

ТЕХНОЛОГІЯ ВИЗНАЧЕННЯ КЛЮЧОВИХ ТОЧОК НА ОСНОВІ СТАБІЛЬНИХ МЕЖ ОБ'ЄКТІВ

Ф.Г. ГАРАЩЕНКО, А.Ю. КОБЗАР

Анотація. Подано ефективну технологію виділення ключових точок на основі меж об'єктів. Наведено опис програмного забезпечення для синтезу експериментальних даних та процесу перевірки результатів роботи різних методів виділення. Створено алгоритм з максимально стійким результатом та швидкодією, який можна використовувати в режимі реального часу. Завдяки високій стійкості запропонованої технології досягається більш ефективна робота алгоритмів реєстрації на наступних етапах.

Ключові слова: реєстрація зображень, оброблення зображень, пошук меж, пошук ключових точок, фотограмметрія, побудова панорамних знімків.

ВСТУП

Збільшення обчислювальних потужностей в останні роки значною мірою посприяло розвитку алгоритмів оброблення зображень та зробило їх доступними для використання на портативних пристроях, таких як ноутбуки та смартфони. Одним з напрямів оброблення зображень є завдання реєстрації, що полягає в пошуку відповідності між об'єктами, показаними на різних зображеннях. Переважна більшість застосувань методів реєстрації зображень пов'язана з фотознімками, або кадрами відеоряду.

Найбільш очевидним їх використанням є побудова панорамних зображень [1]. На першому етапі програмного комплексу склеювання панорамних знімків здійснюється пошук взаємної орієнтації сусідніх зображень на основі об'єктів, наявних на обох зображеннях одночасно. Цей пошук реалізовується за допомогою алгоритмів реєстрації зображень.

Згадані методики використовуються у фотограмметрії. Під цим терміном розуміють відновлення просторової структури фізичних об'єктів на основі їх фотознімків. Останнім часом розвиваються проекти, які додатково використовують інформацію з камери глибини, зокрема, Google Project Tango [2], GeoCV [3], matterport [4] та ін. [5, 6]. Це дозволяє отримати додаткову інформацію про такі проблемні об'єкти, як дзеркала та інші блискучі предмети. У цьому випадку реєстрація зображень допомагає знайти одні і ті самі об'єкти на знімках з різних камер. Завдяки відомому положенню цих

об'єктів на наступних етапах локалізується взаємне розміщення точок зйомки в просторі.

Типова схема алгоритму реєстрації зображень складається з таких етапів:

- 1) виділення характерних точок на зображеннях;
- 2) побудови числових векторів властивостей (дескрипторів) для знайдених точок;
- 3) зв'язування характерних точок різних зображень у пари на основі їх дескрипторів;
- 4) фільтрації пар ключових точок для зменшення кількості хибних елементів;
- 5) пошуку параметрів моделі взаємного розміщення зображень (для панорами — це 3 кути повороту, для задач фотограмметрії — 3 кути повороту та 2 кути, що описують напрямок від першої точки зйомки до другої).

У цій схемі більшість обчислень припадає на перший етап. Типовий розмір вхідного зображення для алгоритмів реєстрації — 640×480 пікселів — це близько 300 000 точок. Щоб отримати задовільний результат алгоритму виділення характерних елементів, необхідно врахувати значення кожної з точок зображення, тоді як на наступних етапах кількість точок, що залишилися для подальшого оброблення, зменшується більш ніж на два порядки. Важливою функцією цього етапу є їх просторова локалізація (ключові точки обираються на певній відстані одна від одної). Це дозволяє розробити технологію побудови дескрипторів стійкою до незначних зміщень об'єктів.

У проведеному дослідженні було побудовано технологію виділення характерних точок на основі властивостей меж об'єктів, що розташовані на зображенні.

Мета роботи — створення швидкодіючого алгоритму, стійкого до афінних та просторових перетворень. Під мірою стійкості розуміють відсоток ключових точок одного зображення, які збігаються з ключовими точками іншого зображення. Велика частка таких збігів свідчить про якість алгоритму, що виражається у підвищенні точності та швидкості роботи алгоритмічного комплексу на наступних етапах за рахунок більш достовірних вхідних даних.

СИНТЕЗ ЕКСПЕРИМЕНТАЛЬНИХ ДАНИХ

Використання реальних зображень під час розроблення не дозволяє повною мірою оцінити якість отриманого результату. Це зумовлено тим, що під час звичайного фотознімання неможливо відслідкувати зміну положення камери в просторі, а також бракує інформації про співвідношення об'єктів, зображених на знімках. Для отримання такого роду інформації слід використовувати спеціальне обладнання, що також має свої похибки, які необхідно додатково враховувати.

Альтернативним вирішенням цієї проблеми є синтез експериментальних даних за допомогою графічної бібліотеки OpenGL. Використання таких зображень дає змогу отримати повну інформацію про розташування камер під час знімання та про зображені на них об'єкти. Ще однією перевагою та-

кого підходу є відсутність шумів, що дозволяє додати їх у потрібній кількості після формування зображень.

Саме тому в межах цієї роботи використовувалися синтезовані зображення, максимально наближені до реальних фотознімків. Для цього на рельєф реальної місцевості накладалися супутникові знімки. Як джерело знімків взято зображення з публічної кадастрової карти України [7] (рис. 1), а інформацію про висоти місцевості — із Shuttle Radar Topography Mission [8] (рис. 2). Отримані зображення максимально наближені до даних, які надходять з безпілотних літальних апаратів, що набули широкого застосування за останні декілька років. Гірську місцевість обрано через те, що її поверхня має значні відхилення від площини і дозволяє врахувати всі можливі деталі розроблюваного алгоритму.

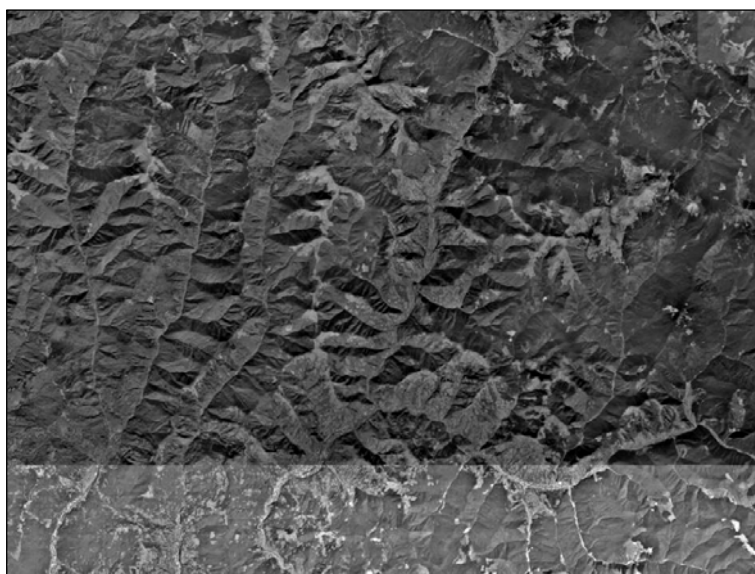


Рис. 1. Супутниковий знімок. Рахівський район Закарпатської області

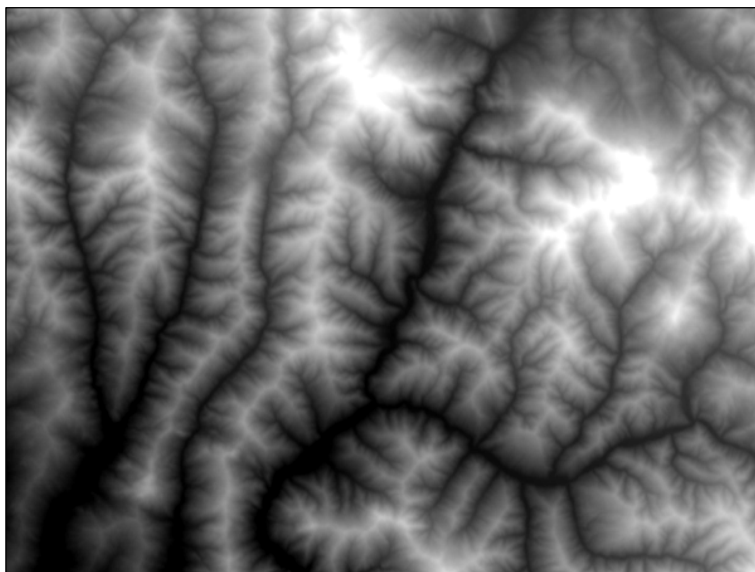


Рис. 2. Рельєф місцевості. Темні ділянки — низини, світлі — підвищення

Програму синтезу розроблено в середовищі Visual Studio Community 2015 з використанням мови програмування C#. Виведення тривимірної графіки здійснювалося засобами бібліотеки OpenGL 3.2.

Результатом роботи програми є серія зображень рельєфу текстурованого супутниковими знімками (рис. 3). До кожного з них додається положення камери в момент генерації зображення та набір відстаней від центра проєкції до всіх видимих точок на рельєфі (рис. 4). Завдяки цим даним можна визначити положення точок у просторі та оцінити точність роботи алгоритмів виділення особливостей.



Рис. 3. Результат проєктування

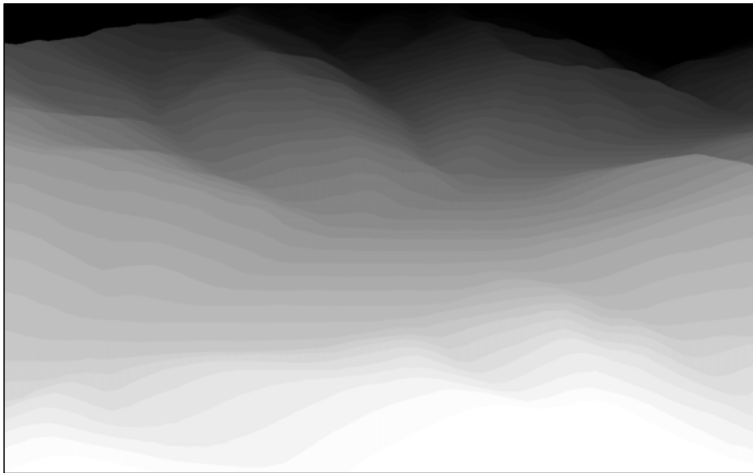


Рис. 4. Відстані від камери до точок, зображених на проєкції місцевості

Можливість перевірки ефективності роботи дозволяє порівняти стійкість запропонованого підходу з уже відомими, такими як SIFT [9] і MSER [10].

ТЕХНОЛОГІЯ ВИДІЛЕННЯ КЛЮЧОВИХ ТОЧОК

На вхід розробленого алгоритму подається зображення, а результатом його роботи є множина ключових точок. Він складається з 8 кроків.

Перший крок. Робота алгоритму починається із завантаження зображення, яке з кольорового зображення переводиться у відтінки сірого (grayscale) таким чином:

$$y = \left\lfloor \frac{77r + 150g + 29b}{256} \right\rfloor,$$

де y — яскравість пікселя у відтінках сірого; r , g , b — насиченість червоним, зеленим та синім кольорами відповідного пікселя вхідного зображення. Їх значення містяться в інтервалі від 0 до 255 включно. Перетворення виконується в цілих числах для збільшення швидкодії алгоритму.

Другий крок. Виконується розмиття зображення гаусіаном з параметром $\sigma = 3$. Ця процедура необхідна для ефективнішого виділення межі на фоні шумів зображення та правильного оброблення різких змін їх орієнтацій. Процес розмиття реалізовується у вигляді згорток зображення ядром гаусіана:

$$f(x_0, x) = e^{-\frac{(x_0 - x)^2}{2\sigma^2}}.$$

Оскільки піксели на зображенні розміщені у вигляді дискретної ґратки, то і згортка буде дискретною. Для збільшення швидкості роботи результат функції f попередньо розраховується для цілих значень $x_0 - x$ в інтервалі від -3σ до 3σ . Ці значення нормуються таким чином, щоб їх сума дорівнювала 2^{16} . Однією з властивостей двовимірної згортки гаусіаном є те, що вона розкладається на композицію двох одновимірних згорток — по вертикалі та по горизонталі. Обчислення згорток ведеться в цілих числах, що є можливим завдяки описаному нормуванню ядра. Урахування кольорів сусідніх точок під час розмиття дозволяє підвищити точність подання яскравостей пікселів зображення з 8 до 16 біт. Такі модифікації позитивно позначаються на якості роботи наступних етапів алгоритму та на швидкості розмиття.

Третій крок. Після розмиття здійснюється пошук градієнтів у кожній точці зображення за формулою

$$G(x, y) = (I(x + 1, y) - I(x - 1, y), I(x, y + 1) - I(x, y - 1)),$$

де $I(x, y)$ — яскравість відповідного пікселя зображення. Якщо параметрами функції G є координати точки, розміщеної біля межі, тоді замість сусідніх точок, що виходять за межі зображення, беруться їх найближчі сусідні точки з поверхні знімка. З урахуванням цього доповнення формула набуває такого вигляду:

$$G(x, y) = \begin{pmatrix} I(\min(x + 1, W - 1), y) - I(\max(x - 1, 0), y) \\ I(x, \min(y + 1, H - 1)) - I(x, \max(y - 1, 0)) \end{pmatrix}^T,$$

де W і H — ширина та висота зображення відповідно, а нумерація пікселів починається з позиції $(0, 0)$. Знайдені значення градієнтів зменшують у 2^k , $k \geq 0$ рази, де k обирається таким чином, щоб максимальне значення градієнта за модулем було меншим ніж 2^{10} . Градієнти безпосередньо

позв'язані з межами об'єктів зображення і надалі використовуватимуться лише вони. Це автоматично робить запропонований алгоритм стійким до змін яскравості на знімках.

Четвертий крок. На завершальній стадії алгоритму (фільтрації знайдених ключових точок) знадобляться середні значення величин градієнтів в околі кожної такої точки. Тому після знаходження градієнтів розраховуються значення їх модулів у кожній точці зображення та проводиться їх розмиття. Щоб уникнути тривалих розрахунків процедури розмиття, розмір знайденого масиву зі значеннями градієнтів зменшують у 2 рази по вертикалі та 2 рази по горизонталі. Цей крок пришвидшує розмиття за рахунок зменшення площі масиву в 4 рази та величини ядра у 2 рази. Таким чином досягнуто восьмикратного зменшення кількості операцій на етапі розмиття.

П'ятий крок. Визначається положення найближчої межі відносно кожної точки зображення. На цьому кроці відкидаються точки, що розміщені на відстані 3 і більше пікселів від найближчої межі. Оскільки після розмиття зображення велика кількість малопомітних меж втратилася, то цей крок дозволяє далі не враховувати більшість точок зображення. Характерною ознакою межі є те, що на ній досягає локального максимуму сума проєкцій градієнтів точок околу на напрямок градієнта в точці цієї межі. Інакше кажучи, при русі в напрямках градієнта або антиградієнта на межі досягається локальний максимум проєкції градієнта на цей напрямок. Для досягнення субпіксельної точності для кожної точки зображення P будемо шукати її відстань до межі в напрямку градієнта. Для цього виділимо 6 прямокутників A_i розміром 1×2 піксели, які розташовані послідовно і формують прямокутник розміром 6×2 піксели, у центрі якого міститься точка P . Повернемо цей прямокутник таким чином, щоб його більша сторона збігалася з напрямком градієнта в точці P . Для кожного з прямокутників A_i знаходимо значення V_i як суму проєкцій градієнтів пікселів, що перетинаються з ним (піксел розглядається як квадрат), помножену на площу їх перетину. Якщо максимальне значення масиву V розміщене або на початку, або в його кінці, тоді точка P вважається далекою від межі і більше не розглядається. У протилежному випадку на основі елементів масиву V шукається положення межі на прямій градієнта точки P як розв'язок задачі мінімізації, що апроксимує значення V_i параболою

$Ax^2 + Bx + C$:

$$\begin{aligned} & \left(\frac{25}{4}A - \frac{5}{2}B + C - V_1 \right)^2 + \left(\frac{9}{4}A - \frac{3}{2}B + C - V_2 \right)^2 + \\ & + \left(\frac{1}{4}A - \frac{1}{2}B + C - V_3 \right)^2 + \left(\frac{1}{4}A + \frac{1}{2}B + C - V_4 \right)^2 + \\ & + \left(\frac{9}{4}A + \frac{3}{2}B + C - V_5 \right)^2 + \left(\frac{25}{4}A + \frac{5}{2}B + C - V_6 \right)^2 \rightarrow \min. \end{aligned}$$

Розв'язком екстремальної задачі є такі величини:

$$A = \frac{5}{56}(V_1 + V_6) - \frac{1}{56}(V_2 + V_5) - \frac{1}{14}(V_3 + V_4);$$

$$B = \frac{1}{7}(V_6 - V_1) + \frac{3}{35}(V_5 - V_2) + \frac{1}{35}(V_4 - V_3).$$

Максимуму парабола досягає в точці $-\frac{B}{2A}$. Якщо це значення перевищує 3, то точка вважається далекою від межі, інакше воно множить на вектор напрямку градієнта в точці P і зберігається як зміщення межі відносно цієї точки в масиві $O(x, y)$. Найбільшу кількість розрахунків на цьому етапі становить пошук площ перетину прямокутників A_i та пікселів зображення. Щоб уникнути обчислювальних затрат, напрямки векторів градієнтів обмежуються 256 можливими варіантами і для всіх цих значень відшукуються площі перетину. Точність на етапі попередніх розрахунків забезпечується використанням чисел з плаваючою комою, тоді як під час основних розрахунків використовуються лише цілочислові операції.

Шостий крок. Із напрямків до меж формуються точки, на основі яких будуватимуться лінії на зображенні. Цей крок складається з двох частин. У першій частині послідовно беруться всі пари сусідніх по горизонталі точок. Якщо ці точки (x, y) і $(x+1, y)$ розміщені близько до межі і виконуються умови

$$O_x(x, y) > 0; O_x(x+1, y) < 0;$$

$$3(O_x(x, y) - O_x(x+1, y)) > |O_y(x, y) - O_y(x+1, y)|,$$

то в перелік точок додається елемент з координатами $\left(x + \frac{O_x(x, y)}{O_x(x, y) - O_x(x+1, y)}, y\right)$. Аналогічним чином пари сусідніх точок по вертикалі опрацьовуються у другій частині цього кроку.

Сьомий крок. На ньому будуються лінії меж у зображенні. Їх конструювання починається з оброблення знайдених на попередньому етапі точок. Для кожної точки P виконуються такі операції. Спочатку з'ясовується чи не оброблялася якась точка на відстані $\sqrt{2,5}$ пікселів від поточної. Якщо оброблялася, тоді в більшості випадків лінія, на якій міститься точка P , вже додавалася і тому відбувається перехід до наступної точки. Якщо лінія ще не додавалася, вважається, що вона буде додана при виборі іншої точки з цієї лінії. Якщо ні точка P , ні сусідні точки не оброблялися, тоді починається процедура додавання лінії.

Процедура уточнення положення точки K починається з визначення значення градієнта шляхом кубічної інтерполяції функції $G(x, y)$. Далі визначаються зміщення O_1 за схемою, описаною на п'ятому кроці алгоритму, а також точки $K_1 = K + O_1$ і $K_2 = K_1 + O_1$. Так само для точки K_2 знаходиться значення зміщення O_2 . Якщо $O_1 O_2 < 0$, тоді межі містяться між точками K і K_2 і положення точки K становить:

$$K + \frac{2O_1}{1 - \frac{O_1 O_2}{O_1^2}}$$

Інакше положення переміщається до межі:

$$K_2 + O_2.$$

Якщо у процесі ітераційної процедури або $|O_1|$, або $|O_2|$ виявилися більшими ніж 3, тоді уточнення положення точки вважається неуспішним. Ця процедура повторюється декілька разів. У межах запропонованого алгоритму емпірично встановлено, що 5 ітерацій дають достатню точність.

Повернемося до процедури додавання лінії. Вона починається із застосування уточнення положення точки P . Якщо в процесі уточнення відбулося віддалення на 2 піксели від початкового місця, тоді є небезпека наближення до лінії, яка вже додавалася, і тому оброблення точки P зупиняється. Далі алгоритм починає додавати точки у двох напрямках. Початкове положення другої точки лінії P' відносно першої P напрямлене перпендикулярно до градієнта в точці P . Після цього уточнюється положення P' за аналогічною до описаної вище схемою, за винятком того, що після кожної ітерації вектор $|P - P'|$ нормується шляхом зміщення точки P' . Інші точки додаються за аналогічною схемою з урахуванням того, що їх початкове положення розраховується як

$$P_{i+1} = 2P_i - P_{i-1}.$$

Процес додавання точок до лінії зупиняється у випадку, якщо виконується одна з таких умов:

- в околі точки радіусом $\sqrt{2,5}$ піксела є точка, яка уже додавалася (і вона відрізняється від попередньої);
- кут між лінією та градієнтом в точці менший від порогового значення $\frac{\pi}{12}$;
- поворот лінії перевищує порогове значення $\frac{\pi}{2}$.

Другий напрямок, у якому алгоритм додає точки до лінії, як і перший перпендикулярний до градієнта точки P , але зорієнтований у протилежний бік. Результатом роботи цього етапу є набір ліній, стабільних до типових перетворень зображення.

Восьмий крок. На ньому виділяються окремі точки на лініях. В основу його роботи покладено характеристику $D(P)$ кожної точки ліній, яка являє собою значення відстані від самої точки до прямої, що з'єднає точки лінії, які віддалені від P на 3 піксели по різні боки. Якщо для точки P значення функції D більше ніж для сусідніх точок по лінії, тоді вона розглядається як потенційна бути ключовою точкою. Для більшої точності її положення наближується параболою відповідно до значень функції D для самої точки та найближчих сусідніх точок.

Після отримання екстремальних точок описаним способом відбувається їх остаточна фільтрація за допомогою підібраних критеріїв з метою досягнення найбільш стабільного результату. Перелік знайдених критеріїв для точки P :

- різниця між значенням $D(P)$ та максимальним значенням $\max_i D(P_i)$, де P_i — точки, віддалені на 1, 1,5 та 2 піксели по обидва боки від точки P на лінії, не менша ніж 0,1;
- відстань від краю лінії не менша ніж 3;
- відношення довжини градієнта в точці P до середньої довжини градієнта в її околі більше ніж 1,4.

ПІДБІР ПАРАМЕТРІВ АЛГОРИТМУ

Останній етап описаного алгоритму передбачає визначення критеріїв фільтрації, які дозволять досягнути максимальної стабільності отриманого результату. Для розв'язання поставленого завдання було розроблено окремий програмний модуль, який дозволяє перевіряти вплив заданих обмежень та графічно виводити параметри стійкості для всіх порогових значень в обраному інтервалі (рис. 5).

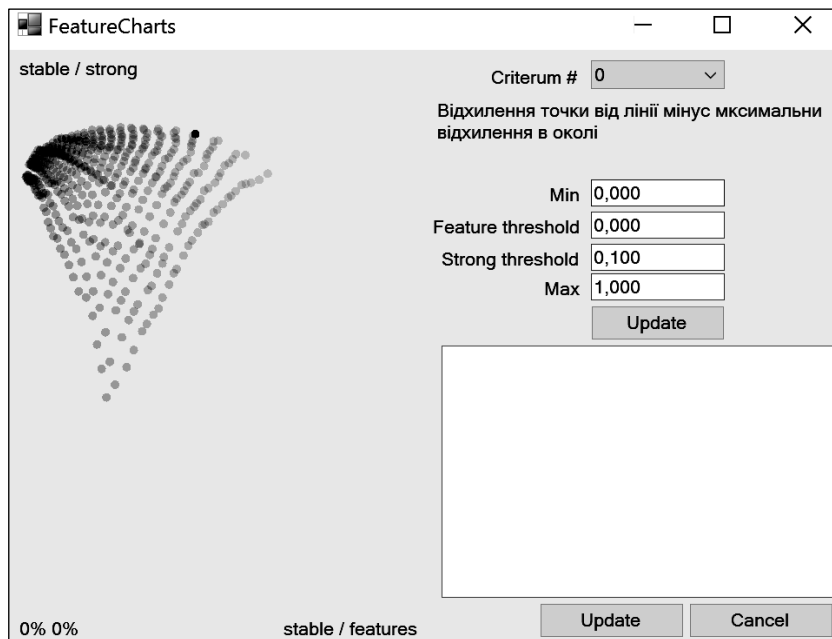


Рис. 5. Вікно підбору меж для критеріїв фільтрації та класифікації ключових точок

У випадковому списку «Criterion #» вибирається номер критерію, який підлягає оптимізації. До розгляду введено такі критерії:

- $D(P) - \max \left(\min_{Q \in [P-2, P]} D(Q), \min_{Q \in [P, P+2]} D(Q) \right)$, де P — точка, яка розглядається; $D(Q)$ — відстань від Q до відрізка, що з'єднує точки на межі, віддалені на 3 піксели від Q (у різні боки);

- $D(P)/\max\left(\min_{Q \in [P-2, P]} D(Q), \min_{Q \in [P, P+2]} D(Q)\right)$;
- відстань від точки P до краю лінії, на якій вона розміщена;
- косинус кута між нормаллю $N(P)$ до лінії в точці P та градієнтом $G(P)$;
- $|G(P)|/|G(Q)| \mid Q \in O(P)$;
- $(G(P), V(P))/|G(Q)| \mid Q \in O(P)$, де $V(P)$ — нормований напрямок уздовж лінії в точці P .

Параметрами для критеріїв є два порогові значення, перехід через одне з них означає належність точки до множини ключових точок, а перехід через друге — до більш стійкої підмножини цих точок. На графіку, розміщеному в лівій частині вікна, зображено статистичні дані, отримані для різних порогових значень поточного критерію. Значення інших критеріїв — фіксовані. На основі графіка окремо підбираються найкращі значення порогів для кожного з критеріїв. У результаті пошуку параметрів за допомогою розробленого модуля виділено лише 3 найголовніші критерії і знайдено для них оптимальні порогові значення.

ПОРІВНЯННЯ РЕЗУЛЬТАТІВ РОБОТИ З ІНШИМИ ТЕХНОЛОГІЯМИ

Підхід, який полягає у використанні двох порогових значень, дозволяє збільшити відношення кількості сформованих пар до кількості перевірок, які виконуються для їх побудови. Для цього база пошуку ключових точок формується з тих, які пройшли слабший критерій, а пошук в базі здійснюється для ключових точок іншого зображення, які пройшли більш сильний критерій (стійкі ключові точки). Такі дії дозволяють урахувати те, що при переході до іншого зображення точка перестає відповідати критерію ключової. У запропонованій технології більшість описаних випадків зводяться до переходу точки з множини стійких до звичайних, які також наявні в базі, на основі якої формуються пари ключових точок.

Використання цієї ідеї та описаного алгоритму в цілому дає змогу значно випередити інші алгоритми за показником стабільності результату, не втративши при цьому в швидкодії.

Тестування якості результату проводилося на наборі синтезованих даних. Знайдені стійкі ключові точки одного зображення переводилися у тривимірний простір і проектувалися на площину іншого зображення. Якщо біля отриманої проекції певної точки містилася ключова точка другого зображення, то вважалося, що вона коректно знайдена. Як стійкість результату роботи алгоритму взято відношення коректно знайдених ключових точок до загальної кількості стійких ключових точок зображення, які у процесі перепроєктування потрапляли у ділянку видимості другого зображення. Значення стійкості усереднювалося для серії пар зображень.

За описаним критерієм запропонована технологія показала результат 91%, тоді як для інших відомих алгоритмів це значення становить: DoG (частина SIFT [9]) — 66%, MSER [10] — 75%.

Синтезовані дані формувалися таким чином, що об'єкти, на яких містилися стабільні ключові точки в одному зображенні, могли повністю зникнути з іншого зображення, оскільки їх перекриває деякий ближчий об'єкт. Це зменшило знайдені значення стійкості на декілька відсотків.

Алгоритм на швидкодію не порівнювався з аналогами через відсутність у відкритому доступі найбільш продуктивних реалізацій інших алгоритмів. Однак запропонований метод побудовано на основі алгоритму, який використовувався для отримання панорамних знімків у режимі реального часу [11, 12]. Його продуктивності вистачало на переведення перерахунку до 20 разів за секунду на смартфоні чотирирічної давності (Samsung Galaxy S3).

ВИСНОВКИ

У межах роботи запропоновано технологію виділення стабільних ключових точок на зображенні, яка за якістю результату значно переважає аналоги. Для оцінювання характеристик розробленої технології та порівняння її з іншими подібними технологіями створено програмне забезпечення для синтезу зображень з інформацією про відстань до об'єктів.

Описаний метод ґрунтується на процедурі виділення стійких меж. Він складається з восьми кроків, на яких послідовно звужується множина точок, що стають ключовими. Запропоновано підхід, який полягає в класифікації ключових точок за ступенем стійкості і допомагає зменшити обсяг процедури пошуку під час компонування в пари. Для вибору критеріїв класифікації та визначення порогових значень розроблено програмний модуль, який у графічному вигляді показує значення стійкості для можливих порогових значень і дає змогу серед усіх варіантів обрати оптимальні.

Результати, отримані в роботі, дозволяють підвищити продуктивність методів реєстрації зображень як за рахунок пришвидшення пошуку ключових точок, так і завдяки тому, що розглянутий алгоритм видає більш стійкі результати, які істотно скорочують розрахунки, необхідні на наступних етапах реєстрації.

ЛІТЕРАТУРА

1. Гаращенко Ф.Г. Интерактивное построение панорамных снимков с помощью портативных компьютеров / Ф.Г. Гаращенко, А.Ю. Кобзарь // Проблемы управления и информатики. — 2013. — № 6. — С. 90–102.
2. *Tango* [Digital resource] // Google Inc. — Access mode: <http://get.google.com/tango>
3. *Nect-generation 3D Virtual Tours* [Digital resource] // GeoCV. — Access mode: <http://www.geocv.com>
4. *Matterport: Immersive 3D Spaces for real-world applications* [Digital resource] // Matterport. — Access mode: <https://matterport.com>
5. *Технология Intel® RealSense™* [Електронний ресурс] // Intel Corporation. — Режим доступу: <http://www.intel.ua/content/www/ua/uk/architecture-and-technology/realsense-overview.html>
6. *Kinect-Windows app development* [Digital resource] // Microsoft Corporation. — Access mode: <https://developer.microsoft.com/en-us/windows/kinect>

7. Публічна кадастрова карта України [Електронний ресурс] // Центр державного земельного кадастру. — Режим доступу: <http://www.map.land.gov.ua/kadastrova-karta>
8. Shuttle Radar Topography Mission [Digital resource] // National Aeronautics and Space Administration. — Access mode: <http://www2.jpl.nasa.gov/srtm/>
9. Lowe D. Distinctive Image Features from Scale-Invariant Keypoints / David Lowe // International Journal of Computer Vision. — 2004. — Vol. 60, Issue 2. — P. 91–110.
10. Donoser M. Efficient Maximally Stable Extremal Region (MSER) Tracking / M. Donoser, H. Bischof // Conference on Computer Vision and Pattern Recognition. — 2006. — Vol. 1. — P. 553–560.
11. Кобзар А.Ю. Обчислювальні технології інтерактивної побудови панорамних зображень в режимі реального часу : дис. канд. техн. наук : 01.05.02 / Кобзар Андрій Юрійович. — Київ, 2015. — 184 с.
12. Гаращенко Ф.Г. Сопоставление снимков панорамы на основании границ объектов / Ф.Г. Гаращенко, А.Ю. Кобзарь // Компьютерная математика. — 2014. — № 1. — С. 46–53.

Надійшла 17.03.2017