

ВІДСТЕЖЕННЯ ПЕРІОДИЧНОГО РУХУ ОБ'ЄКТІВ ЗІ СТАТИЧНОЇ КАМЕРИ НА ПРИКЛАДІ ВІДЕОІГОР

А. Д. Павлюк^{1, а}, В. М. Кригін^{1,2}, В. М. Ткач¹

¹ Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»,
Фізико-технічний інститут

² Міжнародний науково-навчальний центр інформаційних технологій та систем
НАН України та МОН України

Анотація

В роботі розглянуто спосіб вилучення анімацій об'єктів на відео зі статичним фоном на прикладі двовимірних спрайтових відеоігор. Для цього було використано наступні алгоритми: пошук максимального потоку для знешумлення бінарного зображення, швидкі перетворення Фур'є для грубого вирівнювання отриманих зображень та семплювання по Гіббзу для пошуку оптичного потоку.

Ключові слова: оптичний потік, максимальний потік, імітація відпалу, семплювання по Гіббзу, розпізнавання образів

Вступ

Відеоігри та кіберспорт є одними з багатьох популярних у всьому світі явищ. Наразі існує велика кількість відеозаписів старих турнірів досить низької якості і інколи немає іншої можливості покращити їх, окрім як повністю перемалювати. В даній роботі представлено спосіб виокремлення зображень ігрових об'єктів для подальшої кластеризації і заміни. Слід зазначити, що представлений алгоритм можна використовувати для довільних відеозаписів зі статичним фоном і об'єктами, що рухаються на ньому.

1. Виокремлення об'єктів на відео

Поступово розбиваючи відео на послідовність кадрів, на кожному кроці, починаючи з другого, маємо два зображення розміру $h \times w$: $F \in C^{h \times w}$ і $S \in C^{h \times w}$, де C — множина кольорів у форматі RGB . Нехай D — бінарне зображення розміру $h \times w$, таке, що $\forall p \in [1, h] \times [1, w]$:

$$D_p = \begin{cases} 1, & \|F_p - S_p\| \leq \alpha, \\ 0, & \|F_p - S_p\| > \alpha, \end{cases}$$

де α — деякий коефіцієнт, зазвичай в межах 5–30, що виникає внаслідок шуму на відео.

На отриманому бінарному зображенні можемо побачити деякі зашумлені області через те, що зазвичай рух об'єкту не означає зміну кольору кожного пікселя в ньому. Для отримання потрібних нам зображень виникає необхідність модифікувати D .

Задачу знешумлення бінарного зображення можна представити у вигляді задачі дискретної оптимізації

$$\sum_{p \in P} \mathbb{1}\{D_p \neq k_p\} + \sum_{pp' \in N_p} \alpha \cdot \mathbb{1}\{k_p \neq k_{p'}\} \rightarrow \min_{k: P \rightarrow \{0,1\}}.$$

Відомо [1, 2, 3, 4], що така задача зводиться до задачі пошуку мінімального розрізу на графі і має ефективний розв'язок.



Рис. 1. Приклад роботи алгоритма пошуку мінімального розрізу. Оригінал відео: [5]

На рисунку 1 бачимо об'єкт, що рухається, початкове бінарне зображення D та розв'язок задачі видалення бінарного шуму.

2. Оптимальне взаємне розташування зображень

На попередньому кроці ми отримали деяку кількість різних зображень різних об'єктів. Для виконання наступних кроків необхідно визначити їх оптимальне взаємне розташування.

Нехай є два зображення $F = C^{h_1 \times w_1}$ і $S = C^{h_2 \times w_2}$. Треба знайти таку пару (x^*, y^*) , що зміщення зображення F на x^* по горизонталі і на y^* по вертикалі відносно зображення S дасть вирівняне зображення F^* відносно S .

^аpawland7@gmail.com



Рис. 2. Приклади мап згортки. Оригінал відео: [5]

Нехай F' — зображення, що зміщено на таке (x', y') , що

$$\begin{aligned}(x', y') &\in [-h_1, H] \times [-w_1, W], \\ H &= h_1 + h_2 - 1, \\ W &= w_1 + w_2 - 1.\end{aligned}$$

Маємо задачу мінімізації

$$(x^*, y^*) = \operatorname{argmin}_{\substack{x' \in [-h_1, H] \\ y' \in [-w_1, W]}} \sum_{1 \leq i \leq h_2, 1 \leq j \leq w_2} (F'_{i+x', j+y'} - S_{i,j})^2.$$

Спростуючи це рівняння і спираючись на те, що за будь-якого зсуву F сума квадратів кольорів кожної з картинок залишається незмінною (вважаємо, що за межами S всі пікселі чорні), можемо перейти до задачі максимізації кореляції двох зображень

$$(x^*, y^*) = \operatorname{argmax}_{\substack{x' \in [-h_1, H] \\ y' \in [-w_1, W]}} \sum_{1 \leq i \leq h_2, 1 \leq j \leq w_2} F'_{i+x', j+y'} \cdot S_{i,j}.$$

На рисунку 2 бачимо зображення F , два можливих S та мапи згортки для них. На першій мапі видно максимум. Його координати і стануть зсувом (x^*, y^*) для F . Отриману задачу максимізації можна вирішити більш ефективно, використовуючи швидкий перетворення Фур'є [6, 7].

За великої кількості картинок у випадку пошуку взаємного розташування «всіх зі всіма» прямі перетворення Фур'є можна обчислювати один раз для кожного окремого зображення.

3. Пошук оптичного потоку

Після отримання зображень з оптимальним взаємним розташуванням можемо перейти до пошуку оптичного потоку між парами зображень. Для цього розглянемо алгоритм імітації відпау з розподілом Гіббзу. Нехай ϵ випадкове поле на множині пікселів P , структурою сусідства другого порядку $\tau \subset P \times P$, множиною міток K , унарною ваговою функцією $q : P \times K \rightarrow \mathbb{R}$, бінарною ваговою функцією $g : K^2 \rightarrow \mathbb{R}$ та ймовірністю

$$\mathbb{P}(k) = \frac{1}{Z(n)} \exp \left\{ \frac{\sum_{p \in P} q(p, k_p) + \sum_{pp' \in \tau} g(k_p, k_{p'})}{T(n)} \right\},$$

де $Z(n)$ — нормуючий множник, а $T(n)$ — температура системи, що зменшується з часом n . Позначимо множину всіх сусідніх до p пікселів $N_p \subset P$. Наведене поле Гіббза має властивість марковості, тобто

$$\mathbb{P}(k_p | k_{p'}, p' \in P \setminus \{p\}) = \mathbb{P}(k_p | k_{p'}, p' \in N_p).$$

Тому, якщо зафіксувати значення всіх $k_{p'}, p' \neq p$, то ймовірність

$$\mathbb{P}(k_p | k_{p'}, p' \in N_p) = \frac{f(k_p)}{h(k_{p_i}; p_i \in N_p)},$$

де

$$f(k_p) = \exp \left\{ \frac{q(p, k_p) + \sum_{p' \in N_p} g(k_p, k_{p'})}{T(n)} \right\},$$

$$h(k_{p_i}; p_i \in N_p) = \sum_{k \in K} \exp \left\{ \frac{q(p, k) + \sum_{p' \in N_p} g(k, k_{p'})}{T(n)} \right\}.$$

Величина $\mathbb{P}(k_{p_i}; p_i \in N_p)$ для фіксованого p буде однаковою для всіх можливих зсувів і не впливатиме на знаходження оцінки найбільшої правдоподібності, тому її можна прибрати як константу.

В даній роботі P — множина пікселів на зображенні, $K = \{k : l_\infty(k) \leq r\}$ — множина таких $k_p = [x_p, y_p]$, що x_p — можливий зсув p по вісі x , а y_p — по вісі y , l_∞ — норма Чебишова. Унарні ваги рахуються за формулою

$$q(p, k_p) = -\theta \cdot \|F(p + k_p) - S(p)\|,$$

де $F(p)$ і $S(p)$ — кольори у пікселі p на першому і другому зображеннях відповідно, а θ — деякий коефіцієнт важливості для q . Бінарні ваги обчислюються за формулою

$$g(k_p, k_{p'}) = -\alpha \cdot \mathbb{1} \{\|k_p - k_{p'}\| > \beta\},$$

де α — коефіцієнт згладжування, а β — допустиме відхилення (було взято $\alpha = 20$ і $\beta = 0$). Початкова температура $T = 10$ зменшується за законом $T(n) = 10 \cdot 0.9^n$, де n — номер ітерації ($n_{max} = 10$). Замість того, щоб рахувати $p(k_p | k_{p'}, p' \in N_p)$ в кожній можливій точці $k_p \in K$, було прийнято рішення обмежити область пошуку величиною r по всіх напрямках відносно поточного пікселя (в роботі було взято $r = 10$). Це значно покращило час роботи програми.

Алгоритм семплювання наступний:

- 1) для кожного пікселя обираємо довільну мітку та визначаємо кількість ітерацій
$$\forall p \in P : k_p \leftarrow (0, 0),$$

$$n \leftarrow 0,$$

$$n_{max} = 10;$$
- 2) для пікселя p знаходимо всі можливі ймовірності переходів k_p в радіусі r

$$\forall k_p \in K : \mathbb{P}(k_p) = \exp \left\{ \frac{q(p, k_p) + \sum_{p' \in N_p} g(k_p, k_{p'})}{T(n)} \right\};$$

- 3) генеруємо нове k_p з розподілу, що було отримано на попередньому кроці;
- 4) кроки 2–3 виконуємо для всіх пікселів $p \in P$;
- 5) якщо $n < n_{max}$, то $n \leftarrow n + 1$ та повертаємось до кроку 2. Інакше — кінець алгоритму.

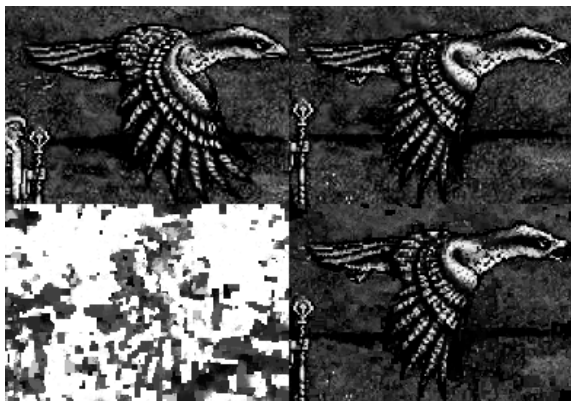


Рис. 3. Приклад пошуку оптичного потоку. Оригінал відео: [5]

На першому кроці немає значення, які початкові мітки обрати, тому що для будь-якого стану наведеного випадкового поля існує послідовність переходів ненульової ймовірності у будь-який інший стан. Порядок обходу пікселів також не має значення.

На рисунку 3 зображено результат роботи алгоритму: перше та друге зображення, отримана мапа зсувів та зображення, яке отримано шляхом зміщення пікселів першого зображення за мапою зсуву. Інтенсивність пікселю на мапі характеризує відстань його зсуву.

Представлений метод пошуку оптичного потоку впорався краще, ніж метод Лукаса-Канаде [8], тому що задача поставлена таким чином, що зсув одного пікселя залежить не тільки від норми різниці кольорів, але і від зсуву всіх інших пікселів. Його можна модифікувати і на його основі розробити алгоритм виокремлення зображень з відео без використання процедури виокремлення об'єктів на відео, описаної в розділі 1, та процедури пошуку оптимального взаємного розташування зображень, описаної в розділі 2.

Висновки та подальша робота

В даній роботі представлено спосіб відстеження та виокремлення об'єктів, що рухаються на відео. Отримані зображення в перспективі можна стилізувати для подальшої заміни [9]. На основі алгоритму пошуку оптичного потоку, представленого в розділі 3, можна розробити технологію створення спрайтів (анімаційних зображень об'єктів) [10] на основі відео не тільки зі статичним фоном, а і з плавним переміщенням камери.

Перелік використаних джерел

1. Savchynskyi B. Discrete Graphical Models — An Optimization Perspective // Foundations and Trends® in Computer Graphics and Vision. — 2019. — Vol. 11, no. 3–4. — P. 160–429. — Access mode: <http://doi.org/10.1561/06000000084>.
2. Boykov Y., Veksler O., Zabih R. Fast approximate energy minimization via graph cuts // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2001. — Vol. 23, no. 11. — P. 1222–1239.
3. Boykov Yuri, Kolmogorov Vladimir. An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision // IEEE Trans. Pattern Anal. Mach. Intell. — 2004. — Sep. — Vol. 26, no. 9. — P. 1124–1137. — Access mode: <https://doi.org/10.1109/TPAMI.2004.60>.
4. Maximum Flows by Incremental Breadth-First Search / Goldberg Andrew, Hed Sagi, Kaplan Haim, Tarjan Robert E., and Werneck Renato // 19th European Symposium on Algorithms (ESA 2011). — Springer Verlag. — 2011. — January. — Access mode: <https://www.microsoft.com/en-us/research/publication/maximum-flows-by-incremental-breadth-first-search/>.
5. zirkoni88: Short gameplay videos of (old) MS-DOS games. DOS Game: Heroes of Might and Magic 2. — Access mode: <https://youtu.be/g42gSMad09Y>.
6. Real-valued fast Fourier transform algorithms / Sorensen H., Jones D., Heideman M., and Burrus C. // IEEE Transactions on Acoustics, Speech, and Signal Processing. — 1987. — Vol. 35, no. 6. — P. 849–863.
7. Winograd Shmuel. On computing the Discrete Fourier Transform // Proceedings of the National Academy of Sciences. — 1976. — Vol. 73, no. 4. — P. 1005–1006. — <https://www.pnas.org/content/73/4/1005.full.pdf>.
8. Barron John, Fleet David, Beauchemin S. Performance Of Optical Flow Techniques // International Journal of Computer Vision. — 1994. — 02. — Vol. 12. — P. 43–77.
9. Kopf Johannes, Lischinski Dani. Depixelizing Pixel Art // ACM Trans. Graph. — 2011. — 07. — Vol. 30. — P. 99.
10. Schödl A., Essa I. Controlled Animation of Video Sprites // ACM/Eurographics Symposium on Computer Animation (SCA) / ACM SIGGRAPH. — San Antonio, TX, USA : ACM Press. — 2002. — August. — Access mode: <http://www.irfanessa.gatech.edu/controlled-animation-of-video-sprites-2/>.