Існуючі різні методи бінаризації зображень можна умовно поділити на глобальні (порогові) та локальні (адаптивні).

У загальному сенсі завдання виявлення об'єктів полягає у встановленні наявності на зображенні об'єкта, що має деякі певні характеристики. Такою характеристикою може бути, наприклад, яскравість зображення.

Локальні (адаптивні) методи бінаризації розбивають зображення на декілька областей, для кожної з яких необхідно обчислити поріг, ґрунтуючись на інформації про інтенсивність пікселів. Алгоритми даного класу передбачають розбиття зображення на блоки певного розміру, при цьому розмір блоку має бути мінімальним, але достатнім для збереження вихідних особливостей та деталей зображення. Однак при цьому блоки мають бути настільки більшими, щоб шуми впливали на результат мінімально. Найбільш популярними адаптивними методами бінаризації зображень є метод Ніблека, метод Бернсена, метод Ейквеля, метод Саувола та метод Крістіана [9]. Адаптивну бінаризацію можна рекомендувати у разі, якщо необхідно обробити напівтонові зображення невисокої якості. Однак, у разі неоднорідностей фону з низькою контрастністю можлива поява хибних об'єктів. Адаптивні способи та пристрої, що їх реалізують, добре підходять для застосувань з обмеженими обчислювальними можливостями та/або обмеженими доступними ресурсами, що використовуються для виконання обробки зображень [10].

Згідно з глобальними (пороговими) методами одним з способів виявлення об'єктів є вибір порога за яскравістю, або гранична класифікація (thresholding). Сенс такого порогу полягає в тому, щоб розділити зображення на світлий об'єкт (foreground) та темне тло (background). Тобто об'єкт – це сукупність тих пікселів, яскравість яких перевищує поріг (I > T), а тло — сукупність інших пікселів, яскравість яких нижче за поріг (I < T). Питання постає у тому, як обрати ключовий параметр – поріг T.

Одним з найбільш популярних методів глобальної бінаризації зображень є метод Оцу [11], запропонований японським вченим Нобуюкі Оцу в 1979 р.

Алгоритм Оцу складається з таких основних кроків:

1. Розрахуйте гістограму інтенсивності зображення.
2. Розрахуйте загальну кількість пікселів у зображенні.
3. Для кожного значення порогу (від 0 до максимальної інтенсивності пікселів) розрахуйте вагу кожної частини зображення (до та після порогу), а також середні значення інтенсивності для кожної частини.
4. Для кожного значення порогу розрахуйте дисперсії кожної частини (відносно середнього значення інтенсивності кожної частини) та загальну внутрішньокласову дисперсію, використовуючи вагу кожної частини.
5. Знайдіть поріг, для якого міжкласова дисперсія максимальна.
6. Застосуйте отриманий поріг до зображення.

Недоліком методу Оцу можливість втрати дрібних деталей, але для обліку кількості людей на зупинках дрібні деталі не є важливими, тому для планування пасажироперевезень алгоритм Оцу дозволяє знайти оптимальний поріг для бінаризації зображення. У цьому випадку він забезпечує точну та надійну бінаризацію зображення, що допомагає покращити якість подальшої обробки зображення.

До того ж, у глобальних (порогових) методах бінаризації визначені порогові значення застосовуються до піксельних значень градації сірого (gray scale), щоб отримати набір двійкових піксельних значень [12]. Але сучасні системи відеоспостереження базуються на IP-камерах, які формують відеострім та/або зображення у кольоровому форматі. Тому при розроблення спеціалізованого ПЗ необхідно передбачити при обробленні кольорових зображень, отриманих з IP-камер систем Smart City, попередню конвертацію таких зображень у відтінки сірого. Слід зауважити, що в залежності від погодних умов якість отриманих знімків може дуже відрізнятись один від одного. В такому разі перед формуванням діаграми яскравості необхідно застосувати до зображення один з алгоритмів покращення. До варіантів такого покращення входять покрокове лінійне розтягування, лінійне вирівнювання, глобальне вирівнювання гістограми, модифікація роздільної здатності тощо [13].

Для генерування гістограми яскравості та розрахунку загальної кількості пікселів у зображенні був розроблений програмний застосунок у середовищі розробки Webstorm, для чого було встановлено кросплатформне середовище Node.js 18.18.2 LTS для виконання JavaScript з відкритим вихідним кодом [14] та менеджер пакетів JavaScript – Node Package Manager (npm), – що керує залежностями в проєктах, написаних на Node.js, та дозволяє завантажувати пакети у застосунок з хмарного сервера ***npm*** [15; 16].

Послідовність кроків, шо виконується програмою, яка буде приймати кольорове зображення наповненості зупинки громадського транспорту людьми та генерувати гістограму яскравості цього зображення:

Крок 1. Прийняти вхідний параметр, який є шляхом до зображення.

Крок 2. Завантажити зображення за заданим шляхом.

Крок 3. Використовуючи пакет ***sharp,*** перетворити зображення на градації сірого.

Крок 4. Отримати масив значень яскравості пікселів зображення.

Крок 5. Створити гістограму на основі отриманих значень яскравості, використовуючи пакет ***D3.js.***

Крок 6. Зображення гістограми зберегти у файл.

Застосунок складається з двох модулів – ***get-brightness.js*** та ***index.js***.

Перший модуль містить в собі всю основну логіку застосунку, а саме: приймає зображення, обробляє його за допомогою бібліотеки ***sharp***, генерує гістограму яскравості за допомогою бібліотеки ***d3-node*** та зберігає її в форматі *png*.

Другий модуль слугує коренем застосунку, він імпортує в себе перший модуль. Спочатку, за допомогою бібліотеки ***readline*** зчитується з консолі шлях до зображення, з якого необхідно отримати гістограму. Після цього, запускаємо функцію ***get-brightness*** (перший модуль), в який в якості аргументу передається шлях до файлу.

Діаграма компонентів програми наведена на рис. 2.

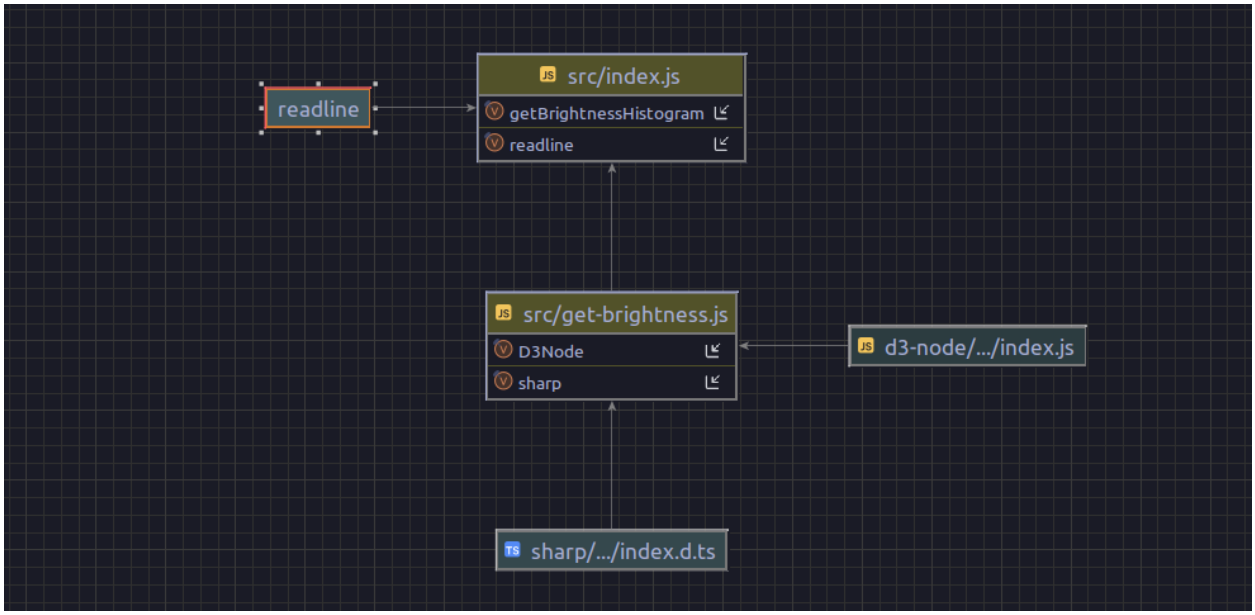

Рис. 2. Діаграма компонентів програми

Для запуску програми потрібно виконати наступні команди:

1) використовуючи термінал, перейти в кореневу директорію застосунку;
2) виконати команду ***npm i*** (для встановлення залежностей);
3) виконати команду ***npm run start***;
4) слідувати вказівкам на екрані;
5) після успішного виконання програми результат буде збережений в корені застосунку у директорії ***output***.

Процес запуску розробленого застосунку наведено га рис. 3.

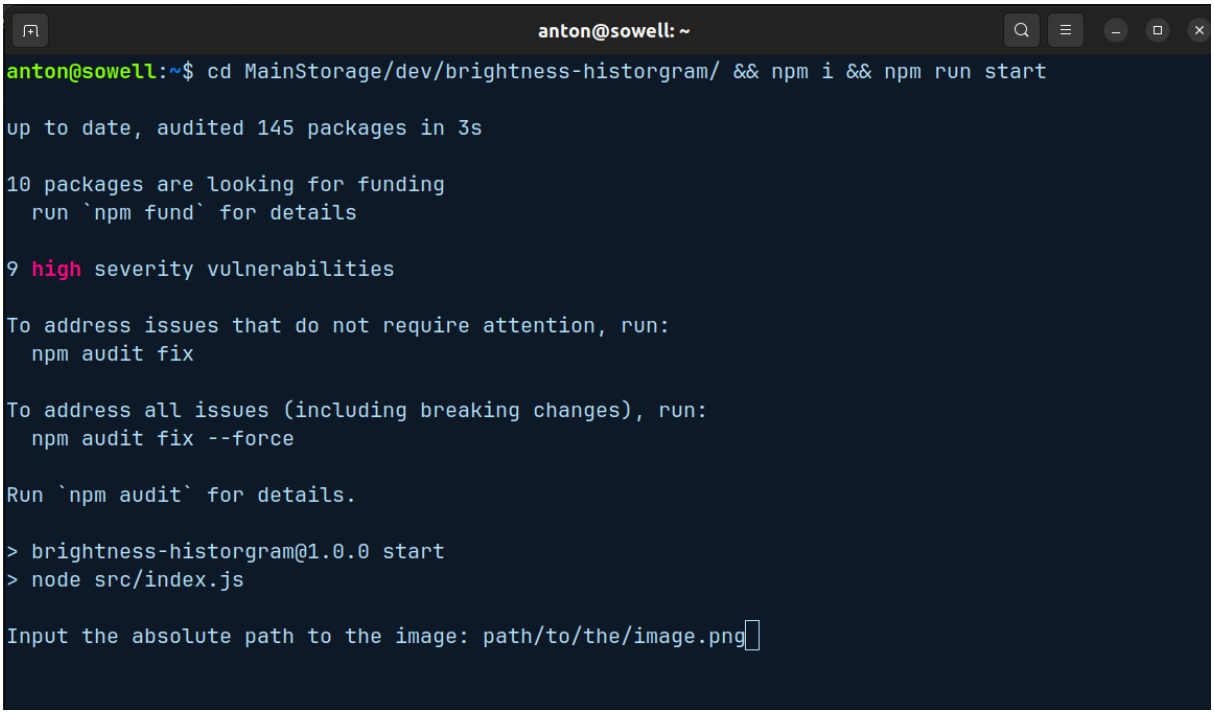

Рис. 3. Приклад запуску застосунку

Приклад розрахунку загальної кількості пікселів із фото людей на зупинці наведено на рис. 4–5.

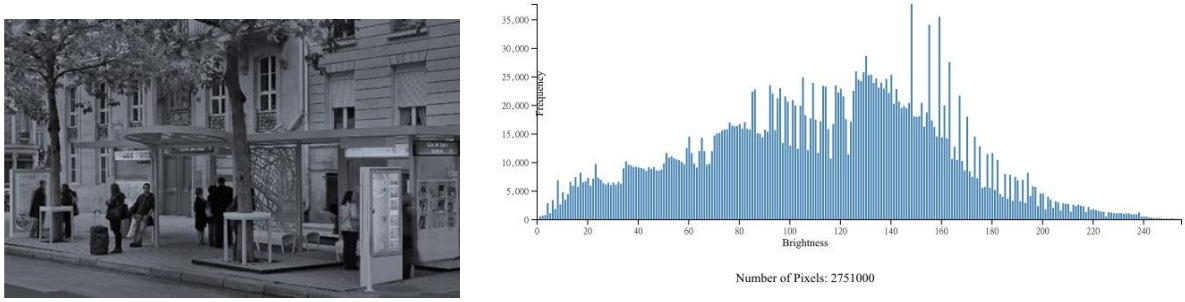

Рис. 4. Розрахунок загальної кількості пікселів з gray-scale фото людей на зупинці транспорту

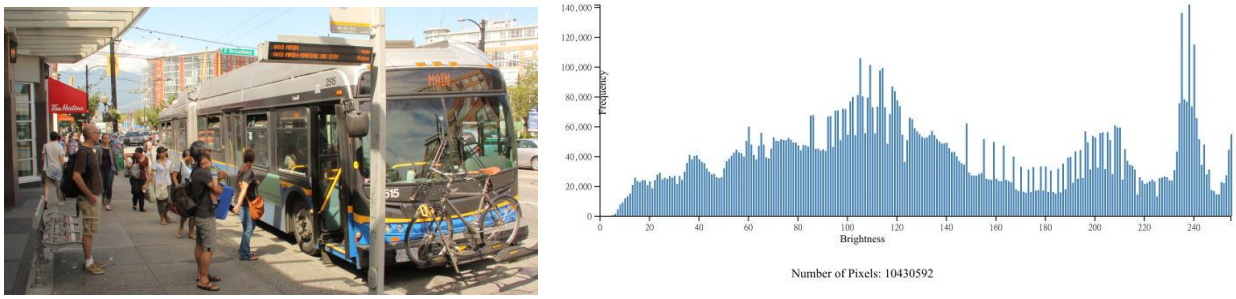

Рис. 5. Розрахунок загальної кількості пікселів з кольорового фото людей на зупинці транспорту

На графічному представленні гістограми яскравості зображень (рис. 4–5) вісь *x* представляє значення яскравості (від 0 до 255), а вісь *y* представляє кількість пікселів у зображенні, які мають це значення яскравості.

Висновки

Моніторинг наповненості зупинок пасажирського транспорту дозволяє операторам ефективно планувати рух транспорту, уникати переповнених автобусів чи трамваїв, а також забезпечити комфортні умови для пасажирів. Крім того, ця інформація може бути використана для аналізу популярності різних зупинок та визначення потреб у розвитку громадського транспорту. Все більше поширення інноваційних технологій у рамках комп'ютерних систем «розумного міста» дозволяє реалізувати облік наповненості людьми зазначених зупинок за допомогою IP-камер системи відеоспостереження Smart City. Проведений аналіз використання глобальних (порогових) та локальних (адаптивних) методів бінаризації зображень, отриманих з IP-камер, показав, що адаптивну бінаризацію можна рекомендувати у разі, якщо необхідно обробити напівтонові зображення невисокої якості. Однак, при наявності неоднорідностей фону з низькою контрастністю можлива поява хибних об'єктів. Втрата дрібних деталей при застосуванні методів глобальної бінаризації зображень не впливає на оцінку завантаженості зупинок транспорту. Доведено доцільність використання алгоритму Оцу для знаходження оптимального порогу при бінаризації зображень у процесі планування пасажироперевезень. Зважаючи на те, що для отримання набору двійкових піксельних значень при застосуванні глобальних (порогових) методів бінаризації виконується аналізування зображень у градаціях сірого (gray scale), у розробленому застосунку на мові JavaScript передбачено конвертацію кольорових фото з IP-камер у Grayscale-зображення. Наведено приклад роботи розробленого застосунку при розрахунку загальної кількості пікселів із фото людей на зупинках міського транспорту.

Література

1. Чернов С. К., Савіна О. Ю. Метод формування ціннісно-орієнтованого портфеля проектів наукомісткого підприємства. Управління розвитком складних систем. 2018. № 34. С. 78–84. URL: https://repositary.knuba.edu.ua/server/api/core/bitstreams/6a9b1321-4dcf-4f00-975c-95167bc0bfad/content.

2. Доля О., Доля К. Методи рішення задач з організації пасажирських перевезень автомобільним транспортом. International Science Journal of Engineering & Agriculture. 2023. Vol. 2, no. 3. P. 101–119. DOI: 10.46299/j.isjea.20230203.10.

3. Chernov S., Titov S., Chernova L., Kunanets N., Chernova L. Determination of approaches for project costs minimization with use of dual problems. Econtechmod : an international quarterly journal. Polish Academy of Sciences. 2019. Vol. 08, No. 2. P. 61–68.

4. Дуда О., Станько А. Архітектура мережевої платформи моніторингу об'єктів у кіберфізичних системах «розумних міст». Вісник Хмельницького національного університету. Технічні науки. 2023. Т. 323, № 4. С. 123–130. DOI: 10.31891/2307-5732-2023-323-4-123-130.

5. Войтович О. А., Ткач В. О., Луб'яний П. В. Модель впровадження додаткових зупинок міського пасажирського транспорту. Вісник Херсонського національного технічного університету. Інженерні науки.

2020. Т. 4, № 75. С. 20–27. DOI: 10.35546/kntu2078-4481.2020.4.2.

6. Bieletska O., Liubyi Y., Ocheretenko S., Muzylyov D., Ivanov V., Pavlenko I. Approach to determine transport delays at unsignalized intersections. Communications – Scientific Letters of the University of Zilina. 2023. Vol. 25, no. 3. P. A124–A136. DOI: 10.26552/com.C.2023.052.

7. Royko Yu., Yevchuk Yu., Bura R., Velhan A. Minimization of public transport delays at arterial streets with coordinated motion. Transport Technologies. 2022. Vol. 3, no. 1. P. 14–29. DOI: 10.23939/tt2022.01.014.

8. Yang C., Guo H. A Lightweight semantic segmentation algorithm based on deep convolutional neural networks. Computational Intelligence and Neuroscience. 2022. Vol. 2022. Article ID 5339664. 9 pages. DOI: 10.1155/2022/5339664.

9. Eyupoglu C. Implementation of Bernsen's locally adaptive binarization method for gray scale images. The Online Journal of Science and Technology. April 2017. Vol. 7, Is. 2. P. 68–72.

10. Patent № US8,351,699 B2 USA. Int. CI (2006.01) G06K 9/00. Methods and apparatus for adaptive auto image binarization. D. Li, J. Elton, S. Steger; Assignee: Accusoft Co., Tampa, FL (US). Appl. No. 12/642,643; Filed: Dec. 18, 2009; Date of Pat.: Jan. 8, 2013. 17 p.

11. Shaikh A., Botcha S., Krishna M., Kumar S. Otsu based differential evolution method for image segmentation : preprint, 19 pages, October 2019. URL: https://www.researchgate.net/publication/364422460_Otsu_based_Differential_Evolution_Method_for_Image_Segmentation

12. Patent № US9,025,897 B1 USA. Int. CI (2006.01) G06K 9/00. Methods and apparatus for adaptive auto image binarization. J. H. Elton, S. A. Martucci; Assignee: Accusoft Co., Tampa, FL (US). Appl. No. 13/867,605; Filed: Apr. 5, 2013; Date of Pat.: May. 5, 2015. 16 p.

13. Pradeep, Namratha M., Manu G. V. Global and localized histogram equalization of an image. International Journal of Computational Engineering Research. Oct. 2012. Vol. 2, Is. 6. P. 238–252.

14. Node.js® is an open-source, cross-platform JavaScript runtime environment. URL: https://nodejs.org/en.

15. NPM: Public collection of packages of open-source code for Node.js. URL: https://www.npmjs.com/.

16. NPM Docs: Downloading and installing Node.js and npm. URL: https://docs.npmjs.com/downloading-and-installing-node-js-and-npm#using-a-node-version-manager-to-install-nodejs-and-npm.

References

1. Chernov S., Savina O. A method of formation of value-oriented project portfolio for a science-based enterprise. Management of the development of complex systems. 2018. Issue 34. P. 78–84. URL: https://repositary.knuba.edu.ua/server/api/core/bitstreams/6a9b1321-4dcf-4f00-975c-95167bc0bfad/content.

2. Doliia O., Doliia K. Methods of solving problems related to the organization of passenger transportation by road transport. International Science Journal of Engineering & Agriculture. 2023. Vol. 2, no. 3. P. 101–119. DOI: 10.46299/j.isjea.20230203.10.

3. Chernov S., Titov S., Chernova L., Kunanets N., Chernova L. Determination of approaches for project costs minimization with use of dual problems. Econtechmod : an international quarterly journal. Polish Academy of Sciences. 2019. Vol. 08, No. 2. P. 61–68.

4. Duda O., Stanko A. The network platform architecture for monitoring objects in cyberphysical systems of smart cities. Herald of Khmelnytskyi national university. Technical sciences. 2023. Vol. 323, Is. 4. P. 123–130. DOI: 10.31891/2307-5732-2023-323-4-123-130.

5. Voytovich O. A., Tkach V. O., Lubyany P. V. Model of implementation of additional stoppings city passenger transport. Herald of Khersonskyi national technical university. Engineering sciences. 2020. Vol. 4, Is. 75. P. 20–27. DOI: 10.35546/kntu2078-4481.2020.4.2.

6. Bieletska O., Liubyi Y., Ocheretenko S., Muzylyov D., Ivanov V., Pavlenko I. Approach to determine transport delays at unsignalized intersections. Communications – Scientific Letters of the University of Zilina. 2023. Vol. 25, no. 3. P. A124–A136. DOI: 10.26552/com.C.2023.052.

7. Royko Yu., Yevchuk Yu., Bura R., Velhan A. Minimization of public transport delays at arterial streets with coordinated motion. Transport Technologies. 2022. Vol. 3, no. 1. P. 14–29. DOI: 10.23939/tt2022.01.014.

8. Yang C., Guo H. A Lightweight semantic segmentation algorithm based on deep convolutional neural networks. Computational Intelligence and Neuroscience. 2022. Vol. 2022. Article ID 5339664. 9 pages. DOI: 10.1155/2022/5339664.

9. Eyupoglu C. Implementation of Bernsen's locally adaptive binarization method for gray scale images. The Online Journal of Science and Technology. April 2017. Vol. 7, Is. 2. P. 68–72.

10. Patent № US8,351,699 B2 USA. Int. CI (2006.01) G06K 9/00. Methods and apparatus for adaptive auto image binarization. D. Li, J. Elton, S. Steger; Assignee: Accusoft Co., Tampa, FL (US). Appl. No. 12/642,643; Filed: Dec. 18, 2009; Date of Pat.: Jan. 8, 2013. 17 p.

11. Shaikh A., Botcha S., Krishna M., Kumar S. Otsu based differential evolution method for image segmentation : preprint, 19 pages, October 2019. URL: https://www.researchgate.net/publication/364422460_Otsu_based_Differential_Evolution_Method_for_Image_Segmentation

12. Patent № US9,025,897 B1 USA. Int. CI (2006.01) G06K 9/00. Methods and apparatus for adaptive auto image binarization. J. H. Elton, S. A. Martucci; Assignee: Accusoft Co., Tampa, FL (US). Appl. No. 13/867,605; Filed: Apr. 5, 2013; Date of Pat.: May. 5, 2015. 16 p.

13. Pradeep, Namratha M., Manu G. V. Global and localized histogram equalization of an image. International Journal of Computational Engineering Research. Oct. 2012. Vol. 2, Is. 6. P. 238–252.

14. Node.js® is an open-source, cross-platform JavaScript runtime environment. URL: https://nodejs.org/en.

15. NPM: Public collection of packages of open-source code for Node.js. URL: https://www.npmjs.com/.

16. NPM Docs: Downloading and installing Node.js and npm. URL: https://docs.npmjs.com/downloading-and-installing-node-js-and-npm#using-a-node-version-manager-to-install-nodejs-and-npm.