

ПРОГРАМНИЙ КОМПОНЕНТ ВИЗНАЧЕННЯ ІНТЕНСИВНОСТІ ТРАНСПОРТНИХ ПОТОКІВ

Параметри інтенсивності транспортного руху є важливою інформацією для прийняття рішень про параметри управління транспортними потоками. Від точності визначення цих параметрів у значній мірі залежить якість систем управління транспортним рухом. В даній статті представлений аналіз існуючих методів визначення інтенсивності дорожнього руху. На основі аналізу інформації з відеокамер спостереження транспортного руху розроблено програмний компонент для визначення інтенсивності руху транспортних засобів.

The traffic intensity parameters are important information for making decisions about the traffic control parameters. The quality of the traffic control systems depends to a large extent on the accuracy of the definition of these parameters. In this article an analysis of existing methods for determining the intensity of traffic is presented. On the basis of the analysis of information from the cameras of traffic monitoring, a software component for determining the traffic volume of vehicles was developed.

Ключові слова: розпізнавання образів, дорожній рух, інформаційна система управління, OpenCV, C++

Вступ

Постійне зростання кількості транспортних засобів призводить до росту інтенсивності дорожнього руху, разом з цим проблеми управління транспортним рухом набувають великого значення. Неправильно визначені параметри управління транспортним рухом призводять до заторів. Тому задача оптимізації параметрів управління транспортним потоком є актуальною. Через обмеження простору для будівництва нових дорожніх розв'язок та проблеми, пов'язані з перебудовою старої дорожньої інфраструктури, в тому числі через недостатню кількість ресурсів, єдиним шляхом для вирішення проблеми поліпшення якості транспортного руху є розробка систем управління, які враховують поточну інтенсивність руху та інші несприятливі умови (аварії, ремонтні роботи на проїжджій частині дороги). Системи управління транспортним рухом мають не тільки забезпечувати безпеку руху, але й працювати на упередження несприятливих для руху ситуацій.

Сучасний рівень розвитку методів моделювання, інформаційних технологій та технічних засобів для моніторингу рухомих об'єктів створює умови для розробки технологій управління транспортним рухом, що враховує прогнозоване зростання інтенсивності руху для визначення оптимальних параметрів управління та

впроваджує їх в реальному часі. У публікації [1] розглянуто побудову імітаційних моделей великих і складних транспортних систем на основі Петрі-об'єктної технології. У публікації [2] запропонована технологія визначення оптимальних параметрів управління транспортного руху на основі методів імітаційного моделювання та еволюційних методів оптимізації. Проте для її ефективного впровадження необхідна система моніторингу та прогнозування інтенсивності транспортного руху.

У даному дослідженні запропоновано використовувати інформацію, яка надходить з відеокамер спостереження дорожнього руху, для визначення інтенсивності руху та розроблена технологія для обробки зображень цим методом.

Методи визначення інтенсивності дорожнього руху

Необхідною умовою отримання достовірних результатів імітаційного моделювання транспортного руху є достовірні вхідні дані про структуру інтенсивності дорожнього руху. Інтенсивність дорожнього руху визначається кількістю транспортних засобів, що проїхали ділянку дороги по одній з її смуг за одиницю часу.

У сучасних системах управління дорожнім рухом дані про інтенсивність руху транспортних потоків отримують з баз даних,

сформованих за результатами вимірювань на автоматизованих облікових пунктах.

При відсутності автоматизованих облікових пунктів виконується вибірковий візуальний облік дорожнього руху з використанням або без використання спеціальних технічних засобів. При виконанні візуального обліку дорожнього руху збір інформації проводиться не рідше 4 разів на квартал по 4 години на добу: по 1 разу на місяць в робочі дні та 1 раз у вихідний день другого місяця кожного кварталу. У робочі дні облік руху проводиться у вівторок, середу або четвер, а у вихідні - в суботу або неділю. Підсумкові параметри інтенсивності руху та складу транспортних потоків за обліковими пунктами на кожній автомобільній дорозі включаються в галузевий автоматизований банк дорожніх даних. Візуальний облік дорожнього руху без використання спеціальних технічних засобів має ряд недоліків:

- низька економічна ефективність (високі трудові затрати та мала точність вимірювань);
- довготривала присутність спостерігачів на місці збору інформації.

Пункти обліку інтенсивності та складу руху обладнані технічними засобами, принцип дії яких заснований на методах детекції типу автотранспортного засобу. Сучасні технічні засоби засновані на електромагнітному та радіолокаційному принципах детекції автотранспортних засобів. Мотоцикли, трактори, автотранспортні засоби з числом осей 5 і більше, спецтранспорт і т.п. не можуть бути розпізнані такими технічними засобами. Отже, частка нерозпізнаних транспортних засобів в загальному потоці автомобілів може бути значною. Наприклад, на автомобільній дорозі М-1 «Білорусь» зафіксована кількість нерозпізнаних таким способом транспортних засобів, що перевищує 30% [3]. Тобто інформація про третину транспортного потоку фактично відсутня, що є значним недоліком.

Найбільш досконалими системами для визначення параметрів транспортного потоку є системи Weigh-in-Motion (WIM). Ці пристрої призначені для реєстрації осьових і повних навантажень транспортних засобів. На відміну від статичних вагових станцій, WIM-системи не потребують зупинки вантажних автомобілів, що робить їх роботу більш ефективною. Інформація, що отримується з постів, обладнаних системами WIM, є достатньо

повною. Поряд з інформацією про кількість транспортних засобів вони надають дані про швидкість руху автомобілів та їх осьові навантаження. До недоліків WIM-систем можна віднести високу вартість та чутливість до електромагнітних перешкод, наприклад, викликаних блискавками.

Розпізнавання образів за допомогою OpenCV

OpenCV є широко відомою бібліотекою програмних засобів для обробки та аналізу вмісту зображень, у тому числі розпізнавання об'єктів на фотографіях (наприклад, осіб і фігур людей, тексту тощо), відстежування руху об'єктів, перетворення зображень з використанням методів машинного навчання, виявлення загальних елементів на різних зображеннях. Перевагою використання OpenCV є широкий вибір цільової платформи, мови реалізації і відсутність обмежень на поширення готових продуктів з її використанням. Бібліотека OpenCV розділена на невеликі модулі по їх функціональному використанню:

- `opencv_core` містить базові структури та обчислення (математичні функції, генерація псевдовипадкових чисел, DFT, DCT, введення / виведення в XML і т.п.);
- `opencv_imgproc` містить методи обробки зображень (фільтри, перетворення і т. д.);
- `opencv_highgui` реалізує простий GUI, завантаження / збереження зображень та відео;
- `opencv_ml` містить методи та моделі машинного навчання (SVM, дерева прийняття рішень і т. д.);
- `opencv_features2d` реалізує дескриптори (SURF);
- `opencv_video` призначений для аналізу руху і відстеження об'єктів (оптичний потік, шаблони руху, усунення фону);
- `opencv_objdetect` призначений для детекції об'єктів на зображенні (вейвлети Хаара, HOG і т. д.);
- `opencv_calib3d` містить методи для калібрування камери, пошук стереовідповідності і елементи обробки тривимірних даних;
- `opencv_flann` є бібліотекою швидкого пошуку найближчих сусідів (FLANN);
- `opencv_contrib` є супутнім кодом, ще не готовим для застосування;

- `opencv_legacy` є застарілим кодом, який збережено заради зворотної сумісності;
- `opencv_gpu` реалізує прискорення деяких функцій OpenCV за рахунок використання CUDA (NVIDIA).

Окрім основних модулів, є додаткові, які потрібно завантажувати окремо. Одним з таких модулів є «`bioinspired`». Цей модуль забезпечує елементи керування моделлю Gipsa/Listicm сітківки ока на основі невід'ємного просторово-часовий фільтру, що моделює два основні інформаційні канали сітківки:

- `Parvo` - фовеальне бачення для детального кольорового зору;
- `Magno` - периферичне бачення для виявлення чутливих перехідних сигналів (рух та події).

Модель походить від роботи Дженні Еро [6], його студента [7] та включає полярні трансформації Бартельмі Дюрете [7]. В моделі високочастотний просторовий та часовий шум фільтрується. Обидва вихідні шляхи `Parvo` та `Magno` використовують результат фільтрації. Зменшення шуму забезпечує невід'ємну просторово-часову фільтрацію. На виході `Parvo` посилюються статичні текстури, а шум фільтрується (на відео, часовий шум добре видаляється). Однак, рухливі текстури згладжуються [8]. Тоді, деталі рухомих об'єктів можна посилити, лише якщо сітківка стежить за ними і зберігає їх статичними. На виході `Magno` це дозволяє більш точно виявляти події (рух, зміни) з послабленими шумовими помилками навіть у складних умовах освітлення. В якості компромісу вихід `Magno` є низьким сигналом просторової частоти і дозволяє точно визначати спрацьовування подій.

Модель може бути використана як етап попередньої обробки з метою виконання аналізу текстури з покращеним співвідношенням сигналу та шуму і покращеними деталями. Для полегшення використання в програмах два канали моделі сітківки (`Magno`, `Parvo`) застосовуються на всіх вхідних зображеннях. Це не відповідає дійсній топології сітківки, але це практично з точки зору обробки зображень.

Для розпізнавання образів можуть бути використані наступні методи:

1) Метод фільтрації. Зображення, що отримані на виході оптико-електронних перетворювачів, містять сторонню інформацію (перешкоди). При аналізі об'єктів на складному зображенні, фон також є перешкодою. Фільтрація допомагає виділити області, які

цікаві для розпізнавання, без їх аналізу. Результатом фільтрації є оцінка корисного сигналу зображення. Зображення являє собою двовимірну функцію просторових координат, що змінюється повільніше, ніж двовимірною функція, яка описує перешкоду. Тому при оцінці корисного сигналу в кожній точці кадру розглядають околицю цієї точки (деяку множину сусідніх з нею точок), використовуючи загальні характеристики сигналу в цій околиці. В інших випадках ознакою корисного сигналу є різкі перепади яскравості.

2) Бінаризація. Операція порогового поділу, яка в результаті отримує бінарне зображення, має на меті радикальне зменшення кількості інформації, що міститься в зображенні. В процесі бінаризації вихідне півтонове зображення, що має певну кількість рівнів яскравості, перетворюється в чорно-біле зображення, пікселі якого мають тільки два значення 0 і 1.

3) Виділення контурів. Перетворення необхідне в тому випадку, коли є досить складне зображення і, використовуючи малі інструменти бібліотеки комп'ютерного зору, необхідно виділити об'єкти на цьому зображенні та потім з ними працювати.

4) Міжкадрова різниця. Використовується поблочне порівняння кадрів, при якому зображення розбивається на окремі блоки. Порівняння між кадрами проводиться на рівні блоків відповідно до обраного критерієм. Перший кадр (базовий) стискається незалежно від інших. Наступні кадри стискаються тільки в обсязі блоків, що змінилися. Блоки, які вважаються незмінними запозичуються з попереднього (базового) кадру [4].

Програмна реалізація визначення руху об'єктів у відеопотоці

Розроблена програма написана на мові C++ з використанням бібліотеки OpenCV та компілятора MSVC 2017. Визначення руху об'єктів у відеопотоці реалізовано в двох варіантах:

- на основі порівняння двох кадрів;
- на основі модуля «`bioinspired`».

Кожен з цих варіантів отримує зображення з відеокамери по 30 кадрів за секунду. Приклад зображення представлений на рисунку 1.



Рис. 1. Початковий кадр



Рис. 2. «Градація сірого» та фільтрація «Розмивання Гауса»

Програмна реалізація визначення руху об'єктів в відеопотоці на основі порівняння двох кадрів

Програмна реалізація складається з таких етапів:

1) перетворення вхідного зображення з RGB в «Градацію сірого»;

2) застосування методу фільтрації «Розмивання Гауса» (рис. 2.), щоб знизити рівень шуму.

3) Отримання різниці між зображенням з пункту 2 та попереднім зображенням (рис. 3). Для цього потрібно поділити зображення на блоки та отримати середнє значення кольору для кожного блоку. Ділити на блоки потрібно з урахуванням оптимізації кількості обчислень. Наприклад, якщо зображення розміром 640x480 пікселів поділити на блоки розміром 10x10 і отримати з кожного блоку кольори 25 пікселів, то в результаті замість 300000 пікселів і ітерацій для аналізу отримаємо 3000 ітерацій і 75000 пікселів для аналізу [9]. Порівнюємо дві таблиці кольорів (від попереднього та поточного кадру) і отримуємо різницю кольорів по кожному блоку в третій таблиці (рис. 4).

4) Порівняння значення отриманої різниці з деяким пороговим значенням. Експериментально було обрано число 25. Якщо значення більше ніж порогове то такий піксель належить рухомому об'єкту, інакше значення дорівнює 0 (рис. 5.).



Рис. 3. Різниця між зображеннями

5	3	1	—	2	3	7	=	3	0	6
2	4	6		2	6	4		0	2	2
3	6	1		1	2	3		2	4	2
Кадр 1				Кадр 2				Результат		

Рис. 4. Різниця між таблицями кольорів



Рис. 5. Різниця між зображення після застосування порогового значення

Програмна реалізація визначення руху об'єктів в відеопотоці на основі модуля «bioinspired»

Модуль bioinspired[5] містить клас Retina, в якому знаходяться просторово-часовий фільтр двох інформаційних каналів (parvocellular pathway і magnocellular pathway) моделі сітківки ока.

Реалізація складається з таких етапів:

- 1) Перетворення вхідного зображення з RGB в «Градацію сірого».
- 2) Запуск модулю «magnocellular» (рис. 6.).
- 3) Перевірка медіанного фільтру від ентропії, для того щоб при відсутності будь якого руху не реагувати на шум. В дослідженні було використано значення 0,60.

4) Порівняння результату з пороговим значенням 90 (рис. 7.).

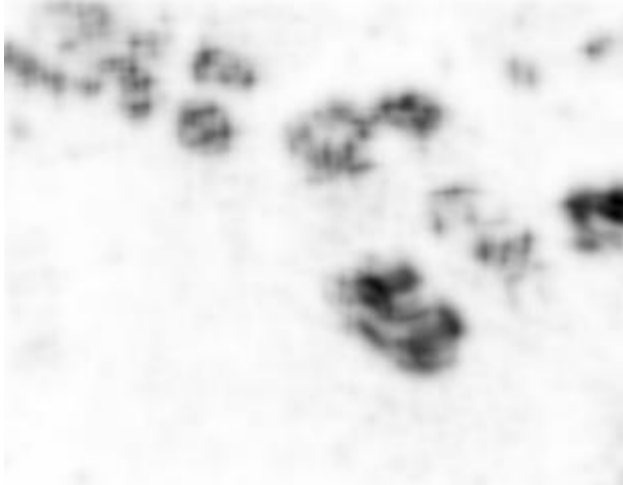


Рис. 6. Результат модулю «magnocellular»



Рис. 7. Результат після застосування порогового значення

Визначення інтенсивності транспортного потоку

В дослідженні для обох реалізацій методу детекції руху було використано камеру, що знаходиться в м. Києві на перехресті вулиць «Олени Теліги» та «Дорогожицька». На рис. 5, 7 представлені результати методу детекції руху. Для того, щоб визначити кількість машин, які проїхали по одній полосі дороги розроблено алгоритм, блок-схема якого представлена на рисунку 8.

Алгоритм застосовується для кожного зображення отриманого в результаті методу детекції руху. В даному алгоритмі кожен піксель перевіряється на належність відрізка MN, який відповідає одній смузі руху, та складається з точок M(x₁;y₁) та N(x₂;y₂).

Для перевірки належності точки відрізка MN використовується формула:

$$\frac{a-x_1}{x_2-x_1} = \frac{b-y_1}{y_2-y_1} \quad (1)$$

де *a*, *b* – координати пікселя, *x*₁, *y*₁ – координати точки *M*, *x*₂, *y*₂ – координати точки *N*.

Якщо точка (*a*, *b*) задовольняє рівності (1), то автомобіль перетнув уявну смугу. З значення лічильника *count* машин, що перетнули смугу, збільшується на одиницю тільки за умови, що попередня машина вже перетнула смугу (див.

рис. 8). Одразу після того, як фіксується перетин автомобілем уявної смуги, програма переходить в режим очікування і не фіксує нових автомобілів для того, щоб не фіксувати перетин смуги новим автомобілем, доки попередній автомобіль повністю не перетне смугу.

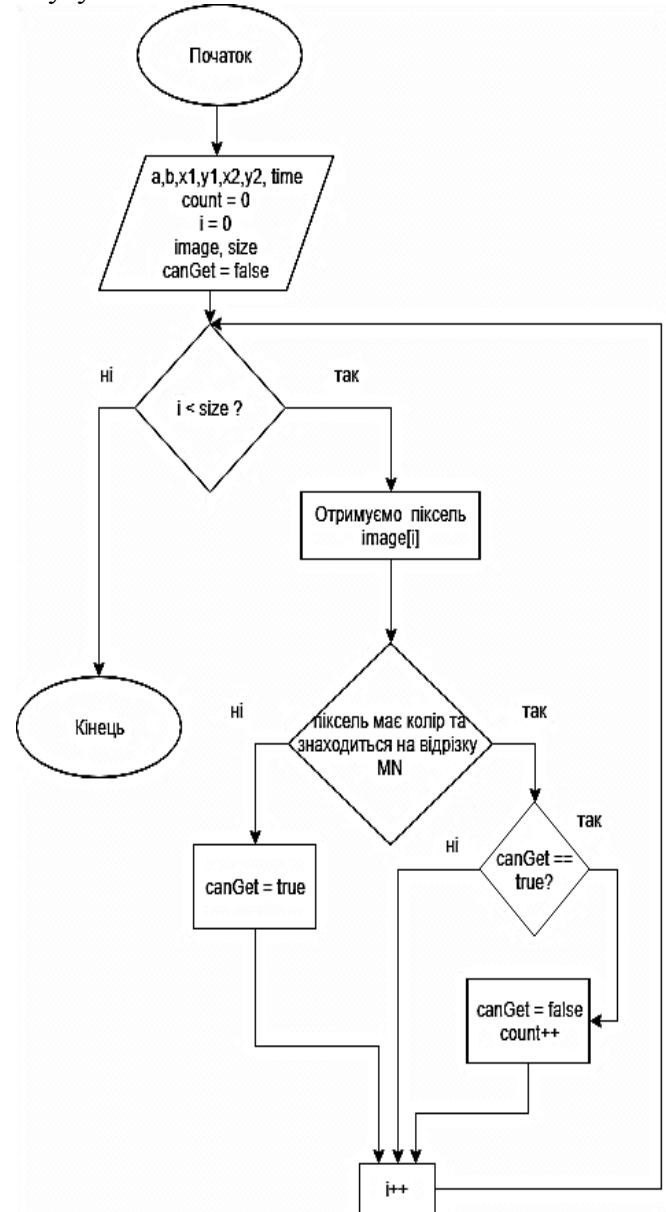


Рис. 8. Блок схема алгоритму визначення кількості машин

Результати дослідження

В таблиці 1 показано порівняння результатів застосування методу визначення інтенсивності транспортних потоків у двох реалізаціях. Відносна похибка визначається за формулою:

$$\varepsilon = \frac{\Delta\alpha}{\alpha} \cdot 100\%$$

Отримана при використанні модулю bioinspired похибка 4,7% є прийнятною і свідчить про спроможність цього методу надавати якісну інформацію про інтенсивність

транспортного потоку. Метод порівняння меншу кількість пікселів рухомих об'єктів, що кадрів дає в 3,5 рази більшу похибку через було отримано з порівняльних таблиць.

Табл. 1. Результати визначення інтенсивності транспортного руху

	Кількість машин	Отримана кількість машин	Час	Інтенсивність	Отримана інтенсивність	Відносна похибка інтенсивності
Порівняння кадрів	468	547	3	156	182,33	16,88%
Модуль bioinspired	468	490	3	156	163,33	4,70%

Висновок

Проведено аналіз існуючих методів визначення інтенсивності транспортних потоків.

Розроблено програмний компонент для визначення інтенсивності дорожнього руху в двох реалізаціях. Експериментально доведено значну перевагу реалізації на основі модуля bioinspired над реалізацією на основі порівняння двох кадрів. Програмний

компонент визначення інтенсивності транспортного руху, який розроблений, є складовою частиною інформаційної системи управління транспортним рухом. Якісна обробка інформації, яка надходить з відеокамер спостереження, розташованих на перехрестях, є потужним джерелом інформації для систем управління рухом з метою попередження негативних ситуацій таких, як призупинення руху внаслідок аварії чи ремонту дороги.

Список посилань

1. Стеценко І.В. Петрі-об'єктна модель системи управління транспортним рухом / І. В. Стеценко // Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка: збірник наукових праць. – Київ, 2011. – № 54. – С. 116–125.
2. Стеценко І.В. Інформаційна технологія визначення оптимальних параметрів управління транспортним рухом через світлофорні об'єкти міста / Стеценко І.В., Батора Ю.В. // Математичні машини і системи. – Київ, 2007. – №.3,4 – С.211-217.
3. Акулов В. В. Анализ методов учёта интенсивности движения на автомобильных дорогах // Наукоедение. – 2012. – №4. – С. 1–7.
4. Ярышев С.Н. Цифровые методы обработки видеoinформации и видеоаналитика: учебное пособие. СПб.: СПбГУ ИТМО, 2011. 83 с
5. Retina : a Bio mimetic human retina model [Електронний ресурс]. - Режим доступу: <https://docs.opencv.org/3.0-beta/modules/bioinspired/doc/retina/index.html#retina>
6. Héroult J. Vision: Images, Signals and Neural Networks-Models of Neural Processing in Visual Perception / Héroult J. // World Scientific, 2010.
7. Chaix De Lavarène B. Efficient demosaicing through recursive filtering / Chaix De Lavarène B. Alleysson D., Durette V., Héroult J. // In Image Processing, 2007. ICIIP 2007. IEEE International Conference on. 189с.
8. OpenCV [Електронний ресурс]. - Режим доступу: <https://docs.opencv.org/3.2.0/index.html> .
9. Алгоритм определения движения через сравнение двух кадров [Електронний ресурс]. URL: <https://habr.com/post/134635>.