

А. М. Кривоножко, О. В. Петров, Г. В. Щербак, В. М. Сургай

Харківський національний університет Повітряних Сил імені Івана Кожедуба, Харків, Україна

## РОЗРОБКА МЕТОДУ ВИЗНАЧЕННЯ ПОЛЯ ВЕКТОРІВ РУХУ ЗІ ЗВАЖУВАННЯМ ВИМІРІВ БЛОКІВ ЗОБРАЖЕННЯ

**Анотація.** На сучасному етапі не в повній мірі досліджено питання впровадження в систему автономного управління БПЛА моделей та методів спрямованих на обробку блоків зображення та оптичного потоку, що отримується безпілотним комплексом в процесі виконання польотного завдання. Впровадження вказаних моделей та методів дозволяє впровадити ієрархічну обробку зображення поверхні, над якою проводиться політ, що значно розширить можливості системи керування БПЛА щодо рефлексивного управління. **Предметом** вивчення в статті є процеси рефлексивного управління безпілотними летальними апаратами. **Метою** статті є розробка методу визначення поля векторів руху зі зважуванням вимірів блоків зображення. **Завдання:** розробити моделі та методи спрямованих на обробку блоків зображення та оптичного потоку, що отримується безпілотним комплексом в процесі виконання польотного завдання. Аналізованими **моделями та методами** є: модель оптичного потоку, адаптивний метод текстурного аналізу для визначення особливостей зображення, метод зважувальної обробки блоків зображення для визначення поля векторів руху. Отримані такі **результати**. Розроблені моделі та методи, які є базисними для розробки методу визначення перешкод та динамічних об'єктів для безпілотних літальних апаратів на основі поля векторів руху. Зокрема, розроблено метод визначення поля векторів руху зі зважуванням вимірів блоків зображення. Вказаний метод є базисним для розробки методу визначення перешкод та динамічних об'єктів для безпілотних літальних апаратів на основі поля векторів руху, застосування якого значно розширить можливості рефлексивного управління безпілотними комплексами при виконанні різноманітних польотних завдань. **Висновки.** Впровадження в систему управління БПЛА моделі та методів спрямованих на обробку блоків зображення та оптичного потоку дозволить в подальшому виявляти перешкоди в реальному часі і забезпечувати безпеку руху шляхом визначення та аналізу поля векторів руху.

**Ключові слова:** безпілотний літальний апарат, блоки зображення, оптичний потік, поле векторів руху.

### Вступ

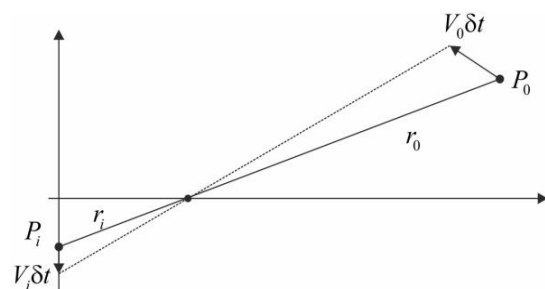
**Постановка проблеми у загальному вигляді.** Науково-технічний прогрес і пов'язане з ним вдосконалення елементної бази істотно змінюють інтенсивність застосування БПЛА, постійно розширюють області використання та задачі, які вирішуються за допомогою безпілотних комплексів. Широке застосування апаратури виготовленої з використанням сучасної елементної бази робить проблематичним застосування існуючого науково-методичного апарату керування БПЛА через неповне використання можливостей щодо автономного управління. Вказані факти обумовлюють необхідність критичного аналізу відомих методів управління БПЛА та розробку нового науково-методичного апарату щодо цього питання [1].

**Аналіз літератури.** Аналіз літератури показав [2, 3], що на сучасному етапі не в повній мірі досліджено питання впровадження в систему автономного управління БПЛА моделей та методів спрямованих на обробку блоків зображення та оптичного потоку, що отримується безпілотним комплексом в процесі виконання польотного завдання. Впровадження вказаних моделей та методів дозволяє впровадити ієрархічну обробку зображення поверхні, над якою проводиться політ, що значно розширить можливості системи керування БПЛА щодо рефлексивного управління. Даний факт тягне за собою необхідність вдосконалення існуючого науково-методичного апарату, що застосовується для здійснення рефлексивного управління БПЛА.

**Метою статті** є розробка методу визначення поля векторів руху зі зважуванням вимірів блоків зображення.

### Основна частина

**Розробка методу визначення поля векторів руху зі зважуванням вимірів блоків зображення.** Оптичний потік являє собою процедуру оцінювання яскравісно-геометричних змін між поточним і попереднім кадрами. Зміни зображень викликаються рухом об'єктів перед нерухомою камерою або рухом камери в навколишньому середовищі. Оптичний потік – це уявний рух видимого поля (двовимірного розподілу яскравості), що спостерігається при русі камери відносно зображуваних об'єктів або об'єктів відносно камери. Визначимо поле руху. З цією метою припишемо кожній точці зображення вектор швидкості. Нехай у певний момент часу точка  $P_i$  зображення відповідає певній точці  $P_0$  на поверхні об'єкта. Ці дві точки зв'язані рівняннями проектування. Точка об'єкта  $P_0$  переміщується відносно камери зі швидкістю  $V_0$ . Відповідно рухається і точка зображення. За час  $\delta t$  точка переміщується на відстань  $V_0 \delta t$  (рис. 1).



**Рис. 1.** Переміщення точки  $P_0$  зовнішнього середовища, що викликає переміщення відповідної точки  $P_i$  зображення

Яскравості розподілу рухаються разом зі спостережуваними об'єктами. В ідеалі оптичний потік відповідає визначеному раніше полю руху. Але практика не завжди це підтверджує.

Кожен кадр відеопослідовності розміром  $M \times N$  пікселів розбивається на множину блоків  $I_{ij}$ . Блоки не перетинаються і мають розміри  $n \times n$  пікселів, де  $i, j$  – координати центра блока.

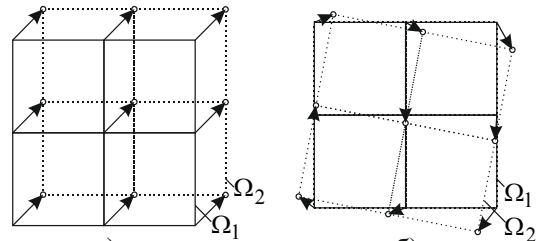
Припустимо, що оптичний потік не сильно відрізняється від поля руху. Тоді можна оцінити відносний рух, виходячи зі зміни зображення в часі. Яскравості зображення рухаються разом зі об'єктами, що спостерігаються. Фактично при оцінюванні параметрів руху послідовностей зображень об'єктом дослідження є система формування зображень. У загальному випадку дана система складається з початкового зображення, середовища, факторів впливу, датчика і пристрою попередньої обробки зображення.

Початкове зображення зазвичай є динамічним. Якщо відеокамера знаходиться на рухомому об'єкті, то одержувані зображення різних кадрів мають взаємні зрушення, повороти, зміни масштабу та ін. Крім того, на послідовності зображень також впливають спотворення оптичних трактів, нестабільність розгортки зчитувальних пристроїв і дефекти виготовлення датчиків, зміна оптичної щільності середовища, турбулентність атмосфери тощо. Тобто, одержувані зображення піддаються впливу майже всіх складових системи формування зображень [4].

Коли зображення рухаються, одні й ті ж елементи сцени на різних кадрах мають різні координати, що геометрично описується деформацією і рухом сітки відліків у просторі при нерухомій сцені або рухом елементів сцени (рис. 2).

У загальному випадку сітка відліків кожного кадру  $\Omega_1$  є не прямокутною, а криволінійною. Зміщення кожного вузла  $ij$  сітки  $\Omega_2$  відносно його положення на  $\Omega_1$  (або зміщення елемента сцени у вузлі  $ij$  відносно його положення на  $\Omega_1$ ) задається векто-

ром  $(u, v)_{ij}$ . Сукупність таких векторів становить векторне випадкове поле  $z = \{(u, v)_{ij} : ij \in \Omega\}$ , задане на сітці  $\Omega$ . При розгляді послідовності кадрів отримується випадкове поле  $\{(u, v)_{ij}^t : ij \in \Omega, t \in T\}$ , де  $t$  – номер кадру. Очевидно, що при такому поданні розмірність векторів  $(u, v)_{ij}$  (або  $(u, v)_{ij}^t$ ) збігається з розмірністю сітки  $\Omega$ .



а – паралельне зміщення б – зміщення і поворот

Рис. 2. Міжкадрова деформація сітки зображення

Міжкадрові перетворення можна описати по іншому. Нехай кожне положення сітки розглядається як система координат. Тоді міжкадрові перетворення подаються як випадкові перетворення

$$(u, v)_{xy} = f(xy) = (f_1(xy), f_2(xy), \dots, f_n(xy))^T \quad (1)$$

системи координат  $\Omega_2$  в систему координат  $\Omega_1$ , де  $n$  – розмірність сітки  $\Omega$ . При відомих міжкадрових зміщеннях (зсув, поворот, зміна масштабу тощо), для полегшення опису перетворення (1) задається параметрично

$$(u, v)_{xy} = f(x, y, \alpha). \quad (2)$$

На рис. 3 зображено алгоритм блокового оцінювання оптичного потоку та формування поля векторів руху.

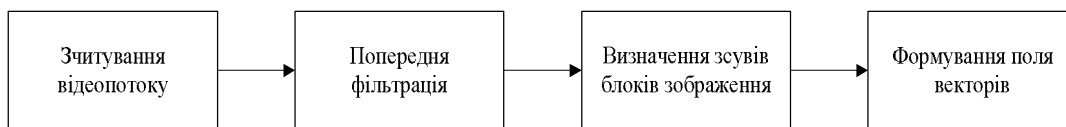


Рис. 3. Алгоритм блокового оцінювання руху

Дослідимо отриману за допомогою відеокамери послідовність кадрів зображень, попередньо розбитих на окремі блоки і заданих на регулярній прямокутній сітці з одиничним кроком.

**Модель оптичного потоку та поблокове визначення векторів руху.** Рівняння оптичного потоку має дві невідомі компоненти вектора руху  $(u, v)$ . Припустимо близькість векторів руху для всіх пікселів з околиці точки  $(x, y)$ . Тоді можна отримати більшу кількість рівнянь для двох невідомих. Запропонований метод базується на полі руху, побудованому на основі оцінок зміщення не точок, а окремих блоків  $B_{ij}$ , на які попередньо розбитий кожен кадр відеопослідовності розміром  $M \times N$  пікселів

$$B_{ij}(t) = \{I(x_i, y_j, t) : x_k, y_l \in B_{ij}\}, \quad (3)$$

$$i - n/2 < k < i + n/2; j - n/2 < l < j + n/2,$$

де  $i, j$  – координати центра блока.

Блоки розміщені послідовно зліва направо, зверху вниз. Сумарна площа всіх блоків, що покривають весь кадр, дорівнює площі кадру. Задача визначення руху зводиться до пошуку векторів руху  $(u, v)_{ij}$  для кожного блока  $I_{ij}$ . Позначимо вектор руху  $\vec{z}_{ij} = (u, v)_{ij}$ . Тоді в момент часу  $t + \Delta t$  яскравісна функція для блока  $B_{ij}$  дорівнює:

$$B_{ij}(t + \Delta t) = \{I(x_i + u_{ij}, y_j + v_{ij}, t + \Delta t)\}. \quad (4)$$

Оптимальна оцінка параметрів вектору руху центра блока складає:

$$\hat{z}_{ij} = \min_{z_{ij}} \sum_{y, x \in B_{ij}}^n |B_{ij}(t) - B_{ij}(t + \Delta t)|^2. \quad (5)$$

Оцінка (5) зводиться чисельно до ітераційної процедури мінімізації

$$Q(z)_{ij} = \sum_{y, x \in B_{ij}}^n |B_{ij}(t) - B_{ij}(t + \Delta t)|^2. \quad (6)$$

Розкладемо  $B_{ij}(t + \Delta t)$  у ряд Тейлора:

$$B_{ij}(t + \Delta t) = B_{ij}^0(t + \Delta t) + \frac{dB_{ij}^0(t + \Delta t)}{dx} u_{ij} + \frac{dB_{ij}^0(t + \Delta t)}{dy} v_{ij}$$

Тоді модельна функція стає лінійною [5].

Після диференціювання функції  $Q(z)_{ij}$  за  $u, v$  і порівнювання частинних похідних до нуля, отримаємо систему рівнянь для визначення оцінок параметрів  $(u, v)_{ij}$ , які забезпечують мінімум функції  $Q$ :

$$\frac{\partial Q(z)_{ij}}{\partial u} = 0, \quad \frac{\partial Q(z)_{ij}}{\partial v} = 0. \quad (7)$$

Диференціюючи вираз (7), отримаємо

$$\begin{cases} \sum_{x, y \in B_{ij}}^n \left[ (B_{ij}(t) - B_{ij}(t + \Delta t)) \left( \frac{dB_{ij}^0(t + \Delta t)}{dx} \right) \right] = 0; \\ \sum_{x, y \in B_{ij}}^n \left[ (B_{ij}(t) - B_{ij}(t + \Delta t)) \left( \frac{dB_{ij}^0(t + \Delta t)}{dy} \right) \right] = 0. \end{cases} \quad (2.14)$$

Для визначення оптимального вектору МНК-оцінки параметра  $\hat{z}_{ij}$  використаємо векторно-матричний вираз

$$\hat{z}_{ij} = (A_{ij}^T A_{ij})^{-1} \cdot A_{ij}^T y_{ij}, \quad (9)$$

де  $A_{ij}$  – визначник матриці даних  $\nabla B_{ij}(t + \Delta t)$ ;  $y_{ij}$  – вектор-стовпець правих частин рівнянь  $\Delta B_{ij}$ ;

$$y = \bar{y} + \varepsilon, \quad (10)$$

де  $\varepsilon$  – вектор-стовпець помилки моделі;

$\bar{y}$  – вектор-стовпець істинних значень  $y$ .

Оцінки  $\hat{z}_{ij}, i = 1, \dots, n; j = 1, \dots, n$  є випадковими величинами, які є сумою істинного значення  $z_{ij}$  і деякої випадкової помилки  $\Delta z_{ij}$

$$\hat{z}_{ij} = z_{ij} + \Delta z_{ij}; \quad \hat{z}_{ij} = z_{ij} + (A_{ij}^T A_{ij})^{-1} A_{ij}^T \cdot \varepsilon, \quad (11)$$

де  $(A_{ij}^T A_{ij})^{-1} A_{ij}^T \cdot \varepsilon$  – вектор помилок оцінювання параметрів  $\hat{z}_{ij}$ .

Мірою точності у статистиці є дисперсія оцінок моделі. Точність та ефективність оцінок залежать

від міри випадкового розкиду значень оцінок  $\hat{z}_{ij}$  відносно істинних значень  $z_{ij}$ . Вона визначається дисперсією і коваріаційною матрицею:

$$\mathbf{cov}(\hat{z}_{ij}) = \begin{pmatrix} \sigma_1^2 & \mathbf{cov}(z_1 z_2) & \dots & \mathbf{cov}(z_1 z_n) \\ \mathbf{cov}(z_2 z_1) & \sigma_2^2 & \dots & \mathbf{cov}(z_2 z_n) \\ \dots & \dots & \dots & \dots \\ \mathbf{cov}(z_n z_1) & \mathbf{cov}(z_n z_2) & \dots & \sigma_n^2 \end{pmatrix}, \quad (12)$$

де  $\sigma_i^2 = D(\hat{z}_{ij})$  – дисперсія оцінки.

Дисперсію оцінки  $\hat{z}_{ij}$  можна знайти як діагональні елементи матриці (12). Вираз матриці оцінок (12), як показано у роботі [6], можна спростити

$$\mathbf{cov}(\hat{z}_{ij}) = \sigma_\varepsilon^2 \cdot (A_{ij}^T A_{ij})^{-1}. \quad (13)$$

Позначимо матрицю  $A_{ij}^T A_{ij}$  як  $G_{ij}$ . Вона тісно зв'язана з коваріаційною матрицею, яка є вихідним об'єктом для різних видів багатовимірного аналізу. Дисперсії і коваріації оцінки  $\hat{z}_{ij}$  можуть бути визначені як елементи матриці  $\sigma_\varepsilon^2 \cdot (A_{ij}^T A_{ij})^{-1}$ .

**Адаптивний метод текстурного аналізу для визначення особливостей зображення.** Відомі декілька алгоритмів визначення особливостей зображення. Так, алгоритм Lucas-Kanade передбачає пошук особливих точок зображення за допомогою матриці Гессе. Детермінант матриці Гессе (гессіан) досягає екстремуму в точках максимальної зміни градієнта яскравості. Він добре детектує точки, кути і краї ліній. Матриця даних  $G$  є добутком матриць Якобі відповідно до виразу  $M \{ \nabla B^T \nabla B \} = -M \{ \nabla^2 B \}$ . В асимптотичному сенсі аналіз матриці Гессе відповідає аналізу добутку матриць Якобі.

Для більш об'єктивної оцінки в роботі використовуватимемо аналіз на основі числа обумовленості матриці даних  $G$ . Дане число є мірою чутливості системи лінійних рівнянь до похибок завдання вектора правих частин рівнянь. Число обумовленості обчислюється за формулою [5]

$$\mathit{cond}(G) = \|G^{-1}\| \cdot \|G\|, \quad (14)$$

де  $\|G\|$  – норма матриці  $G$ ;  $\|G^{-1}\|$  – норма оберненої матриці  $G$ . Чим більше число обумовленості  $\mathit{cond}$ , тим сильніше вплив і тим більше нестійкий процес знаходження розв'язок. Число обумовленості матриці залежить від вибору норми. Якщо матриця добре обумовлена, то її число обумовленості буде малим при будь-якому виборі норми, а якщо вона погано обумовлена, то її число обумовленості буде великим при будь-якому виборі норми. Доведено, що стійка модель має число обумовленості  $\mathit{cond} \{1 \dots 100\}$ . При  $\mathit{cond}$  від 1 до 10 модель має хорошу обчислювальну стійкість; від 10 до 100 – задовільну, при  $\mathit{cond} > 100$  – незадовільну чутливість.

Вимірювання називаються нерівноточними, якщо результати вимірювань отримано не в однакових умовах і їм відповідають різні дисперсії, а отже, і середні квадратичні похибки. Для обробки нерівноточних вимірювань використовують нову характеристику точності вимірювання – вагу вимірювання. Для характеристики зображення або його фрагмента введемо коефіцієнт текстурованості (показник оцінки числа обумовленості матриці  $G$ ):

$$k = \frac{1}{NM} \sum_{ij} \frac{1}{\text{cond}(G_{ij})} \times 100, \quad (15)$$

де  $NM$  – розмір області.

Значення коефіцієнта текстурованості  $k$  по всьому зображенню для різних типів тестових зображень (рис. 4), наведено в табл. 1.

Отже процедура складається з декількох взаємопов'язаних кроків, основним з яких є розрахунок величини  $k$  для певної області. За її значенням можна адаптивно визначити поріг для помилок векторів. При неперевищенні порога блок необхідно розбити на чотири блоки меншого розміру, використовуючи принципи адаптації для областей зі складною текстурою або в областях, де поточний розмір блока не може забезпечити прийнятний рівень числа обумовленості. Мінімальний розмір блоку складає 8x8 пікселів.

Таблиця 1 – Коефіцієнти  $k$  при блочному розбитті на блоки 8x8 пікселів

Зображ.	Chessboard	Fingerprint	Cameraman	Karta	Baboon	House
Коеф. $k$	3.4685	3.1355	2.4398	2.4084	2.3558	2.2465

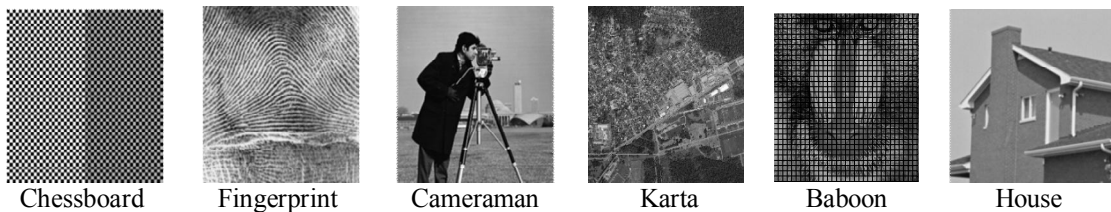


Рис. 4. Типи тестових зображень

Проведемо текстурний аналіз для тестового зображення Karta (рис. 5) і визначимо розподілення блоків з різним показником текстури (рис. 6).

Розподіл блоків з різними показниками текстури показав, що блокові алгоритми оцінки руху є вигідним компромісом за співвідношенням обчислю-

вальна складність / точність знайдених векторів [8]. Побудова універсальних алгоритмів текстурного аналізу зображень із заданими властивостями і простою апаратною реалізацією можлива шляхом комбінування алгоритмів різних категорій в рамках класу блокових методів.

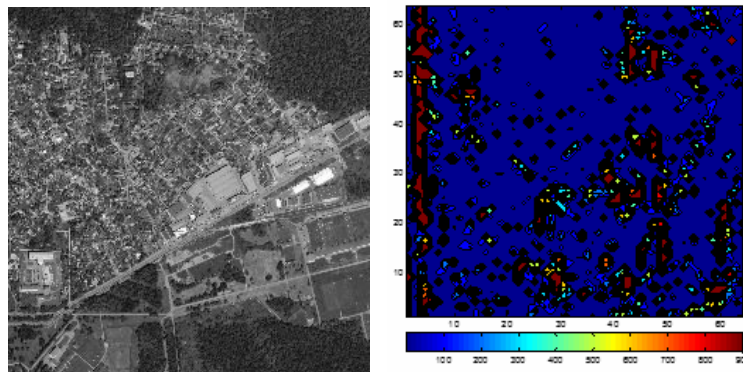


Рис. 5. Тестове зображення Karta

а – блоковий текстурний аналіз б – діаграма розподілу показника текстурованості

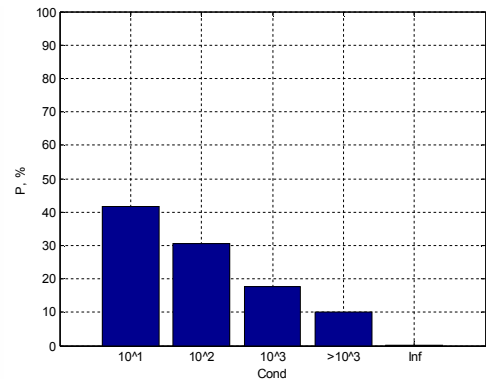


Рис. 6. Текстурний аналіз тестового зображення Karta

**Метод зважувальної обробки блоків зображення для визначення поля векторів руху.** Для визначення руху об'єкта необхідно розв'язати задачу пошуку вектора руху кожного блока  $I_{ij}$ . При цьому вектори  $(u, v)_{ij}$  визначаються співвідношенням

$$(u, v)_{ij} = \underset{(u, v)_{ij} \in B_{ij}}{\text{arg min}} (F(t, i, j, (u, v)_{ij})),$$

де  $F(t, i, j, (u, v)_{ij})$  – функція відповідності блоків, тобто міра близькості блоків поточного і попере-

днього кадрів. Такою функцією, наприклад, є функція SAD (Sum of Absolute Differences)

$$SAD(t, i, j, u, v) = w_{ij} \sum_{p \in I_{i, j}} |I(p, t-1) - I(p + (u, v), t)|, \quad (16)$$

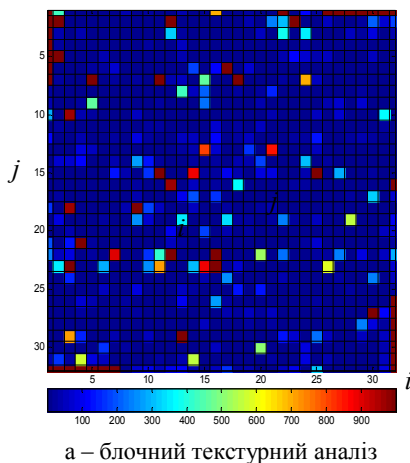
$$w_{ij} = \begin{cases} 1, & \text{if } \text{cond}(G_{ij}) < 100; \\ 0, & \text{if } \text{cond}(G_{ij}) \geq 100. \end{cases}$$

Розглянемо відеопослідовність з двох кадрів (рис. 7). Результатом аналізу цієї послідовності є розподіл показника текстури (числа обумовленості

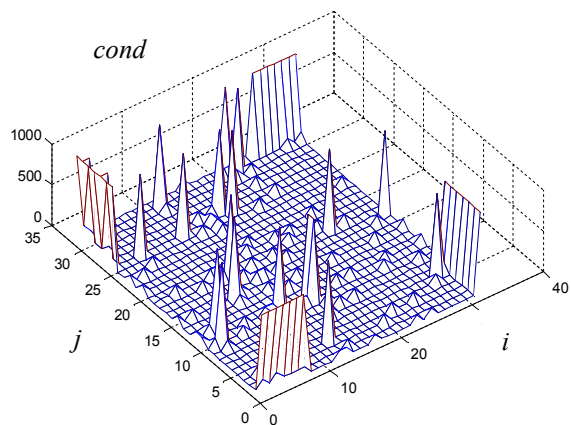


матриці  $G$ ) кожного блоку зображення. Аналізу поточного кадру (рис. 8) дозволяє відокремити неінформативні блоки і визначити відносне розташування інформативних блоків один до одного. Кількісний показник, що перевищує певний поріг, визначає групи блоків, що відповідають області дослідження. Інші блоки вважаються помилково виявленими та ігноруються.

Раніше доведено, що вектор руху вважається визначеним в областях з числом обумовленості від 1 до 10. Якщо  $cond > 10$ , блоки вважаються помилково виявленими та ігноруються (вектор руху дорівнює 0). У гладких областях визначник матриці даних нульовий,  $cond \rightarrow \infty$ , і вектор руху для цих областей не обчислюється.



а – блочний текстурний аналіз



б – 3D діаграма розподілу показника текстурованості

Рис. 8. Аналіз поточного кадру

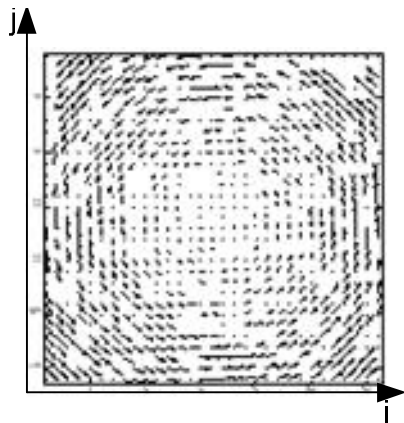
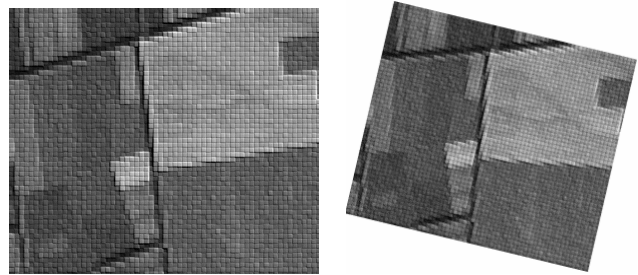


Рис. 9. Поле векторів поточного кадру

Алгоритм дій у методі текстурного аналізу зображення наведений на рис. 10.

Для кожного блока поточного кадру мінімізується функція відповідності блоків за четвертим аргументом. Область мінімізації має бути підмножиною області пошуку. Вектором руху для кожного блока є обчислений аргумент мінімуму функції відповідності. Іншими словами, при обчисленні функції відповідності визначається схожість двох блоків – поточного і попереднього кадрів, зміщеного на вектор  $(u, v)_{ij} \cdot \Delta t$ . Отже, у процесі мінімізації функції відповідності відшукується блок попереднього кадру, найбільш схожий на поточний блок.



а – попередній кадр

б – поточний кадр

Рис. 7. Послідовність зображень

Для визначення руху у послідовності зображень, наведених на рис. 7, побудовано поле векторів (рис. 9).

На першому кроці визначається розмір поточного зображення, яке далі розподіляється на чотири рівні блоки.

На третьому кроці обчислюється ймовірність помилки області. Якщо помилка знаходиться у межах заданого порогу (4-й крок), то визначаються вагові коефіцієнти для всіх блоків області (5-й крок). Якщо ймовірність помилки області знаходиться за межами порогу, то область розбивається на чотири менших підблоки (6-й крок). Якщо підблок знаходиться у межах мінімального розміру (7-й крок), то здійснюється перехід на 3-й крок. Для підблоків, розмір яких знаходиться за межами мінімального розміру, визначаються вагові коефіцієнти (8-й крок).

Таким чином, для розрахунку блокового визначення руху можуть бути використані повний перебір, шаблонні методи, метод ієрархічного пошуку [7] і методи, що ґрунтуються на векторах-кандидатах [6].

У роботі використовується комбінований метод, що об'єднує методи повного перебору і ієрархічного пошуку.

**Метод ієрархічної обробки зображення для оцінки поля векторів руху в реальному часі.** Пари кадрів однакового розміру називаються рівнями. Відповідно опорний і поточний кадри одного рівня мають однаковий розмір. Тоді всі пари кадрів можна подати  $N$  рівнями.

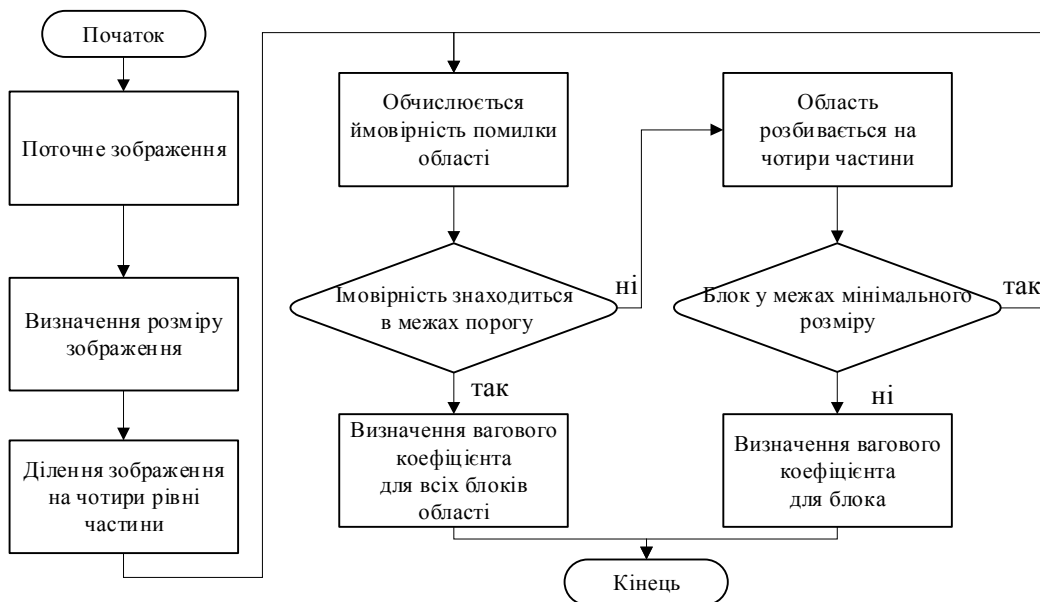


Рис. 10. Алгоритм дій у методі текстурного аналізу зображення

Пронумеруємо рівні відповідно до розміру кадрів, що містяться в них, від меншого до більшого: на 1-му рівні містяться кадри мінімального розміру,

на  $N$ -му – кадри вихідного розміру.

Відповідно до такої схеми, процес оцінювання руху складається з  $N$  кроків. На кожній ітерації обробляється пара кадрів певного рівня – від кадрів меншого розміру до більшого. При цьому оцінюється рух, а стартовою точкою на кожній ітерації є векторне поле, отримане з попередньої ітерації. Іншими словами, на кожній наступній ітерації уточнюються вектори, розраховані на попередній ітерації. Для забезпечення постійної кількості блоків в кадрі на

кожній ітерації при переході на наступну ітерацію розміри області пошуку і блоків, для яких оцінюються вектори, зазвичай збільшуються в  $2n$  рази. Даний метод є стійким до шуму.

Це пояснюється тим, що при зменшенні зображень, як правило, видаляються високочастотні шуми [6, 7].

Але з видаленням високочастотних шумів часто видаляються дрібні деталі. В окремих випадках це є причиною неправильного визначення руху в деталізованих областях.

Схема методу ієрархічної обробки зображення для оцінки поля векторів руху в реальному часі наведена на рис. 11.

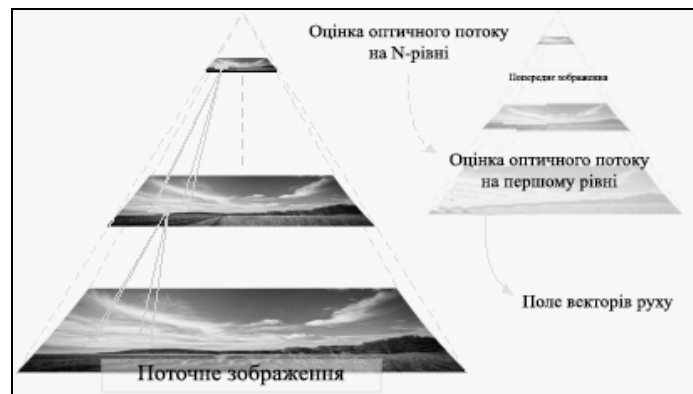


Рис. 11. Схема методу ієрархічної обробки зображення для оцінки поля векторів руху в реальному часі

Суть ієрархічної обробки зображень полягає в такому. Нехай є  $N$  рівнів кадрів. Спочатку розраховуються  $N - 1$  зменшених «копій» поточного і опорного кадрів.

При цьому кожна копія меншого рівня в  $2n$  разів менша попередньої. Тоді всі пари опорного і поточного кадрів представлені  $N$  рівнями.

Етапи методу визначення поля векторів руху з зважуванням вимірів блоків зображення наведені на рис. 12.

## Висновки

1. Розроблено метод визначення поля векторів руху зі зважуванням вимірів блоків зображення.

2. Вказаний метод є базисним для розробки методу визначення перешкод та динамічних об'єктів для безпілотних літальних апаратів на основі поля векторів руху, застосування якого значно розширить можливості рефлексивного управління безпілотними комплексами при виконанні різноманітних польотних завдань.

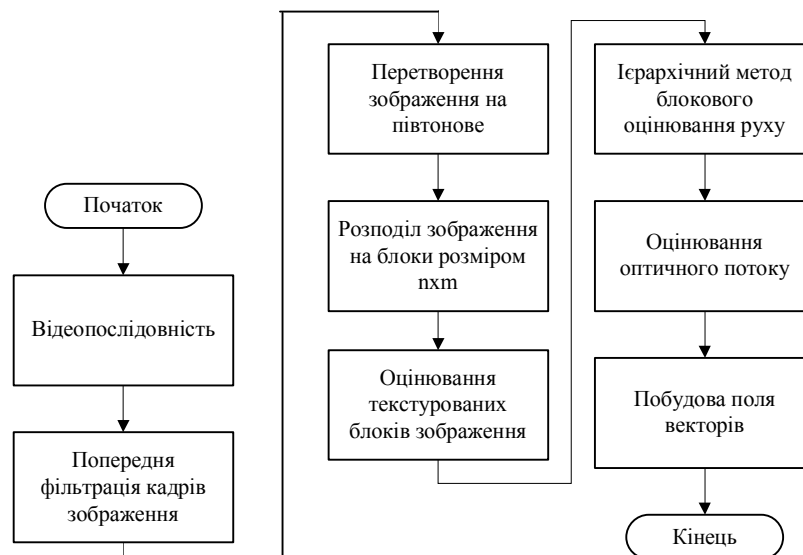


Рис. 12. Етапи методу визначення поля векторів руху з зважуванням вимірів блоків зображення

3. Впровадження в систему управління БПЛА аналізу оптичного потоку зі зважуванням блоків зображення дозволить виявляти перешкоди в реальному часі і забезпечувати безпеку руху шляхом визначення та аналізу поля векторів руху.

#### СПИСОК ЛІТЕРАТУРИ

1. Р. Мохаммади Фархади, "Оценивание аэродинамических коэффициентов и синтез робастного управления боковым движением БПЛА при малых скоростях", Вопросы проектирования и производства конструкций летательных аппаратов, № 3 (91), с. 71-82, 2017.
2. К. В. Мельник, и А. А. Туник, "Проектирование многомерных робастных систем управления полетом на основе процедур  $H_\infty$ -оптимизации и  $\mu$ -синтеза", 2009.
3. . Кортунов, А. Мазуренко, и А. Х. В. Мухаммед, "Средства управления мини и микро-БПЛА", Радиоелектронні і комп'ютерні системи, с. 45–55, 2016.
4. R. D. Colgren, Applications of robust control to nonlinear systems vol. 205: AIAA, 2004.
5. J. O. Rawlings, S. G. Pantula, and D. A. Dickey, Applied regression analysis: a research tool: Springer Sc. & B. Media, 2001.
6. P. G. Hamel, and R. V. Jategaonkar, "Evolution of flight vehicle system identification", J. of aircraft, vol. 33, pp. 9-28, 1996.
7. Свиридов А. С., Коваленко А. А., Кучук Г. А. Метод перерозподілу пропускнуої здатності критичної ділянки мережі на основі удосконалення ON/OFF-моделі трафіку. Сучасні інформаційні системи. 2018. Т. 2, № 2. С. 139–144. DOI: <https://doi.org/10.20998/2522-9052.2018.2.24>
8. J. C. Doyle, B. A. Francis, and A. R. Tannenbaum, Feedback control theory: Courier Corporation, 2013.

Received (Надійшла) 11.05.2020

Accepted for publication (Прийнята до друку) 22.07.2020

#### Development of methods for determining the field of motion vectors with weighting of image blocks measurements

A. Krivonozhko, O. Petrov, G. Shcherbak, V. Surhai

**Abstract.** At the present stage, the issue of introducing into the UAV autonomous control system models and methods aimed at processing image blocks and optical flow obtained by the unmanned complex in the process of performing the flight task is not fully investigated. The introduction of these models and methods allows the introduction of hierarchical image processing of the surface over which the flight is performed, which will significantly expand the capabilities of the UAV control system for reflexive control. The **subject matter** of the article is the reflexive control processes of unmanned aerial vehicles. The **goal** is the development of a method for determining the field of motion vectors with weighing measurements of image blocks. The **tasks**: to develop models and methods aimed at processing image blocks and optical flow received by the unmanned complex in the process of performing a flight task. Analyzed **methods** are: optical flow model, adaptive texture analysis method for determining image features, weighting processing method for image blocks to determine the field of motion vectors. The following **results** were obtained: Models and methods have been developed that are the basis for the development of a method for determining obstacles and dynamic objects for unmanned aerial vehicles based on the field of motion vectors. In particular, a method for determining the field of motion vectors with weighing of image block measurements has been developed. This method is the basis for the development of a method for determining interference and dynamic objects for unmanned aerial vehicles based on the field of motion vectors, the application of which will significantly expand the possibilities of reflexive control of unmanned systems when performing various flight tasks. **Conclusions.** The introduction of a model and methods aimed at processing image blocks and optical flow into the UAV control system will allow in the future to detect obstacles in real time and ensure traffic safety by determining and analyzing the field of motion vectors.

**Keywords:** unmanned aerial vehicle, image blocks, optical flow, motion vector field.