чи можуть два фрагменти, фрагмент і об'єкт, або два об'єкти, які існують одночасно, відноситися до одного реального об'єкту, аналізуються їх траєкторії.

Нехай f_i і f_j — два фрагменти, причому $T_{\min} = \max(T_i^{\min}, T_j^{\min})$, $T_{\max} = \min(T_i^{\max}, T_j^{\max})$ і $T_{\max} \geq T_{\min}$. Нехай $CS(i, j, t_2) = \min(\text{Scale}(f_i, t), \text{Scale}(f_j, t))$.

Запропоновано наступні критерії узгодженості руху фрагментів f_i і f_j :

Впродовж всього руху фрагменти повинні знаходитися достатньо близько:
 $\forall t \in [T_{\min}, T_{\max}]$:

$$\max(X_i(t) + Width_i, X_j(t) + Width_j) - \min(X_i(t), X_j(t)) \leq MaxObjectWidth \cdot CS(i, j, t),$$

$$\max(X_i(t) + Width_i, X_j(t) + Width_j) - \min(X_i(t), X_j(t)) \leq MaxObjectWidth \cdot CS(i, j, t).$$

1) Взаємне розташування фрагментів не повинне сильно змінюватися:

$$\forall t_1, t_2 \in [T_{min}, T_{max}]: \left| \left(\frac{X_i(t_1) - X_j(t_1)}{CS(i, j, t_1)}, \frac{Y_i(t_1) - Y_j(t_1)}{CS(i, j, t_1)} \right), \left(\frac{X_i(t_2) - X_j(t_2)}{CS(i, j, t_2)}, \frac{Y_i(t_2) - Y_j(t_2)}{CS(i, j, t_2)} \right) \right| \leq D_1.$$

2) Швидкість фрагментів не повинна сильно відрізнятися:

$$\forall t \in [T_{min}, T_{max} - t']:$$

$$\left| (X_i(t+t') - X_i(t), Y_i(t+t') - Y_i(t)), (X_j(t+t') - X_j(t), Y_j(t+t') - Y_j(t)) \right| \leq D_2 \cdot CS(i, j, t), \text{ де } t' \text{ — константа.}$$

Позначимо через $Matched(f_i, f_j)$ виконання трьох приведених критеріїв для фрагментів f_i і f_j . Аналогічні критерії використовуються і при перевірці узгодженості руху фрагмента і об'єкту або двох об'єктів. При цьому можуть бути використані інші значення порогів D_1 і D_2 .

Всі запропоновані критерії є обов'язковими, але не достатніми.

Побудова траєкторії руху об'єкту

Траєкторія руху будується на базі траєкторій фрагментів, що відносяться до об'єкту, а також положень об'єкту в окремих кадрах, де він був розпізнаний.

Введемо наступні позначення:

$$Displacement(f_i, t) = \begin{cases} (X_i(t) - X_i(t-1), Y_i(t) - Y_i(t-1)), & t \in [T_i^{min} + 1, T_i^{max}] \\ (0, 0), & t \notin [T_i^{min} + 1, T_i^{max}] \end{cases}$$

$GoodFragments(o, t)$ — підмножина фрагментів $\{f_i : f_i \in Fragments(o) \& Displacement(f_i, t) \neq (0, 0)\}$, що мають високі коефіцієнти упевненості у відстежуванні у момент часу t (2) і великий час існування;

$$Displacement(o, t_1, t_2) = \sum_{t=t_1}^{t_2} \sum_{f_i \in GoodFragments(o, t)} Displacement(f_i, t).$$

Нехай невідомі координати об'єкту o_i у момент часу t .

1) Якщо відомі координати об'єкту o_i у момент часу $t_1 < t$, але не відомі в подальші моменти, то $(OX_i(t), OY_i(t)) = (OX_i(t_1), OY_i(t_1)) + Displacement(o_i, t_1 + 1, t)$.

2) Якщо відомі координати об'єкту o_i у момент часу $t_2 > t$, але не відомі в попередні моменти, то $(OX_i(t), OY_i(t)) = (OX_i(t_2), OY_i(t_2)) - Displacement(o_i, t + 1, t_2)$.

3) Якщо відомі координати об'єкту o_i у моменти часу $t_1 < t$ і $t_2 < t$ то

$$(OX_i(t), OY_i(t)) = \frac{t_2 - t}{t_2 - t_1} (OX_i(t_1), OY_i(t_1)) + Displacement(o_i, t_1 + 1, t) + \frac{t - t_1}{t_2 - t_1} ((OX_i(t_2), OY_i(t_2)) - Displacement(o_i, t_1 + 1, t_2))$$

Якщо координати об'єкту не відомі в усі моменти часу, то вони визначаються для одного будь-якого моменту із прямокутника, що обмежує фрагменти даного об'єкту, а для решти моментів часу визначаються згідно з пунктами 1 і 2.

Перерахунок координат об'єкту відбувається при кожній зміні в списку його фрагментів на тому інтервалі часу, який зачіпається даною зміною.

Відділення фрагментів від об'єкту

Якщо для деякого об'єкту o існує такий фрагмент $f \in Fragments(o)$, що $\neg Matched(f, o)$, то фрагмент f відділяється від об'єкту o і перестає відноситися до жодного об'єкту.

Якщо в деякий момент часу від об'єкту відділилася група фрагментів, відношення сумарної площі яких до сумарної площі фрагментів, що залишилися, перевищує порогове значення, то вони негайно об'єднуються в новий об'єкт і вважається, що початковий об'єкт розділився на два об'єкти.

Віднесення фрагмента до існуючого об'єкту

Віднесення фрагмента до існуючого об'єкту відбувається в двох випадках:

1) Якщо існують такі фрагмент f_i і об'єкт o , що $Id_i = None$, $CommonTime(f_i, o) \geq T_{min}$ і $Matched(f_i, o)$, то відбувається віднесення фрагмента f_i до об'єкту o . T_{min} — порогове значення. Якщо фрагмент може бути віднесений відразу до декількох об'єктів, то відбувається його віднесення до об'єкту з максимальною значущістю.

2) При успішному розпізнаванні об'єкту до нього відносяться всі фрагменти, які перетинаються з його зображенням і не перетинаються із зображеннями інших об'єктів.

Злиття двох об'єктів

Два об'єкти o_1 і o_2 повинні бути об'єднані в один коли виконуються наступні умови:

1) Інтервали часу їх існування перетинаються.

2) Якщо o — об'єкт, утворений з множини фрагментів $F = Fragments(o_1) \cup Fragments(o_2)$ то $\forall f \in F$ виконується $Matched(f, o)$.

Траєкторія об'єкту o обчислюється на основі траєкторій фрагментів так само, як і для будь-якого іншого об'єкту.

Створення нового об'єкту

Новий об'єкт створюється із одного або декількох фрагментів, що не відносяться до об'єктів, в двох випадках:

1) Якщо $Id_i = None$ і $T_i^{max} - T_i^{min} \geq T_{threshold}$, то на основі фрагмента f_i створюється новий об'єкт. Після цього робиться спроба віднести до нового об'єкту всі інші фрагменти, що не відносяться до об'єктів.

2) Якщо в деякому місці кадру було знайдене і розпізнане зображення об'єкту заданого типу, причому всі фрагменти, які перетинаються з цим зображенням, не віднесені до об'єктів, то з цих фрагментів створюється новий об'єкт.

Корекція помилок на рівні об'єктів

Корекція помилок на рівні об'єктів може містити ряд правил, що використовують форму, колір і траєкторію об'єктів, і дозволяють виправляти деякі помилки, допущені на попередніх етапах. Реалізовано 2 правила:

1) Якщо деякий об'єкт не вдалося відстежити, а поряд з місцем його останнього розташування був знайдений новий об'єкт, то ці два об'єкти об'єднуються в один. Це правило дозволяє у багатьох випадках продовжувати відстежувати об'єкти в умовах повного перекриття.

2) Якщо деякий об'єкт, що має вік не менше t_{del} , в кожному кадрі знаходився на відстані не більше ніж r_{del} від об'єкту, значущість якого перевищує значущість даного об'єкту в k_{del} раз, то даний об'єкт видаляється. Це правило дозволяє видаляти помилкові об'єкти, викликані помилками при відстежуванні фрагментів.

Практичні результати

Описаний алгоритм було реалізовано програмно.

Експерименти показали, що розроблений алгоритм може відстежувати об'єкти в умовах часткового перекриття. На рис. 2 приведені фрагменти трьох проаналізованих кадрів з відеозапису футбольного матчу, що містить такі ситуації.



Рисунок 2 — Приклад відстежування об'єктів в умовах часткового перекриття

Середній час повного аналізу одного кадру відеозапису дозволом 670x367 пікселів, що містить зображення дванадцяти футболістів, на комп'ютері з процесором Athlon 64 X2 5400+ із задіяним одним ядром, за умови, що в кожному другому кадрі виконується тільки відстежування фрагментів, дорівнює 31 мс. З них в середньому 18 мс витрачається на відстежування фрагментів і 10 мс витрачається на пошук і розпізнавання людей. Така швидкодія дозволяє обробляти відеозапис футбольного матчу у реальному часі.

Висновки

Для надійного відстежування об'єктів в умовах часткового перекриття потрібно відстежувати переміщення їх частин. У якості відстежуваної частини об'єкту може бути використаний фрагмент зображення довільної форми.

Одержані наступні основні результати:

1) Розроблено новий метод і алгоритм відстежування об'єктів. Його головне достоїнство — можливість виконувати відстежування об'єктів в умовах їх часткового перекриття, зокрема в умовах постійного перекриття.

2) Розроблено програмну модель запропонованого алгоритму.

3) Показано, що запропонований алгоритм має достатню швидкодію для відстежування об'єктів у реальному часі на персональному комп'ютері.

Однією з сфер застосування розробленого алгоритму є автоматизований аналіз спортивних змагань.

Запропонований алгоритм може бути ефективно реалізований на багатопроесорних комп'ютерах. Також можливе застосування спеціалізованих процесорів, розроблених для кодування відео.

Література

1. A. Cavallaro, O. Steiger, T. Ebrahimi. Tracking video objects in cluttered background. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 15, Issue 4, April 2005. URL: http://lts1pc19.epfl.ch/repository/Cavallaro2004_865.pdf
2. P. S. Kumar, P. Guha A. Mukerjee. Colour and Feature Based Multiple Object Tracking Under Heavy Occlusions. Computer Vision Group, IIT Kanpur. URL: http://www.cse.iitk.ac.in/research/pdfs/ddmtech/sateesh-guha-mukerjee-icap07_colour-feature-multi-obj-tracking-occlusion.pdf
3. Alper Yilmaz, Xin Li, Mubarak Shah. Contour-Based Object Tracking with Occlusion Handling in Video Acquired Using Mobile Cameras. IEEE Transactions on Pattern Analysis and Machine Intelligence. Volume 26, Issue 11 (November 2004) URL: http://www.cs.ucf.edu/~vision/papers/yilmaz_pami_2004.pdf
4. R. Cucchiara, C. Grana, G. Tardini. Track-based and object-based occlusion for people tracking refinement in indoor surveillance. Proceedings of the ACM 2nd international workshop on Video surveillance & sensor networks 2004, New York, NY, USA, October 15, 2004. URL: <http://doi.acm.org/10.1145/1026799.1026814>
5. D. Comaniciu, V. Ramesh, P. Meer. Kernel-Based Object Tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence Volume 25, Issue 5 (May 2003). URL: <http://portal.acm.org/citation.cfm?id=776799&dl=>
6. Adaptive Motion Estimation Processor for Autonomous Video Devices. T. Dias, S. Momcilovic, N. Roma, L. Sousa. EURASIP Journal on Embedded Systems, vol. 2007, Article ID 57234, 10 pages, 2007. URL: <http://www.hindawi.com/GetArticle.aspx?doi=10.1155/2007/57234>

Здано в редакцію:
12.03.2009р.

Рекомендовано до друку:
д.т.н, проф. Башков Є.О.