

УДК 004.932

*В.О. Білозерський, К.Ю. Дергачов, Л.О. Краснов*

## АНАЛІЗ І ПОПЕРЕДНЯ ОБРОБКА ВІДЕОДАНИХ ДЛЯ ПІДВИЩЕННЯ ЯКОСТІ РОБОТИ СИСТЕМ ТЕХНІЧНОГО ЗОРУ

**Білозерський Владислав Олександрович**

Національний аерокосмічний університет ім. М.Є. Жуковського «Харківський авіаційний інститут», ORCID: 0000-0002-5503-3163,

*b1llays123@gmail.com*

**Дергачов Костянтин Юрійович**

Національний аерокосмічний університет ім. М.Є. Жуковського «Харківський авіаційний інститут», ORCID: 0000-0002-6939-3100,

*k.dergachov@khai.edu*

**Краснов Леонід Олександрович**

Національний аерокосмічний університет ім. М.Є. Жуковського «Харківський авіаційний інститут», ORCID: 0000-0003-2607-8423,

*leonid.krasnov.1947@gmail.com*

Запропоновано новий метод аналізу та оцінки мінливості вихідних даних у системах відеоспостереження під впливом негативних зовнішніх факторів. За результатами аналізу запропоновано алгоритми стабілізації параметрів відеопотоку, що приймається, і дано оцінку якості відеоданих у результаті їх попередньої обробки. Метою роботи є аналіз мінливості основних параметрів вхідних даних у системах відеоспостереження під дією негативних зовнішніх впливів, створення ефективних алгоритмів стабілізації параметрів відеопотоку, оцінка якості відеоданих у результаті їх попередньої обробки, перевірка достовірності запропонованого методу на реальних записах відеоспостережень. При аналізі відеозаписів або відео в реальному масштабі часу визначено найбільш універсальний, стійкий та інформативний показник якості, що дозволяє дати об'єктивну оцінку якості відео за різних умов зйомки та характеру освітленості сцени, а також розробити та програмно реалізувати алгоритми стабілізації параметрів відео. Проведено статистичний аналіз та обробку параметрів аналізованої відеопослідовності як випадкового просторово-часового процесу, алгоритми обробки відеоданих методами цифрової фільтрації та адаптивної стабілізації параметрів яскравості відеопотоку. Показано, що найбільш стійким показником якості потоку відео є середня яскравість кадру AFB (Average Frame Brightness). Запропоновано універсальний алгоритм просторово-часової обробки відеоданих, що формує з вихідного відеопотоку послідовність значень середньої яскравості кадрів AFB, а також алгоритми

© В.О. БІЛОЗЕРСЬКИЙ, К.Ю. ДЕРГАЧОВ, Л.О. КРАСНОВ, 2023

фільтрації та стабілізації яскравості відеопотоку. Запропоновано новий метод об'єктивного аналізу та оцінки параметрів вихідних даних відеоспостереження, алгоритми фільтрації та стабілізації яскравості відеопотоку. Якість запропонованих алгоритмів верифікована на реальних результатах відеоспостереження. Запропоновані алгоритми реалізовані у програмному середовищі Python із застосуванням функцій бібліотеки OpenCV.

**Ключові слова:** аналіз характеру змін яскравості відео, середня яскравість кадру у відеопотоці, цифрова фільтрація тренду яскравості відеопотоку, алгоритми стабілізації яскравості відеоданих.

### Вступ

В останні роки спостерігається суттєвий прогрес у створенні нових технологій штучного інтелекту на основі нейронних мереж із глибоким навчанням. Це значно сприяє успішному вирішенню багатьох завдань розпізнавання образів у різних додатках. За допомогою засобів технічного зору створені та успішно експлуатуються системи детектування та розпізнавання осіб (Face Detection and Recognition), системи оптичного розпізнавання текстів (OCR — Optical Character Recognition), системи розпізнавання дорожніх знаків (Detection and Recognition of Traffic Signs) та багато інших корисних додатків. Необхідно відзначити високу ефективність їхньої роботи — ймовірність правильного виявлення об'єктів спостереження та їхнього успішного розпізнавання наближається до одиниці за відсутності негативних факторів та зовнішніх впливів. Однак існує ціла низка причин (несприятливий геометричний фактор, погані та дуже мінливі умови освітлення сцени, наявність шумів, помилки операторів при проведенні зйомки тощо), що різко знижують ефективність виявлення та розпізнавання об'єктів. У деяких випадках це призводить навіть до збоїв та порушення нормального функціонування систем, що у більшості випадків неприпустимо.

Тому завдання дослідження нових методів та алгоритмів усунення зовнішніх негативних впливів на роботу засобів технічного зору залишається актуальною, і їхня роль тільки зростає в міру вдосконалення інтелектуальних систем розпізнавання образів. У даний час при проектуванні сучасних систем технічного зору сформувалося ціле наукове спрямування, присвячене попередній обробці вихідних даних. Це дозволяє захистити інформаційне ядро системи від негативного впливу зовнішніх факторів та підтримувати ефективність роботи на високому рівні.

Для підвищення якості роботи систем розпізнавання образів методами технічного зору ставилися такі завдання:

- провести об'єктивний аналіз мінливості основних параметрів вхідних даних у системах відеоспостереження під впливом негативних зовнішніх впливів;
- створити за результатами аналізу ефективні алгоритми стабілізації параметрів вихідного відеопотоку;
- дати оцінку якості відео в результаті їх попередньої обробки;
- написати мовою Python із використанням функцій бібліотеки OpenCV програмні коди цих алгоритмів;
- перевірити достовірність отриманих результатів на реальних записах відеоспостережень.

### Огляд публікацій, пов'язаних з роботою

Розвиток технологій виявлення та розпізнавання об'єктів за допомогою систем технічного зору на базі нейронних мереж в останні роки проводиться за двома основними напрямками. Перше, і головне, що цілком очевидно, — розвиток архітектури нейронних мереж, удосконалення способів їхнього навчання, створення нових способів управління наявними ресурсами, створення спеціалізованих бібліотек. Другий, не менш важливий напрямок, — застосування різно-

манітних засобів попередньої обробки вихідних даних для компенсації негативних зовнішніх впливів. Це дозволяє використовувати ресурси нейронних мереж значно повніше та ефективніше. Тому розглянемо найцікавіші публікації з цієї проблематики докладніше.

Одним з найефективніших методів розв'язання задач розпізнавання образів є використання згорткових нейронних мереж CNN (Convolutional Neural Network). Це основний інструмент класифікації та розпізнавання зображень та сигналів. Є безліч варіантів застосування CNN, такі як Deep Convolutional Neural Network (DCNN), Region-CNN (R-CNN), Fully Convolutional Neural Networks (FCNN) та ін.

Одна з перших робіт, присвячених новій архітектурі згорткових нейронних мереж [1], опублікована в 1998 р. і отримала назву CNN (Convolutional Neural Network). У 2012 р., після змагань з комп'ютерного зору, в рамках проекту ILSVRC (ImageNet Large Scale Visual Recognition Challenge — кампанія з широкомасштабного розпізнавання образів на основі бази даних ImageNet [2]) розроблена нейронна мережа AlexNet [3]. Вона класифікує мільйони зображень із тисяч різних категорій з помилкою лише 15,8 % [4]. AlexNet — переможець ILSVRC-2012. Це проста, але потужна мережева архітектура зі згортковими рівнями, за якими йдуть повністю пов'язані вищі рівні. Така архітектура зазвичай використовується для глибокого навчання завдань комп'ютерного зору. Ці результати були стимулом до впровадження CNN у різних додатках з розпізнавання образів у системах технічного зору.

Це суттєво зменшує навантаження розробників програмного забезпечення з навчання нейронних мереж. Прикладом є розширення функціональних можливостей бібліотеки OpenCV, що широко використовується для кращого використання можливостей мови Python.

Ієрархічне узагальнення та управління ресурсами підтримки нейронних мереж глибокого навчання реалізовано у бібліотеці OpenCV — створено пакет функцій та алгоритмів керування параметрами мереж глибокого навчання. У версії OpenCV 3.1 введено модуль глибоких нейронних мереж (DNN), що реалізує прямий зв'язок з нейромережами, попередньо глибоко навчений з використанням популярних фреймворків.

Було створено низку популярних бібліотек на формування сучасних нейронних мереж. Це бібліотеки Caffe [5] та TensorFlow [6], наукова обчислювана платформа Torch/Pytorch [7], фреймворк нейронної мережі Darknet [8], бібліотека Keras [9]. В OpenCV 3.3 статус модуля Deep learning підвищений із статусу репозиторію `opencv_contrib` до статусу основного сховища [10]. У цій версії значно збільшено і швидкодію модуля DNN. Основна можливість DNN полягає, звичайно, у завантаженні та запуску нейронних мереж (inference). Модель може бути створена за допомогою будь-якого з трьох фреймворків глибинного навчання (Caffe, TensorFlow або Torch). Спосіб завантаження моделі та її використання не залежить від того, за допомогою якого фреймворку вона створена. У модулі DNN підтримуються всі основні шари: починаючи від базових (Convolution і Fully connected) і закінчуючи більш спеціалізованими — загалом понад 30. Крім підтримки окремих шарів, важлива також підтримка конкретних архітектур нейронних мереж. Модуль DNN містить приклади для класифікації (AlexNet, GoogLeNet, ResNet, SqueezeNet), сегментації (FCN, ENet), детектування об'єктів (SSD).

Бібліотеки та фреймворки під управлінням модуля DNN OpenCV успішно використовуються для створення та навчання нейронних мереж у задачах розпізнавання образів.

Необхідно зазначити, що останнім часом помітно зріс інтерес і до реалізації процедур попередньої обробки вихідних даних підвищення ефективності засобів

розпізнавання образів за допомогою нейронних мереж. Це сприяло величезному потоку нових публікацій. Зупинимося докладніше на деяких із них.

Так, наприклад, великий інтерес має робота, присвячена аналізу та особливостям застосування нейронних мереж для інтелектуальної обробки відеопотоків у системах технічного зору [11]. Показано, що процес інтелектуальної обробки відеоданих складається з багатьох етапів обробки зображень, одним з яких є обробка з використанням нейронних мереж як інтелектуальних компонентів. Наведена дуже корисна модель обробки відеозображень, що включає попередню обробку відеоданих, пошук групових об'єктів, їх класифікацію і розпізнавання.

У роботах [12, 13] представлені результати досліджень алгоритмів попередньої обробки відеопотоку для промислового контролю та пропонується оптимізований для сприйняття метод попередньої обробки з використанням сучасної моделі візуальної системи людини (HVS), що підходить для стиснення відео. Модель оптимізована для кількох відстаней перегляду, щоб враховувати сценарії реального світу. Для оцінки запропонованої попередньої обробки проведена велика суб'єктивна та об'єктивна оцінка. Заслуговує на увагу і дослідження з оцінки сприйняття попередньої обробки для транскодування відео [14]. Розглядаються різні підходи для створення бази даних попередньо оброблених та транскодованих відео. Проведено експеримент, який показує, що в порівнянні з відео, відправленим кодеку безпосередньо з тим же бітрейтом, методи попередньої обробки дійсно покращують якість сприйняття.

У роботі [15] запропоновано попередню обробку відеоспостереження за дорожнім рухом. Це швидкий метод виявлення та відстеження транспортних засобів у режимі реального часу. Попередньо обробляється тільки одне зображення з високою роздільною здатністю кожного виявленого транспортного засобу разом з його траєкторією, що знижує вимоги до пропускної спроможності систем зв'язку. У роботі [16] досліджено гібридну архітектуру попередньої обробки відеокадрів для HD-відео. Описано архітектуру, що дозволяє безпосередньо генерувати ряд примітивів обробки зображень з відеокadra, коли він проходить через стандартний конвеєр обробки зображень.

Предметом дослідження у статті [17] є сучасна концепція підвищення якості роботи систем технічного зору за рахунок використання набору різних алгоритмів попередньої обробки зображень на розсуд користувача. Синтезовано алгоритми, що компенсують зовнішні негативні впливи (несприятливий геометричний фактор, погані умови освітлення під час відеозйомки та фотографування, вплив шумів та ін.).

Проведений аналіз наочно демонструє актуальність досліджень, які принесуть безперечну користь на практиці.

### **Аналіз параметрів досліджуваних відеоданих**

При вирішенні різних завдань виявлення і розпізнавання об'єктів за допомогою відеоспостереження всі види збурюючих впливів вносять свій, цілком певний, негативний внесок у якість вихідних відеоданих. Однак найістотнішим чинником погіршення роботи відеосистем є несприятливі умови освітленості сцени. Це вкрай ускладнює застосування алгоритмів бінаризації з відсіканням за порогом яскравості, алгоритмів визначення меж об'єктів при обробці зображень окремих фреймів. Але, крім цього, з часом змінюється загальний рівень освітленості. Іншими словами, зміни яскравості відеоданих можна розглядати як випадковий нестаціонарний процес, що містить швидкі (високочастотні) компоненти і складові, що повільно змінюються (низькочастотні), обумовлені змінами загального рівня освітленості.

При аналізі відеозаписів або відео в реальному масштабі часу необхідно використовувати найбільш універсальний, стійкий та інформативний критерій

якості, який дозволяє об'єктивно оцінити якість відео за різних умов зйомки та характеру освітленості сцени. Автори вважають, що найбільш доречним та чи не єдиним стійким показником якості є середня яскравість кадру AFB (Average Frame Brightness). Зазвичай яскравість пікселів відео прийнято кодувати у числовому форматі uint8. Отже, значення параметра AFB перебувають у діапазоні [0...255]. Не менш важливими є і показники зміни середньої яскравості всієї послідовності відео, яка строго періодична. Нагадаємо, що стандарт швидкості запису відео характеризується кількістю кадрів за секунду (fps — frames per second). Найчастіше використовують fps = 25, 30 чи 60.

На думку авторів, для отримання об'єктивної оцінки зміни яскравості відео доцільно простежити залежність середньої яскравості кадрів AFB від часу та розрахувати статистичні характеристики цього випадкового процесу (середнє значення на всьому інтервалі аналізу або окремих ділянках, дисперсії відхилення яскравостей від середнього значення, гістограми розподілу параметра AFB).

Як правило, записи відео представляються в ортогональному трикомпонентному кольоровому просторі RGB. Цей простір є звичним і зручним для перегляду, але окремі компоненти кадру R, G і B не містять повної інформації про яскравість пікселів зображення. Цей недолік прийнято долати шляхом переведення зображень з простору RGB в простір HSV. Нагадаємо, що HSV — кольорова модель, заснована на трьох характеристиках кольору — кольоровому тоні (Hue), насиченості (Saturation) та значенні кольору (Value), який також називають яскравістю (Brightness). Ці властивості мають такі характеристики.

Hue — кольорний тон (наприклад, червоний, зелений чи синій). Варіюється в межах 0–360°, але іноді наводиться в діапазон 0–100 або 0–1. У Windows весь кольорний спектр ділиться на 240 відтінків (що можна спостерігати в редакторі палітри MS Paint), тобто Hue зводиться до діапазону 0–239 (відтінків 240 відсутній, оскільки він дублював би 0).

Saturation — насиченість. Варіюється в межах 0–100 або 0–1. Чим більший параметр, тим чистіший колір, тому цей параметр іноді називають чистотою кольору. Зменшення параметру наближає колір до нейтрального сірого.

Value — значення кольору або Brightness — яскравість. Також представляється в межах 0–100 або 0–1.

Перехід з кольорового простору RGB в простір HSV дозволяє просто оцінити середній рівень яскравості кадру за компонентом яскравості V, який обчислюється як

$$AFB = \frac{1}{MN} \sum_i \sum_j V(i, j), \quad (1)$$

де  $V(i, j)$  — двовірний масив чисел, що визначають яскравість пікселів зображення кадру розміром  $M \times N$ . Зазначимо, що обчислення AFB — досить складна процедура, оскільки кількість арифметичних операцій різко зростає зі збільшенням роздільної здатності кадру відео, стандартний кадр відео розмірністю 1920 x 1080 містить близько 2 млн пікселів. Однак ці складнощі можна подолати, використовуючи стандартні програмні засоби для обчислення середніх значень двовірних матриць, які оптимізовані за швидкістю.

Аналіз параметрів досліджуваних відеоданих проводився з використанням ресурсів мови програмування Python та бібліотеки OpenCV. Для стислості викладу не наводитимемо повні фрагменти програмного коду, а опишемо лише процедури переходу з кольорового простору RGB в простір HSV та програмну функцію визначення середньої яскравості AFB.

Перетворення зображення з формату BGR на HSV здійснюється за допомогою функції `cv2.cvtColor(img, cv2.COLOR_BGR2HSV)`. Нагадаємо, що в бібліотеці

OpenCV порядок прямування колірних компонентів зображення змінено з RGB на BGR.

Функція обчислення середньої яскравості кадру AFB показана на рис. 1 як фрагмент програмного коду. Все це виконується досить просто, тому немає потреби пояснювати роботу програми, оскільки в її тексті зроблено докладні коментарі. Зазначимо лише, що це повністю відповідає алгоритму обчислення AFB за формулою (1).

```

51 # Average frame brightness calculation
52 def brightness_avg(image_hsv):
53     # Getting separate channels
54     h, s, v = cv2.split(image_hsv)
55     # Calculate the average frame brightness as a result
56     return np.mean(v)

```

Рис. 1

Як зазначалося раніше, для повноти аналізу необхідно досліджувати залежність середньої яскравості кадрів AFB від часу і розрахувати статистичні характеристики цього випадкового процесу (середнє значення за усім інтервалом аналізу чи окремих ділянках відеопослідовності). Для цього запропоновано універсальний алгоритм просторово-часової обробки відеоданих, що формує з вихідного відео послідовність значень середньої яскравості кадрів AFB. Структура цього алгоритму наведена на рис. 2.

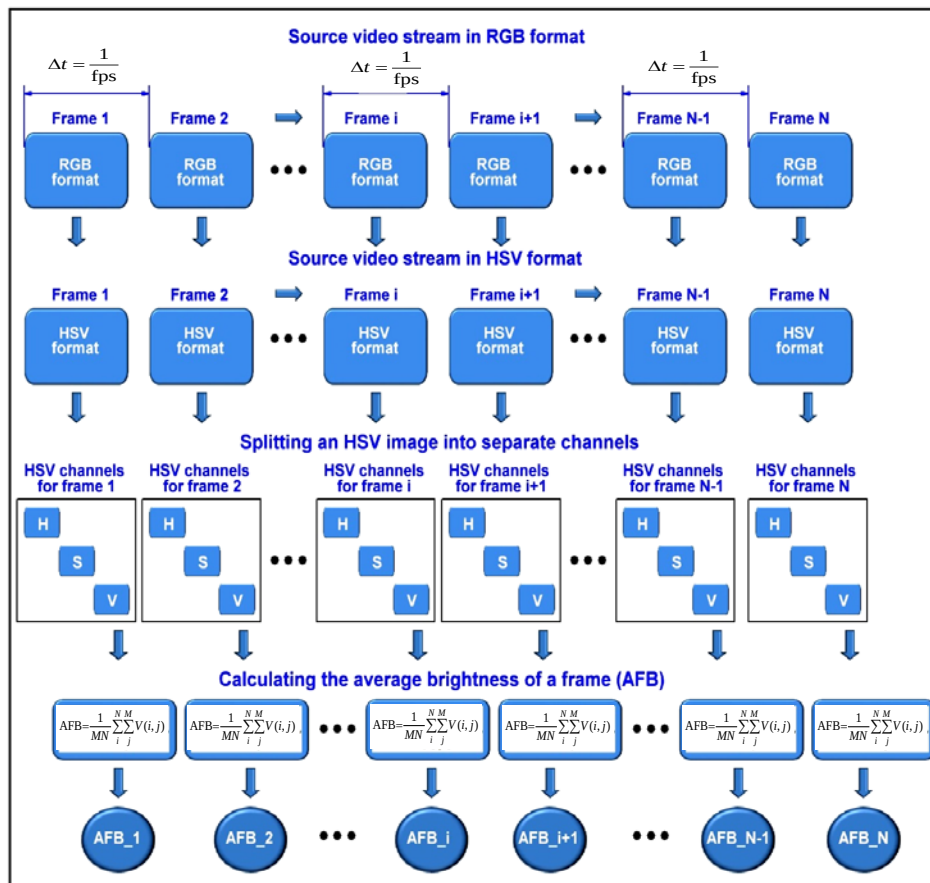


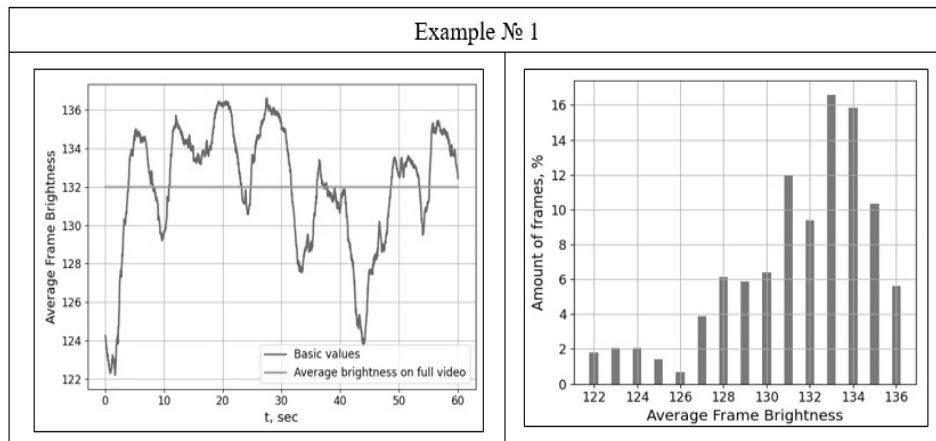
Рис. 2

Навіть при поверхневому візуальному аналізі привертає увагу велика мінливість характеристик цих відеозаписів. Це різний рівень освітленості сцени, різна щільність і швидкість руху транспорту, кольорове різноманіття об'єктів

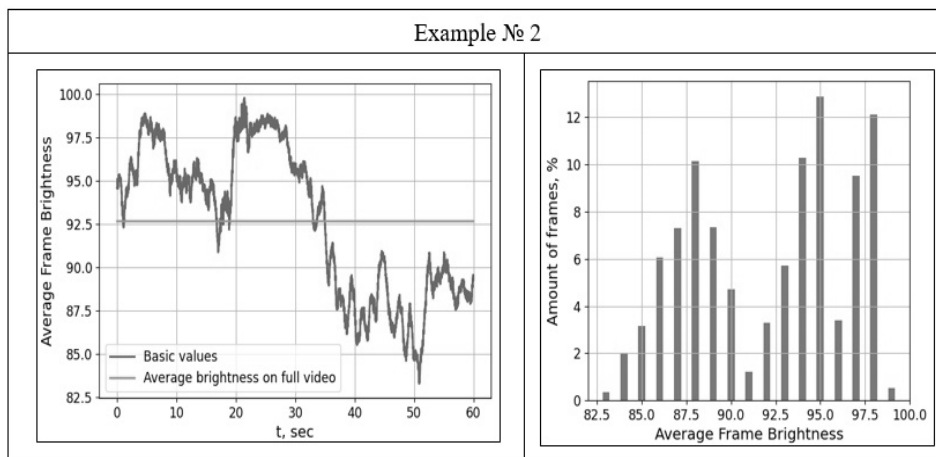
у зоні огляду та наявність сигнальних маячків, які поступово виходять із зони видимості). Тому результати розрахунку змін середньої яскравості кадрів протягом усього запису дозволяють об'єктивно судити про характеристики відеоданих. Ці результати показано на рис. 4. Крім графіків змін AFB, практичний інтерес викликають і нормовані гістограми розподілу значень AFB у відповідних діапазонах яскравості. Вони наочно показують рівень розкиду яскравостей кадрів щодо середніх значень.

| Example № 1                                                                         |                                                                                                                                                                                   |
|-------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   | <p>Analyzed<br/>video sequence<br/>Video format – mp4<br/>Frame sizes:<br/>width x height<br/>640 x 360<br/>frames per second<br/>(fps): 25.0<br/>number of frames:<br/>1501</p>  |
| a                                                                                   |                                                                                                                                                                                   |
| Example № 2                                                                         |                                                                                                                                                                                   |
|   | <p>Analyzed<br/>video sequence<br/>Video format – mp4<br/>Frame sizes:<br/>width x height<br/>3840 x 2160<br/>frames per second<br/>(fps):60.0<br/>number of frames:<br/>3600</p> |
| б                                                                                   |                                                                                                                                                                                   |
| Example № 3                                                                         |                                                                                                                                                                                   |
|  | <p>Analyzed<br/>video sequence<br/>Video format – mp4<br/>Frame sizes:<br/>width x height<br/>1920 x 1080<br/>frames per second<br/>(fps): 30.0<br/>number of frames:<br/>811</p> |
| в                                                                                   |                                                                                                                                                                                   |

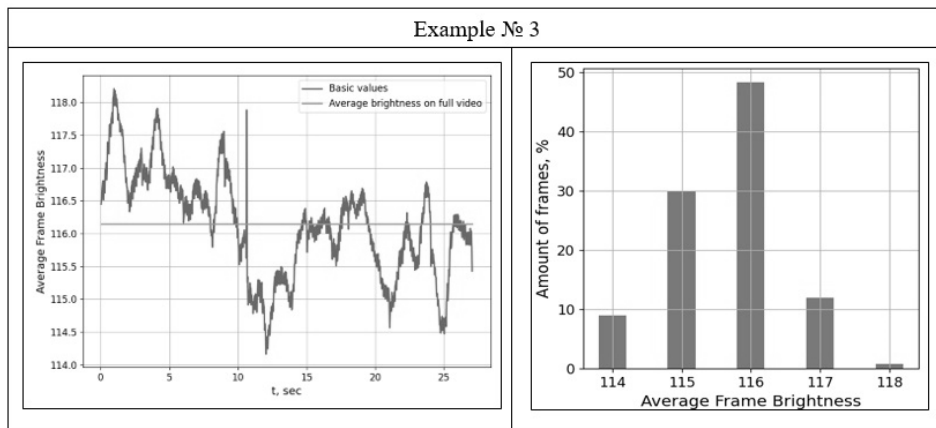
Рис. 3



a



б



в

Рис. 4

### Фільтрування функції яскравостей кадрів відеопотоку

При аналізі графіків зміни яскравості кадрів відеопотоку на рис. 4 нескладно помітити, що це графіки випадкових процесів, що містять низькочастотну та високочастотну компоненти. Низькочастотна компонента відображає умови освітленості сцени, що повільно змінюються, і несе в собі корисну інформацію. Так, у прикладі 2 є тенденція до зменшення загального рівня освітленості (протягом 50 с AFB знизилася від 100 до 85 градацій яскравості в діапазоні  $[0 \dots 255]$ ). У той



же час високочастотна компонента має шумоподібний характер. Це свідчить про швидку і спонтанну зміну яскравості кадрів відео. ВЧ-компоненту слід розглядати як перешкоду, для усунення якої необхідно використовувати процедуру НЧ-фільтрації.

З великого набору можливих методів цифрової фільтрації ми використовували один із найпростіших і найефективніших алгоритмів — усереднюючий фільтр зі ковзним вікном.

Такий алгоритм працює за принципом буфера, де зберігаються останні дані для усереднення. Це числа, що відповідають середній яскравості кадрів AFB, кількість яких визначена розмірами вікна фільтра та швидкістю зміни кадрів (fps). Так, при ширині вікна фільтрації 5 с і  $\text{fps} = 30$  кількість чисел AFB в буфері для усереднення дорівнює 150. На кожному кроці фільтрації процедура усереднення даних визначається наступним чином:

$$\text{AFB}_{\text{filtr}} = \frac{1}{W} \sum_{w=1}^W \text{AFB}_w, \quad (2)$$

де  $\text{AFB}_w$  — одновимірний масив чисел, що становлять послідовність значень середньої яскравості кадрів AFB не більше вікна  $W$  фільтра. На кожному новому кроці буфер зсувається, до нього додається нове значення  $\text{AFB}_{w+1}$  і забирається найстаріше —  $\text{AFB}_1$ , після чого процедура усереднення повторюється. Так відбувається ковзання вікна фільтра масиву значень AFB.

Результати фільтрації даних про середню яскравість кадрів відеопослідовностей наведено на рис. 5 для всіх трьох прикладів запису систем відеоспостереження. Для оцінки ефективності фільтрації вибирався різний розмір вікна  $T$  — 1, 5, 10 та 30 с. Для кожного з трьох прикладів відеозапису побудовано поєднані в одному вікні графіки зміни середньої яскравості кадрів AFB до та після фільтрації (див. рис. 5, а). Вочевидь, що з малої ширини вікна ( $T = 1\text{с}$ ) ефективність фільтрації вкрай невелика — шумова компонента частково зберігається, а НЧ-компонента модулюється середньочастотними змінами. За більшого постійного часу фільтра ( $T = 5\text{с}$ ) якість фільтрації помітно підвищується — шумова компонента повністю пригнічується, а вплив середньочастотної суттєво зменшується. Цей варіант фільтрації показано на рис. 5 у вигляді окремого графіка на тлі вихідної кривої розподілу AFB.

Не менш інформативними є й результати порівняння гістограм розподілу AFB до і після фільтрації з постійною часу  $T = 5\text{с}$  (гістограми ліворуч — для вихідного розподілу, а гістограми праворуч — для розподілу після фільтрації). З гістограм видно, що характер розподілу не змінився, але розкид параметра яскравості щодо середнього після фільтрації помітно зменшується.

Точність представлення даних усереднення яскравості, як до, так і після фільтрації, визначається помилками округлення і становить не більше половини елементарної градації шкали яскравостей. При використанні формату даних `uint8` помилка округлення дуже мала і становить не більше 0,4 %.

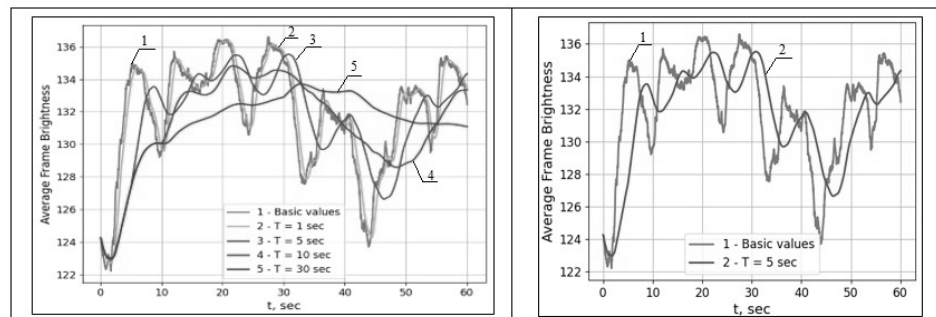
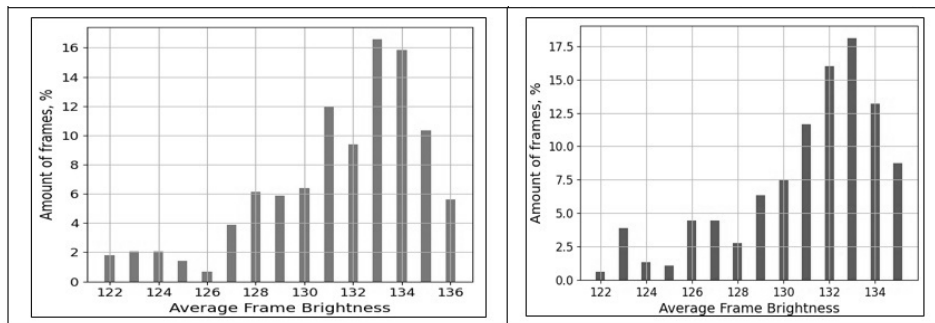
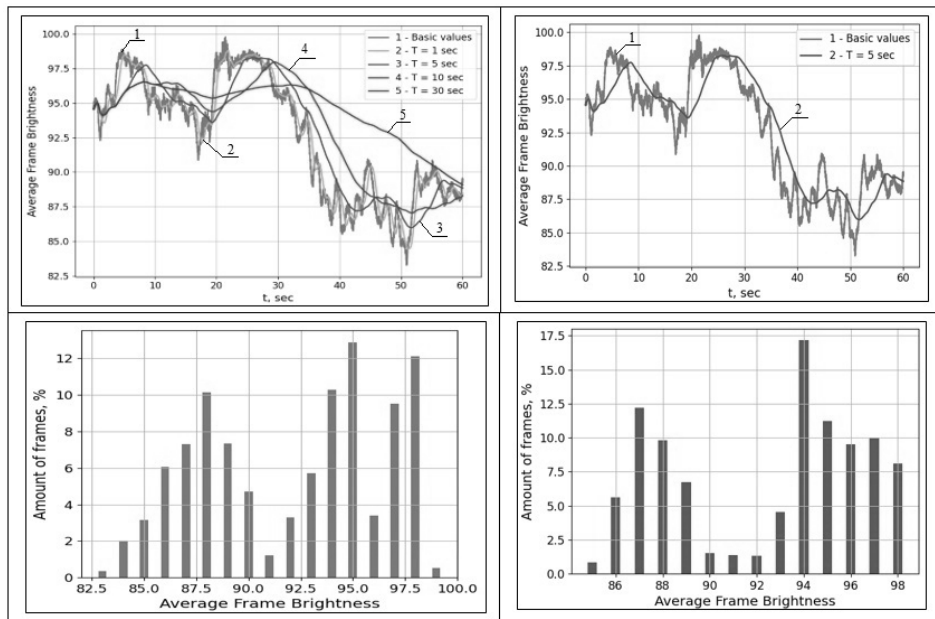


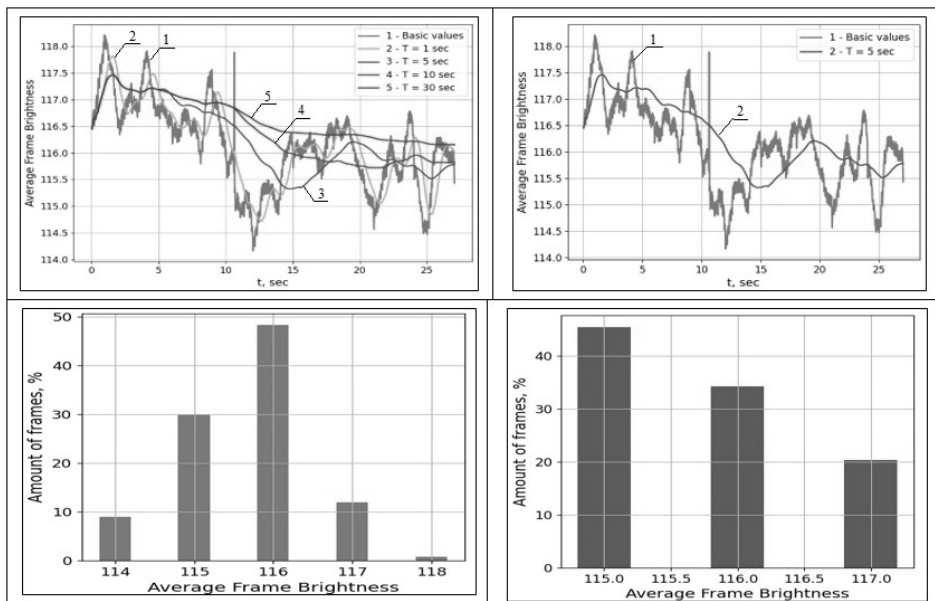
Рис. 5



a



б



в

### Стабілізація яскравості відеопотоку

Результати аналізу використані для стабілізації яскравості кадрів відеопотоку. Для цього автори запропонували такий алгоритм:

- середня яскравість фреймів після згладжування НЧ-фільтром вибирається як опорне значення;
- для кожного кадру відеопотоку визначається різниця значень середньої яскравості вихідного значення  $AFB_i$  та опорного  $AFB_{filtr\ i}$  з урахуванням знака;
- на базі отриманих прирощень  $\Delta AFB_i$  будується лінійний дискримінатор за правилом:
  - якщо виконується умова  $AFB_i - AFB_{filtr\ i} > 0$ , то  $AFB_{i\ correction} = AFB_i - \Delta AFB_i$ ;
  - якщо виконується умова  $AFB_i - AFB_{filtr\ i} < 0$ , то  $AFB_{i\ correction} = AFB_i - \Delta AFB_i$ ;
  - для всіх значень коригування вводяться обмеження, які не дозволяють пікселям стабілізованого кадру вийти за межі діапазону яскравостей  $[0 \dots 255]$ .

Програмна реалізація цього алгоритму проста, код процедури стабілізації яскравості кадрів відео показаний на рис. 6, а результати стабілізації яскравості відеопотоку — на рис. 7.

```
53 # change the image brightness
54 def brightness_change(image_hsv, value):
55     # Getting separate channels
56     h, s, v = cv2.split(image_hsv)
57     # Changing the brightness
58     if value >= 0:
59         lim = 255 - value
60         v[v > lim] = 255
61         v[v <= lim] += value
62     else:
63         v[v >= -value] -= -value
64         v[v < -value] = 0
65     # Getting an image with changed brightness
66     return cv2.merge((h, s, v))
```

Рис. 6

Для всіх трьох відеозаписів представлені характерні приклади вихідних та стабілізованих фреймів, гістограми розподілу яскравості пікселів цих фреймів до та після стабілізації, а також графіки зміни параметрів  $AFB_i$ , опорного значення  $AFB_{filtr\ i}$  та крива прирощень  $\Delta AFB_i$ .

У першому прикладі відеозапису неважко помітити, що в результаті стабілізації яскравості скоригований кадр дещо яскравіший за вихідний. При цьому гістограма цього кадру не змінила свою форму, а лише стала більш яскравою. Це дозволяє зробити дуже важливий висновок — процедура стабілізації яскравості лінійна. А отже, не порушується колірний баланс кадрів вихідного відеопотоку.

### Результати експериментальних досліджень

У заключній частині роботи приведені результати дослідження якості запропонованої попередньої обробки відео. Відомо, що у більшості практичних застосувань якість сприймається як міра близькості двох зображень: перетвореного

і вихідного. Для порівняння використовуються критерії візуального сприйняття і об'єктивні критерії, що базуються на аналізі пар пікселів двох зображень. Зазвичай як показник подібності використовують пікове відношення сигнал/шум PSNR (peak signal to noise ratio), що визначається формулою

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\text{MSE}}, \quad (3)$$

де середньоквадратична помилка MSE (mean square error) позначена виразом

$$\text{MSE} = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (I(m, n) - C(m, n))^2. \quad (4)$$

Цей показник використовують для аналізу якості перетворень у більшості програм. Однак автори вважають, що в даному випадку доцільніше використовувати лінійний захід відповідності, оскільки запропонована процедура стабілізації яскравості кадрів лінійна. Цей показник називається середньою абсолютною помилкою (mean absolute error) та обчислюється за формулою

$$\text{MAE} = \frac{1}{M \times N} \sum_{m=1}^M \sum_{n=1}^N |I(m, n) - C(m, n)|, \quad (5)$$

де  $I(m, n)$  та  $C(m, n)$  — вихідне та скориговане зображення кадру.

Результати аналізу ефективності стабілізації яскравості кадрів відеопослідовності показані на рис. 7 та 8 у вигляді гістограм розподілу середньої абсолютної помилки перетворення (MAE) для всіх кадрів відео. Шкала помилок відградуєвана в одиницях подання пікселів за яскравістю в діапазоні  $[0 \dots 255]$ .

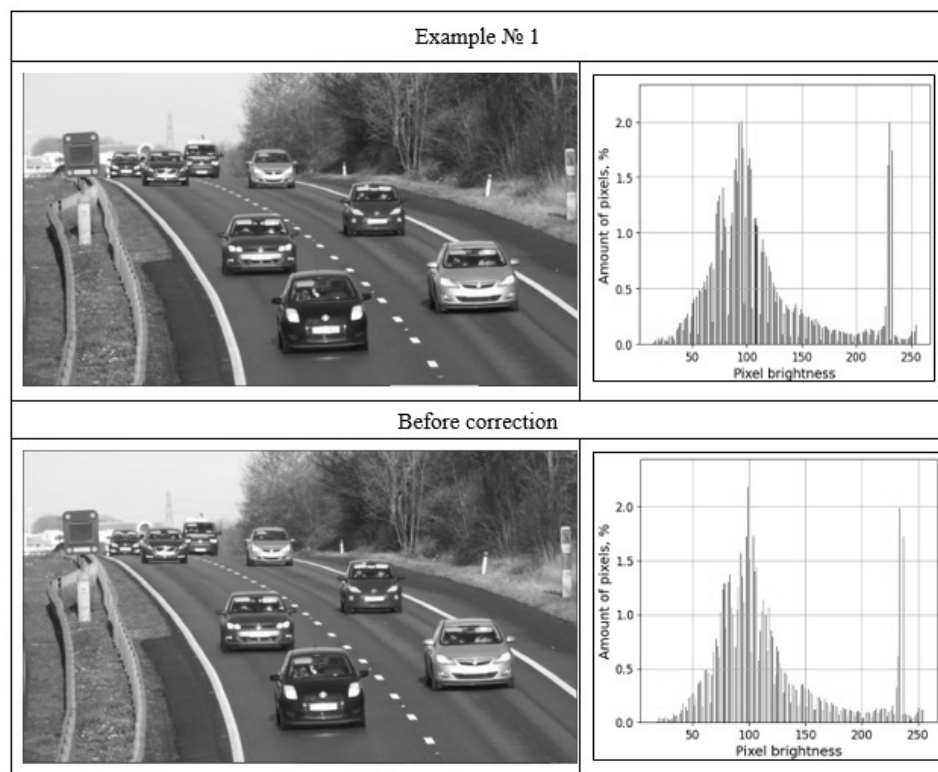
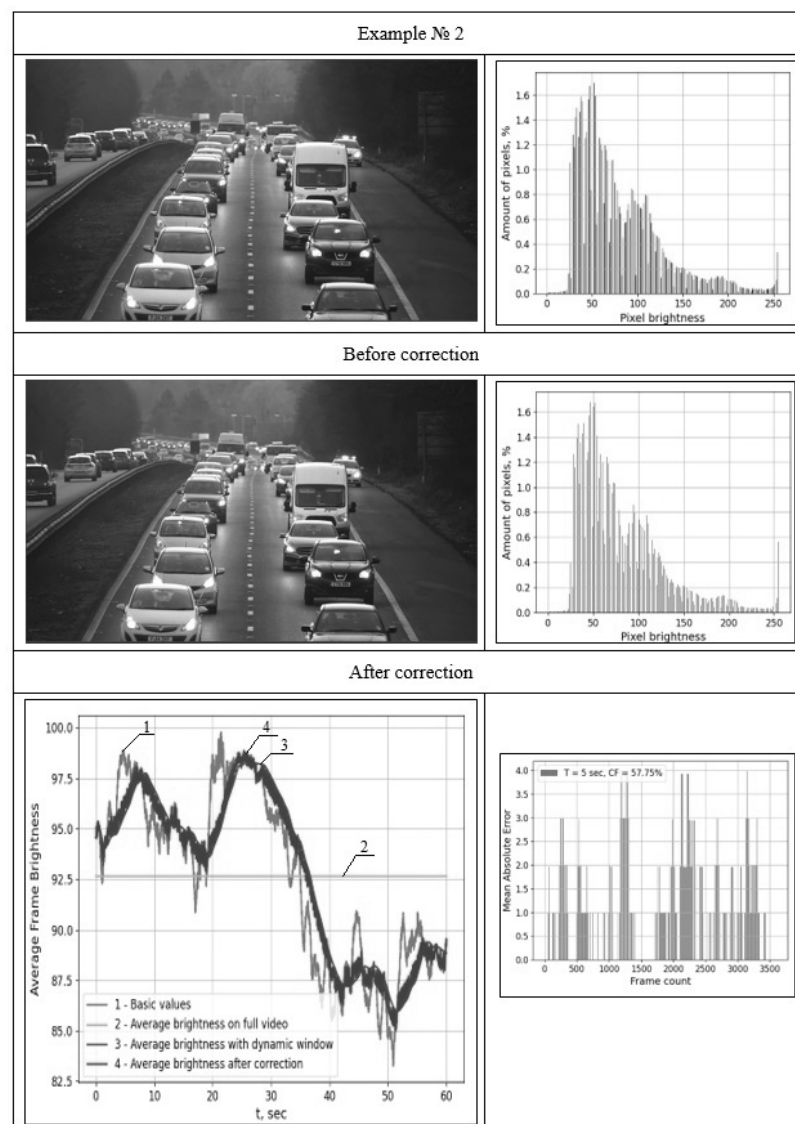
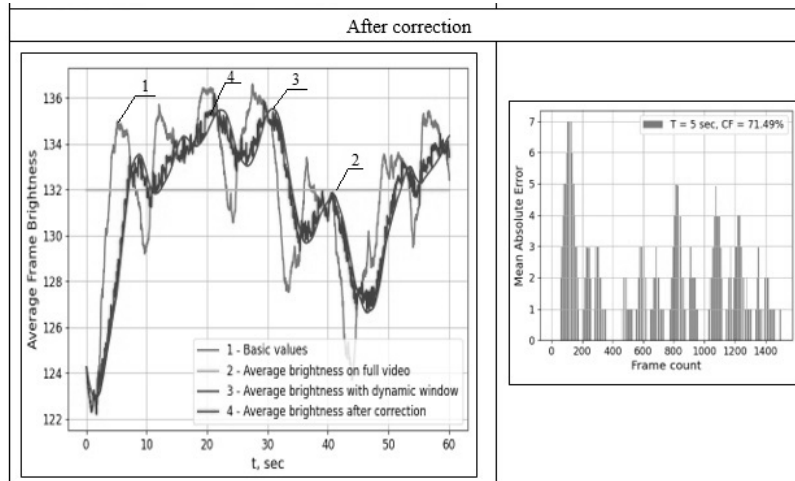
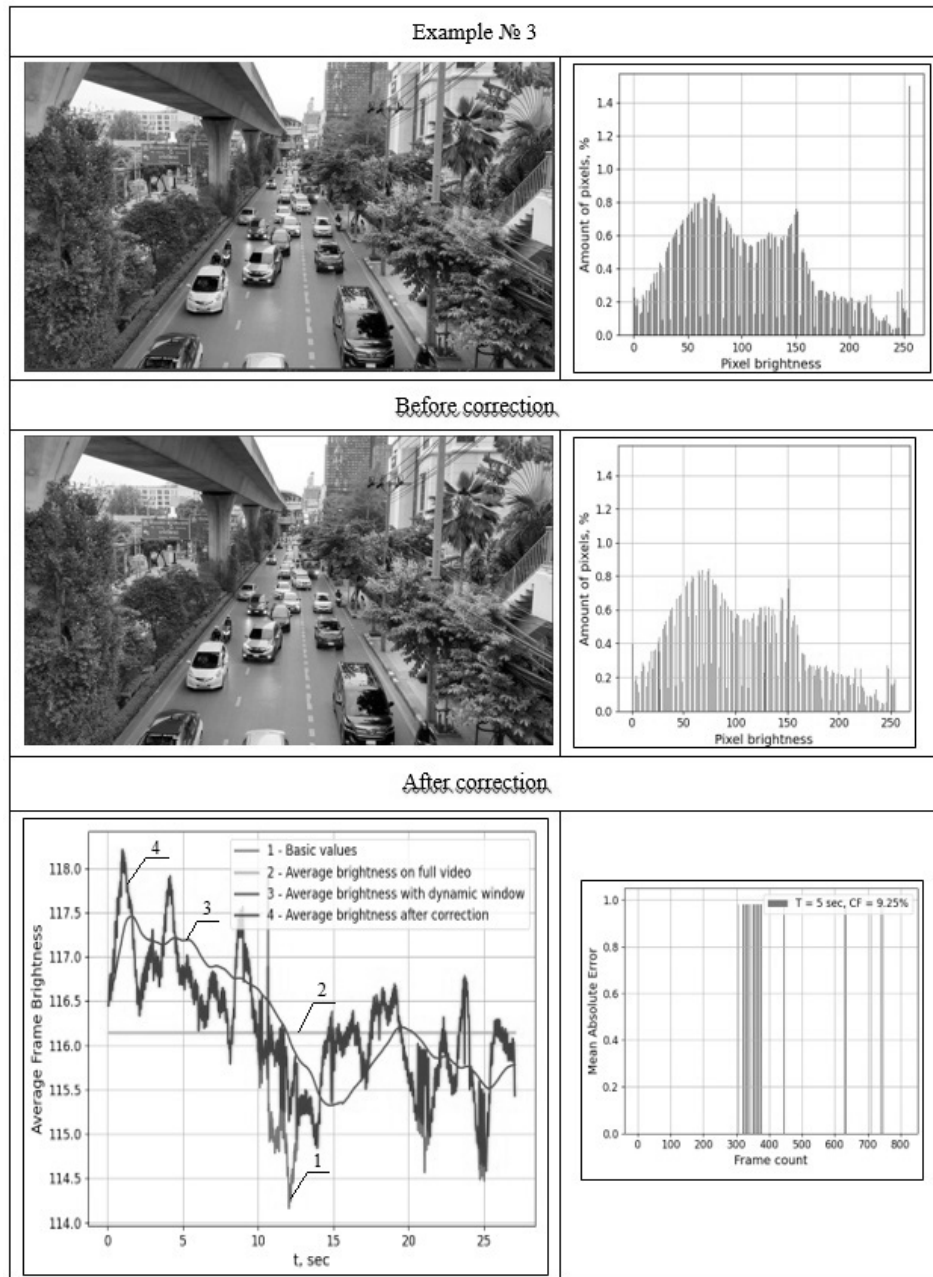


Рис. 7





6

Рис. 7

Крім цього, розраховувався показник фактора корекції CF як відношення кількості кадрів, що зазнали корекції, до загальної кількості кадрів у записі відео. Індекс CF відображається у відсотках.

На рис. 7 розподіл помилки MAE показано для трьох різних відеозаписів, тренд середньої яскравості яких AFB<sub>i</sub> відфільтрований НЧ фільтром з постійною  $T = 5$  с. На рис. 8 показано розподіл MAE тільки для одного відеозапису, але після фільтрації з різним постійним часом:  $T = 1, 5, 10$  або  $30$  с.

Аналіз отриманих даних показав, що:

- у всіх прикладах значення MAE порівняно невелике і не перевищує 8 одиниць за шкалою яскравості;
- у більшості прикладів у відеопослідовності можна спостерігати кілька інтервалів, в яких фрейми не піддавалися корекції по яскравості;
- значення показника CF у наведених прикладах коливається в діапазоні від 64 до 85 %, за винятком прикладу, в якому через малу ефективність фільтра ( $T = 1$  сек) воно становить 26 %.

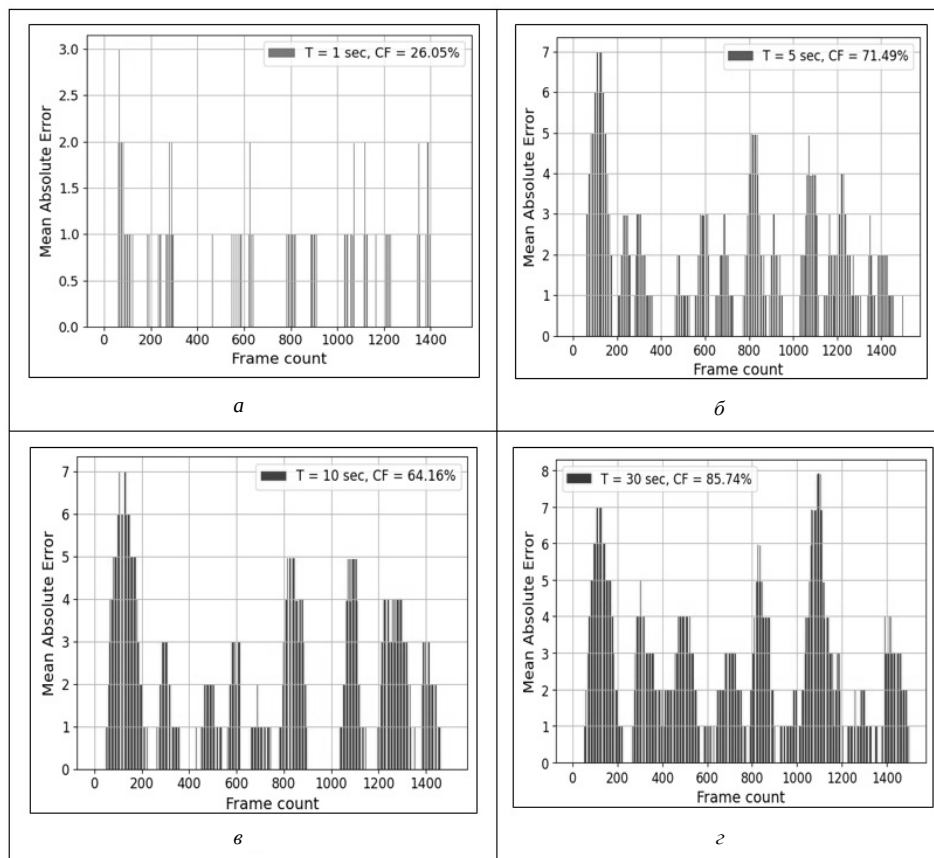


Рис. 8

Отже, для отримання високої якості стабілізації яскравості відеопотоку є можливість гнучко налаштовувати параметри попередньої обробки відеоданих залежно від характеру запису та мінливості умов освітлення сцени.

### Висновок

Попередня обробка відеоданих сприяє підвищенню якості систем технічного зору, що використовуються для вирішення широкого кола завдань розпізнавання образів з використанням технологій штучного інтелекту на базі нейронних мереж. Тому проведені дослідження актуальні та корисні. Рекомендації щодо аналізу та алгоритми стабілізації яскравості фреймів відеопотоку мають універсальний характер і можуть використовуватися при побудові нових та модернізації існуючих систем. Робота має явно виражену практичну спрямованість. Пропоновані алгоритми реалізовані у вигляді програмних модулів мовою програмування Python та бібліотеки OpenCV. Експериментально досліджено показники якості роботи різних методів стабілізації яскравості відеопотоку.

*V. Bilozerskyi, K. Dergachov, L. Krasnov*

## ANALYSIS AND PRE-PROCESSING OF VIDEO DATA TO IMPROVE THE QUALITY OF VISION SYSTEMS

**Vladislav Bilozerskyi**

National Aerospace University «Kharkiv Aviation Institute»,

*b11lays123@gmail.com*

**Konstantin Dergachov**

National Aerospace University «Kharkiv Aviation Institute»,

*k.dergacho@khai.edu*

**Leonid Krasnov**

National Aerospace University «Kharkiv Aviation Institute»,

*leonid.krasnov.1947@gmail.com*

The subject of study. The article proposes a new method for analyzing and evaluating the variability of initial data in video surveillance systems under the influence of negative external factors. Based on the results of the analysis, algorithms for stabilizing the parameters of the received video stream are proposed and an assessment of the quality of video data as a result of their preliminary processing is given. The aim of the work is to analyze the variability of the main parameters of input data in video surveillance systems under the influence of negative external influences, to create effective algorithms for stabilizing the parameters of the received video stream, to assess the quality of video data as a result of their preliminary processing, and to check the reliability of the proposed method on real video surveillance records. The task was to determine the most universal, stable and informative quality indicator when analyzing video recordings or video in real time, which allows to give an objective assessment of the quality of video data under various shooting conditions and the nature of the illumination of the scene, as well as to develop and programmatically implement algorithms for stabilizing video parameters. Statistical analysis and processing of the parameters of the analyzed video sequence as a random spatio-temporal process, algorithms for processing video data by digital filtering and adaptive stabilization of video stream brightness parameters. It has been shown that the most stable indicator of video stream quality is the average frame brightness AFB (Average Frame Brightness). A universal algorithm for spatio-temporal processing of video data is proposed, which forms a sequence of values of the average brightness of AFB frames from the original video stream, as well as algorithms for filtering and stabilizing the brightness of the video stream. Conclusions. Scientific novelty of the obtained results — as a new method of objective analysis and evaluation of the parameters of the initial video surveillance data, algorithms for filtering and stabilizing the brightness of the video stream are proposed. The quality of the proposed algorithms has been verified on real video surveillance results. The proposed algorithms are implemented in the Python software environment using the functions of the OpenCV library.

**Keywords:** Analysis of the nature of changes in the brightness of video data; average frame brightness in the video stream; digital filtering of the video stream brightness trend, video data brightness stabilization algorithms.



## ПОСИЛАННЯ

1. Lecun Y. et al. Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*. 1998. Vol. 86, N 11, P. 2278–2324. DOI: 10.1109/5.726791.
2. Інформація стосовно бібліотеки ImageNet. Режим доступу: <https://ru.wikipedia.org/wiki/ImageNet>.
3. Krizhevsky A., Sutskever I. and Hinton G.E. ImageNet classification with deep convolutional neural networks. *In Advances in neural information processing systems*. 2012. № 25. P. 1097–1105.
4. Krizhevsky A. et al. ImageNet Classification with Deep Convolutional Neural Networks. *Communications of the ACM*. 2017. Vol. 60, N 6, P. 84–90. DOI:10.1145/3065386.
5. Програмне забезпечення Caffe . <http://caffe.berkeleyvision.org/>.
6. Бібліотека TensorFlow. <https://www.tensorflow.org/>.
7. Наукова обчислювальна платформа Torch . <http://torch.ch/>.
8. Фреймворк нейронної мережі Darknet . <https://pjreddie.com/darknet/>.
9. Бібліотека Keras. <https://keras.io/>.
10. Бібліотека OpenCV DNN . <https://github.com/opencv/opencv/tree/master/modules/dnn>.
11. Timoshin Y., Yuzhda Y. Analysis of features of application of neural networks for intellectual processing of video flows of technical vision systems. *Mizhvidomchyy naukovotekhnichnyy zbirnyk «Adaptyvni systemy avtomatychnoho upravlinnya» Interdepartmental scientific and technical collection Adaptive automatic control systems*. 2021. N 2 (39). <https://ascac.kpi.ua/article>.
12. Hajimowlana S.H., Jullien G.A., Muscedere R. and Roberts J.W. Efficient pre-processing algorithms for an FPGA based in-camera video-stream processing system for industry inspection, CCECE '97. *Canadian Conference on Electrical and Computer Engineering. Engineering Innovation: Voyage of Discovery. Conference Proceedings, St. John's, NL, Canada, 1997*, Vol. 2, P. 835–838. DOI: 10.1109/CCECE.1997.608373.
13. Bhat M., Thiesse J. -M. and Callet P. L. HVS based perceptual pre-processing for video coding, 2019 27th European Signal Processing Conference (EUSIPCO), A Coruna, Spain, 2019, P. 1–5. DOI: 10.23919/EUSIPCO.2019.8903172.
14. Huang S., Luo Z., Xu J., Zhou W. and Chen Z. Perceptual Evaluation of Pre-processing for Video Transcoding. *International Conference on Visual Communications and Image Processing (VCIP)*. Munich, Germany. 2021. P. 1–5. DOI: 10.1109/VCIP53242.2021.9675438.
15. Skadins A., Ivanovs M., Rava R. and Nesenbergs K. Edge pre-processing of traffic surveillance video for bandwidth and privacy optimization in smart cities. *17th Biennial Baltic Electronics Conference (BEC)*. Tallinn, Estonia. 2020. P. 1–6. DOI: 10.1109/BEC49624.2020.9276799.
16. Zaharia C., Bigioi P. and Corcoran P.M. Hybrid video-frame pre-processing architecture for HD-video. *IEEE International Conference on Consumer Electronics (ICCE)*. Las Vegas, NV, USA. 2011. P. 89–90. DOI: 10.1109/ICCE.2011.5722927.
17. Dergachov K. et al. Data pre-processing to increase the quality of optical text recognition systems. *Radioelektronni i komp'uterni sistemi — Radioelectronic and computer systems*. 2021. N 4(100). P. 183–198. DOI: 10.32620/reks.2021.4.15.

Отримано 16.03.2023