

УДК 004.932

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

ОЦІНКА ЕФЕКТИВНОСТІ НОВОГО МЕТОДУ СТАБІЛІЗАЦІЇ ЯСКРАВОСТІ ВІДЕОПОТОКУ В РЕАЛЬНОМУ МАСШТАБІ ЧАСУ

Білозерський Владислав Олександрович

Національний аерокосмічний університет ім. М.С. Жуковського «Харківський авіаційний інститут»,

orcid: 0000-0002-5503-3163

v.o.bilozerskyi@khai.edu

Білоус Олександр Олександрович

Національний аерокосмічний університет ім. М.С. Жуковського «Харківський авіаційний інститут»,

orcid: 0009-0004-2362-7460

aabelous@ukr.net

Дергачов Костянтин Юрійович

Національний аерокосмічний університет ім. М.С. Жуковського «Харківський авіаційний інститут»,

orcid: 0000-0002-6939-3100

k.dergachov@khai.edu

Краснов Леонід Олександрович

Національний аерокосмічний університет ім. М.С. Жуковського «Харківський авіаційний інститут»,

orcid: 0000-0003-2607-8423

leonid.krasnov.1947@gmail.com

Проведено порівняльний аналіз класичних нелінійних методів стабілізації яскравості та підвищення контрастності відеоданих, а також уперше запропоновано оригінальний метод лінійної стабілізації яскравості відеопотоку і працюючий в реальному масштабі часу алгоритм, що його реалізує. Дано об'єктивну оцінку якості стабілізації яскравості відеоданих за результатами попереднього оброблення порівняно з нелінійними методами. Метою роботи є створення методології аналізу мінливості параметрів відеоданих під впливом негативних чинників і розробка ефективних алгоритмів стабілізації параметрів відеопотоку; ретельно перевіряється достовірність нового методу на реальних відеозаписах, здійснених у різних умовах. Визначено найбільш універсальний, стійкий до зовнішніх впливів та інформативний показник, необхідний для об'єктивного оцінювання якості відеоданих за різних умов знімання та особливостей освітлення сцени; розроблено та програмно реалізовано лінійний алгоритм, що дає змогу ефективно стабілізувати яскравість відео в реальному масштабі часу. Розробка базується на основі сучасних засобів програмування. Для дослідження проведено статистичний аналіз та попередню обробку параметрів відеопотоку як випадкового просторово-часового процесу, алгоритми обробки відеоданих в реальному масштабі часу шляхом цифрової фільтрації та адаптивної стабілізації параметрів відеопотоку. Запропоновано та експериментально доведено, що оптимальним показником якості відеопотоку є середня кадрова яскравість (Average Frame Brightness — AFB). Описано лінійний алгоритм просторово-часової обробки відеоданих, який генерує послідовність значень AFB з вихідного відеопотоку, а також цифрові алгоритми фільтрації та стабілізації яскравості відеопотоку та досліджено ефективність їх застосування. Наукова

новизна отриманих результатів полягає в новому методі аналізу та оцінки параметрів даних відеоспостереження, алгоритмах фільтрації та стабілізації яскравості відеопотоку в реальному масштабі часу. Продуктивність запропонованих алгоритмів перевірено на реальних даних. Алгоритми реалізовані в програмному середовищі Python з використанням функцій бібліотеки OpenCV.

Ключові слова: відеопотік, середня яскравість кадру, тренд яскравості відео, цифрова фільтрація трендів, алгоритми стабілізації яскравості в реальному масштабі часу.

Вступ

Нині спостерігається значний прогрес у розв'язанні різних задач розпізнавання образів методами комп'ютерного зору. Дуже важливу роль при цьому відіграють методи й алгоритми штучного інтелекту на базі нейронних мереж з глибоким навчанням. Їх ефективність висока — ймовірність безпомилкових рішень становить понад 99 %. Однак існує низка негативних експлуатаційних чинників, що істотно обмежують можливості їх використання. Іноді це призводить навіть до порушення працездатності систем обробки даних. До основних чинників, що обмежують якість процедур виявлення і розпізнавання образів, відносять погані та мінливі умови освітлення сцени, а також неконтрольовані переміщення і тремтіння камери під час зйомки відео. Стабілізація яскравості та контрастності відеопотоку і компенсація впливу неконтрольованого положення камери — основна суть попередньої обробки відеоданих програмними засобами. Слід зазначити наполегливі спроби радикального розв'язання проблеми попереднього опрацювання відео з боку багатьох дослідників, їхні результати розглядаються далі. Але, на жаль, остаточного вирішення цих завдань на сьогоднішній день немає. Тому подальші дослідження актуальні й, безумовно, корисні.

Мета роботи — досягнення стабілізації яскравості відеопотоку з максимальною якістю та без відчутного погіршення візуальної складової при його перегляді. Для досягнення цієї мети планується розробка нового методу стабілізації та відповідного алгоритму його реалізації. Окремо увагу приділено забезпеченню можливості обробки відеопотоку в реальному масштабі часу. Для досягнення даної мети ставилися такі завдання:

- провести порівняльний аналіз відомих робіт щодо поліпшення контрастності та стабілізації яскравості даних у системах відеоспостереження;
- розробити універсальний, стійкий та інформативний показник якості, що дає змогу об'єктивно оцінити стан відеоданих за різних умов зйомки та характеру освітленості сцени;
- створити ефективний лінійний алгоритм стабілізації показника яскравості прийнятого відеопотоку за результатами аналізу;
- дослідити можливість роботи алгоритму з відеопотоками високої роздільної здатності у форматі Full HD і Ultra HD у режимі реального часу;
- оцінити властивості відеоданих у результаті їх попередньої обробки за загальноприйнятими статистичними критеріями якості;
- написати мовою Python з використанням функцій бібліотеки OpenCV програмні коди для роботи нового алгоритму;
- перевірити і підтвердити на реальних записах відеоспостережень достовірність отриманих результатів.

Як зазначалося, попереднє опрацювання вихідного відео для компенсації негативних зовнішніх впливів сприяє комфортнішому візуальному сприйняттю відео і повнішому й ефективнішому використанню методів і алгоритмів комп'ютерного зору.

Перші ранні спроби підвищення контрасту цифрових зображень і відео засновані на методі еквалізації (вирівнювання) гістограм розподілу яскравостей пікселів [1, 2]. Це простий і досить ефективний метод поліпшення якості зображення. Однак йому

притаманна й низка суттєвих недоліків. Головний з них — значна зміна середньої яскравості зображення. Це зазвичай тягне за собою надмірне посилення шумів і появу інших дратівливих артефактів. Глобальна зміна яскравості призводить до недонасичення, або перенасичення окремих фрагментів зображення.

Проблема стабілізації яскравості зображення для поліпшення якості зорового сприйняття значно загострилася з розвитком засобів комп'ютерного зору, що вимагають більш якісної стабілізації інформаційних параметрів. У зв'язку з цим опублікована низка робіт [3–5], у яких ставилося завдання — поліпшити метод еквалізації гістограм. Розглядалися й аналізувалися модифіковані методи еквалізації гістограм, засновані на бігістограмній або мультигістограмній еквалізації, і еквалізації гістограм з обрізанням. Пізніше з'явилися й інші публікації з різними пропозиціями щодо поліпшення процедури підвищення контрастності [6–9]. На найбільшу увагу заслуговують публікації [10, 11], які мають оглядовий характер цієї проблеми. Однак усі спроби збільшення контрастності зі збереженням характеристик яскравості кадрів не мали успіху, оскільки процедура еквалізації сама по собі нелінійна, а це неминуче порушує колірний баланс кадрів відео та неконтрольованої зміни їхньої середньої яскравості.

У роботах [12–14] запропоновано альтернативний метод розв'язання задачі підвищення якості даних відеоспостереження — метод лінійної стабілізації яскравості кадрів на основі аналізу показника середньої яскравості (Average Frame Brightness — AFB), який істотно перевершує за характеристиками якості класичні алгоритми поліпшення яскравості та контрастності. Оцінювання якості перетворень проводили з використанням сучасних статистичних метрик, що враховують властивості зорової системи людини (Human Vision System — HVS) [15, 16].

Аналіз показав необхідність продовження досліджень, які, безсумнівно, сприятимуть підвищенню ефективності систем комп'ютерного зору.

Реалізація алгоритмів підвищення контрастності та стабілізації яскравості відео

Для об'єктивного порівняння показників якості запропонованого лінійного алгоритму стабілізації яскравості відео та класичних алгоритмів підвищення його контрастності виконано їхню програмну реалізацію. На рис. 1, а, б наведено два варіанти класичного алгоритму еквалізації кадрів відеопотоку: з використанням колірного простору HSV і простору YIQ.

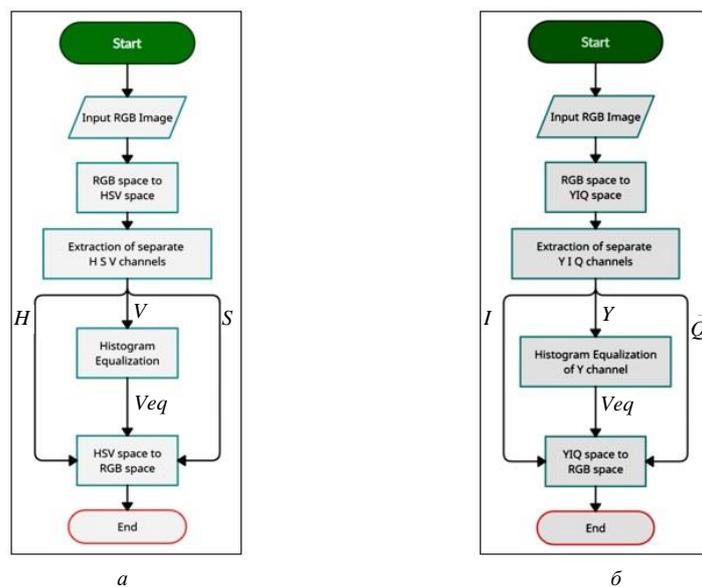


Рис. 1

Структура цих алгоритмів гранично проста, особливість їх використання зумовлена нелінійністю процедури еквалізації, виконання якої в ортогональному колірному просторі RGB істотно порушує колірний баланс. Тому потрібен перехід в інший колірний простір з незалежною компонентою яскравості та двома каналами колірного балансу зображення. Такими просторами є HSV і YIQ, який з них кращий, покажемо далі. Після еквалізації каналу яскравості здійснюється зворотний перехід у RGB.

Програмний код одного з таких алгоритмів показано на рис. 2. Тут і далі всі приклади реалізації алгоритмів і розрахунки показників якості наведено мовою програмування Python з використанням ресурсів бібліотеки OpenCV. Зазначимо, що в програмі для оцінки можливості роботи запропонованого алгоритму в реальному масштабі часу розраховується час виконання необхідних процедур.

```

8  def equalize_v_channel(image):
9      #Start time counting
10     start_time = time.time()
11
12     hsv_image = cv2.cvtColor(image, cv2.COLOR_BGR2HSV)
13
14     # Separate the H, S, and V channels
15     H, S, V = cv2.split(hsv_image)
16
17     # Equalize the V channel
18     equalized_V = cv2.equalizeHist(V)
19
20     # Merge the equalized V channel with the original H and S channels
21     equalized_hsv_image = cv2.merge((H, S, equalized_V))
22
23     # Convert the equalized HSV image back to BGR color space
24     equalized_image = cv2.cvtColor(equalized_hsv_image, cv2.COLOR_HSV2BGR)
25
26     #Counting time spent
27     end_time = time.time()
28     adjustment_time = end_time - start_time

```

Рис. 2

Запропонований авторами лінійний алгоритм стабілізації яскравості відеопотоку детально описано в [12–14], його структуру показано на рис. 3. Видно, що структуру класичного алгоритму доповнено такими блоками: зміна розмірів кадру (Frames Resizing), обчислення середньої яскравості кадру AFB (Average brightness calculation), НЧ-фільтрації (Low-pass Filtration), лінійного дискримінатора рівня середньої яскравості кадру (Linear Discriminator for Brightness Level), алгоритму стабілізації яскравості кадру (Brightness Stabilization Algorithm).

Блок зміни розмірів кадру забезпечує роботу алгоритму стабілізації в реальному масштабі часу під час використання відео високої роздільної здатності, наприклад, у форматі Full HD. Зменшення розмірів кадру помітно зменшує навантаження на процесор комп'ютера при обчисленні показника AFB у блоці Average Brightness Calculation за формулою

$$AFB = \frac{1}{MN} \sum_{n=1}^N \sum_{m=1}^M V(n, m), \quad (1)$$

де $V(n, m)$ — двовимірний масив чисел, що визначають яскравість пікселів зображення кадру розміром $M \times N$.

На рис. 4 показано програмний код функції обчислення AFB-кадру в HSV-форматі, а на рис. 5 — програмний код лінійного алгоритму корекції яскравості відео.

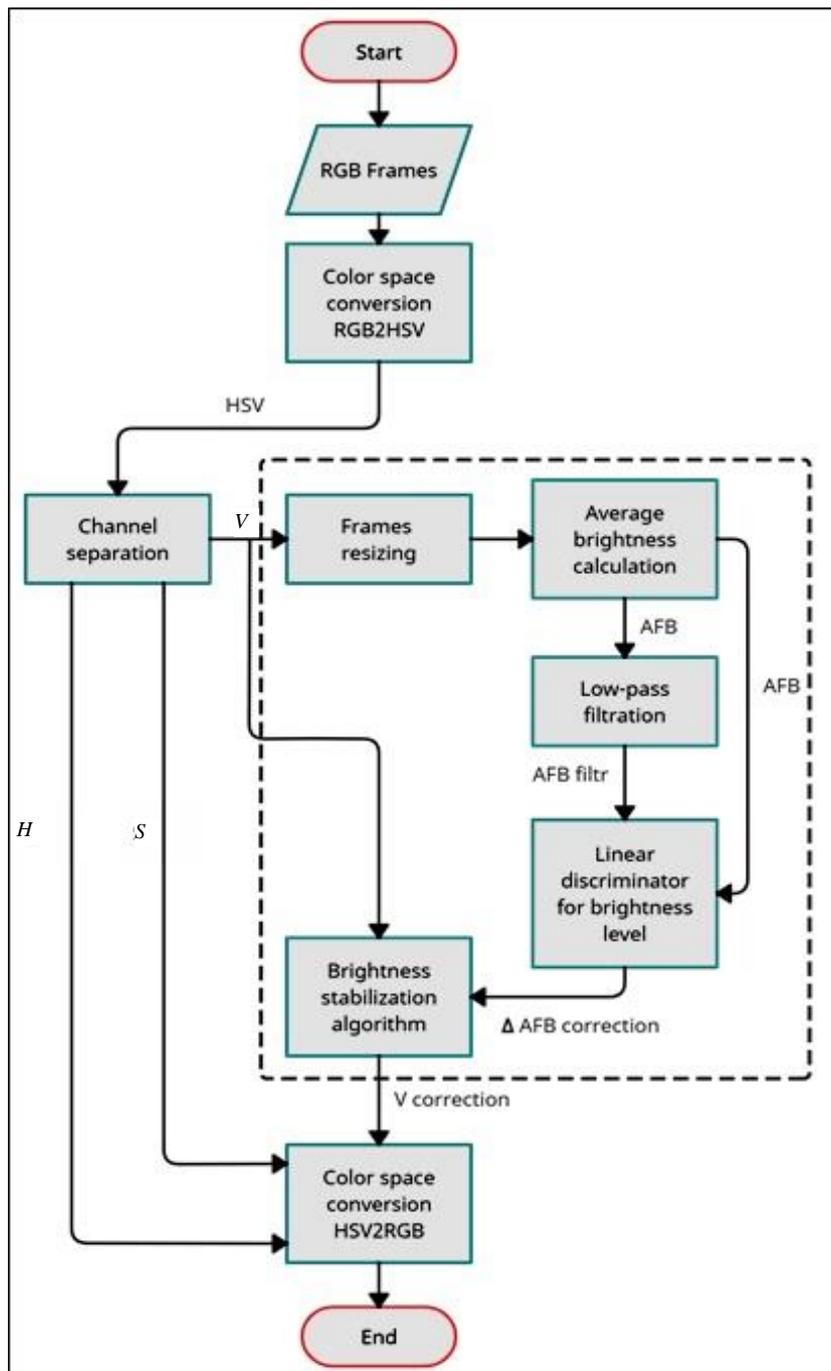


Рис. 3

Під час аналізу зміни яскравості кадрів відеопотоку добре видно, що це випадковий процес, який містить низькочастотну та високочастотну компоненти. Швидка і спонтанна зміна яскравості кадрів відео характеризує ВЧ-компоненту як перешкоду, для усунення якої необхідно використовувати процедуру НЧ-фільтрації. Блок НЧ-фільтрації проводить обчислення за формулою

$$AFB_{filtr} = \frac{1}{W} \sum_{w=1}^W AFB_w, \quad (2)$$

де AFB_w — одновимірний масив чисел, що представляють послідовність значень середньої яскравості кадрів AFB (див. формулу (1)) у межах вікна W-фільтра. На кожному новому кроці фільтрації буфер зсувається, до нього додають нове значення AFB_{w+1} і прибирають найстаріше AFB_1 , після чого процедуру усереднення повторюють. Так вікно фільтра ковзає масивом значень AFB.

```

35     def get_image_for_brightness_stab(self, frame):
36         # Working with individual frames
37         # Moving to the HSV space and recognizing medium brightness
38         original_hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
39         # Reducing the current frame for further processing
40         resized_hsv = resize(original_hsv, self.size_multiplier)
41
42         b_avg = brightness_hsv(resized_hsv)
43         # Updating the dynamic window
44         self.dynamic_window.append(b_avg)
45         dynamic_window_size = len(self.dynamic_window)
46
47         if dynamic_window_size > self.T_frame:
48             self.dynamic_window.pop(0)
49
50         base_value = np.mean(self.dynamic_window)
51         # Perform brightness averaging
52         diff = int(base_value - b_avg)
53         changed_image = brightness_change(original_hsv, diff)
54         changed_image_rgb = cv2.cvtColor(changed_image, cv2.COLOR_HSV2RGB)
55         return changed_image_rgb
56

```

Рис. 4

```

26 def brightness_change(image_hsv, value):
27     """Linear correction of the Value channel of the source image in HSV format
28     Input data: image_hsv - source image in HSV format;
29                value - value for brightness correction (0-255)
30     Output: image in HSV format with corrected Value channel"""
31     # Selecting separate image channels
32     h, s, v = cv2.split(image_hsv)
33     # Brightness correction based on extreme values (0, 255)
34     if value >= 0:
35         lim = 255 - value
36         v[v > lim] = 255
37         v[v <= lim] += value
38     else:
39         v[v >= -value] -= -value
40         v[v < -value] = 0
41     # Reverse image formation taking into account the corrected Channel Value
42     return cv2.merge((h, s, v))

```

Рис. 5

Далі в блоці лінійного дискримінатора рівня середньої яскравості кадру середня яскравість кадрів після згладжування НЧ-фільтром обирається як опорне значення, і для кожного кадру відеопотоку визначається різниця значень середньої яскравості вихідного значення AFB_i та опорного значення AFB_{filtr} (2) з урахуванням знака.

Блок алгоритму стабілізації яскравості кадру працює за логікою, що визначається формулою

$$\Delta AFB_i = AFB_i - AFB_{filtr};$$

$$\text{if } \Delta AFB_i \geq 0 \text{ then } AFB_{i \text{ correction}} = AFB_i - \Delta AFB_i;$$

$$\text{if } \Delta AFB_i < 0 \text{ then } AFB_{i \text{ correction}} = AFB_i + \Delta AFB_i.$$

Для всіх значень корекції вводяться обмеження, які не дають змоги пікселям стабілізованого кадру вийти за межі діапазону яскравостей $[0, \dots, 255]$. Це оберігає від появи недонасиченості або перенасиченості окремих фрагментів кадру.

Оцінка ефективності алгоритмів поліпшення якості відео

Для об'єктивного оцінювання якості різних алгоритмів необхідно провести їхнє тестування на наборі відеозаписів з різними властивостями: різною роздільною здатністю кадрів і різною частотою їхнього проходження (fps). Сформовано набір відеофайлів для тестування, їхні характерні кадри показано на рис. 6.

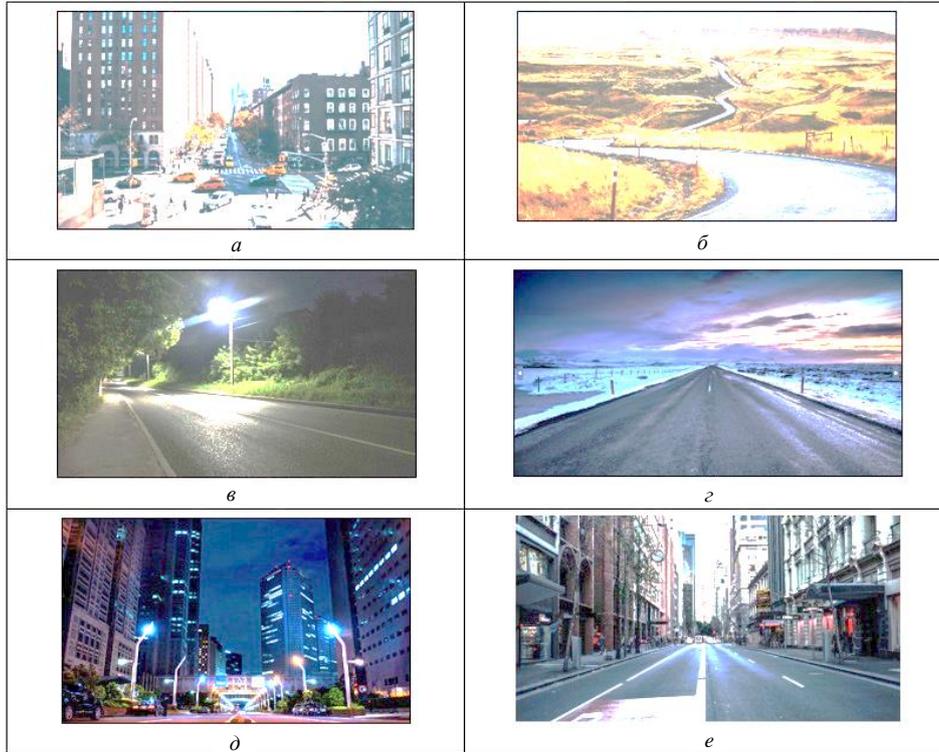


Рис. 6

Для повноти аналізу розглянемо як візуальні, так і кількісні маркери. До візуальних показників слід віднести зображення кадрів відео до і після підвищення контрастності та стабілізації яскравості, а також гістограми розподілу яскравостей кадрів відео до і після цих перетворень.

Для опрацювання цифрових зображень прийнято використовувати кілька популярних статистичних показників для оцінювання якості перетворень. У даному випадку доцільно використовувати такий набір показників.

1. AMBE (Absolute Mean Brightness Error). Для оцінки ефективності збереження вихідної яскравості кадру прийнято оцінювати абсолютну помилку середньої яскравості AMBE. Вона визначається як абсолютна різниця між середнім значенням яскравості вихідного і перетвореного кадрів і є важливим маркером якості перетворення [10, 11]:

$$AMBE = \left| AFB_{I_{output}} - AFB_{I_{input}} \right|, \quad (3)$$

де $AFB_{I_{input}}$ і $AFB_{I_{output}}$ — середня яскравість вихідного і перетвореного кадрів відповідно, AFB — середнє статистичне значення яскравості кадру, і обчислюється за формулою

$$AFB = \frac{1}{MN} \sum_{n=1}^N \sum_{m=1}^M V(n, m).$$

Зменшення значення AMBE вказує на краще збереження яскравості зображення. Рівняння (3) показує, що AMBE призначений для виявлення одного з неприємних спотворень — надмірної зміни яскравості.

2. PSNR (Peak Signal to Noise Ratio). Зазвичай як показник подібності використовують пікове відношення сигнал/шум, що визначається формулою

$$PSNR = 10 \log_{10} \frac{255^2}{MSE},$$

де середньоквадратична помилка MSE (Mean Square Error) визначається виразом

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

де $I(m, n)$ і $C(m, n)$ — вихідне і скореговане зображення кадру. Цей показник використовують для аналізу якості перетворень у більшості додатків.

3. NCC (Normalized-Cross-Correlation). Нормалізована крос-кореляція NCC показує, наскільки сильно вихідний та перетворений кадри корелюють між собою. Значення NCC лежить у діапазоні від 0 до 1. Якщо значення NCC дорівнюють 1, це означає повну ідентичність кадрів. NCC представляється формулою

$$NCC = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (S(i, j) \times C(i, j)) / \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (S(i, j))^2.$$

4. SSIM (Structural Similarity Index Measurement) — вимірювання індексу структурної подібності. Це метрика оцінки якості зображення, яка порівнює два зображення (вихідне та перетворене) для отримання подібності між ними. Пропонується як поліпшення $PSNR$. $SSIM$ набуває форми

$$SSIM(C, S) = \frac{(2\mu_C \mu_S + c_1) \times (2\sigma_{CS} + c_2)}{(\mu_C^2 + \mu_S^2 + c_1) \times (\sigma_C^2 + \sigma_S^2 + c_2)},$$

де μ_C — середнє значення C перетвореного кадру, μ_S — середнє значення S -пікселів вихідного кадру, σ_C^2 — дисперсія C , σ_S^2 — дисперсія S , σ_{CS} — коваріація C і S , $c_1 = (K_1 L)^2$ і $c_2 = (K_2 L)^2$ — дві змінні, що стабілізують розподіл зі слабким знаменником, L — динамічний діапазон значень пікселів, $K_1 = 0,01$ і $K_2 = 0,03$ за замовчуванням.

Наведемо кілька прикладів оцінки якості за візуальними маркерами.

Приклад 1. Використовується класичний алгоритм підвищення контрастності відеоданих із переходом для еквалізації з колірною простору RGB у простір HSV. На рис. 7, а, б показано зображення кадру до і після еквалізації, а на рис. 7, в в одному вікні наведено гістограми розподілу яскравостей кадру до і після перетворення. У цьому ж вікні у вигляді різнокольорових стовпчиків наведено розрахункові значення середньої яскравості кадру до і після еквалізації. На рис. 7, б добре помітні структурні зміни кадру після еквалізації — на тлі світлого неба зображення хмари явно перенасичені. Гістограма після перетворення набуває більш коливального характеру, а середня яскравість кадру зростає приблизно на 10 одиниць шкали яскравості.

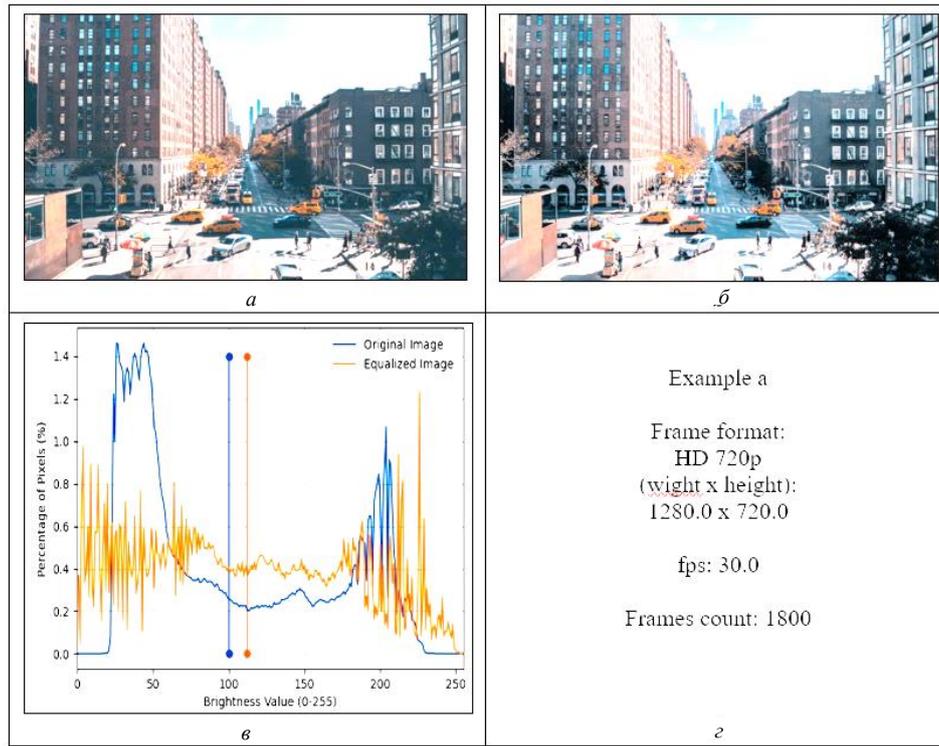


Рис. 7

У разі використання для підвищення контрастності цього відеопотоку алгоритму з переходом з колірному простору RGB у простір YIQ результати мають значно гірший вигляд (рис. 8). Візуальне сприйняття погіршується, значно помітнішими стають структурні зміни, гістограма розподілу яскравостей має хаотичний характер, а зміни середньої яскравості кадру досягають 25 одиниць шкали.

Приклад 2. Розглянемо аналогічний приклад роботи класичного алгоритму підвищення контрастності відео з переходом у простір HSV. Однак як тестове відео будемо використовувати запис з низьким рівнем освітленості. Результати обробки показано з переходом у простір HSV на рис. 9. На рис. 9, в видно, що рівень середньої яскравості кадру помітно зріс — від 40 до 120 одиниць шкали яскравості, гістограма розподілилася більш рівномірно в діапазоні всієї шкали яскравості, але почала мати гребінчастий характер, візуальне сприйняття кадру стало дещо комфортнішим, проте з'явилися ознаки перенасиченості зображення. У разі використання колірному простору YIQ негативні тенденції тільки посилюються. Як видно на рис. 10, середній рівень яскравості кадру зріс ще більше (від 40 до 130 одиниць шкали), гістограма розподілу яскравостей стала більш гребінчастою, а візуальне сприйняття погіршилося до дискомфортного.

На підставі аналізу цих двох прикладів можна зробити висновок, що в розглянутих алгоритмах підвищення контрастності доцільно використовувати тільки перехід з колірному простору RGB у простір HSV. Для цього є свої причини. Еквалізація за каналом Y у форматі YIQ створює зображення, воно втрачає в контрастності та кольорі, руйнується колірний баланс. Тому що канал V в HSV — яскравість пікселя, а канал Y в YIQ — освітленість. Це означає, що еквалізація, застосована до каналу V в HSV, з більшою ймовірністю дасть візуально приємніші результати з підвищення контрастності, оскільки вона безпосередньо націлена на рівні яскравості зображення.

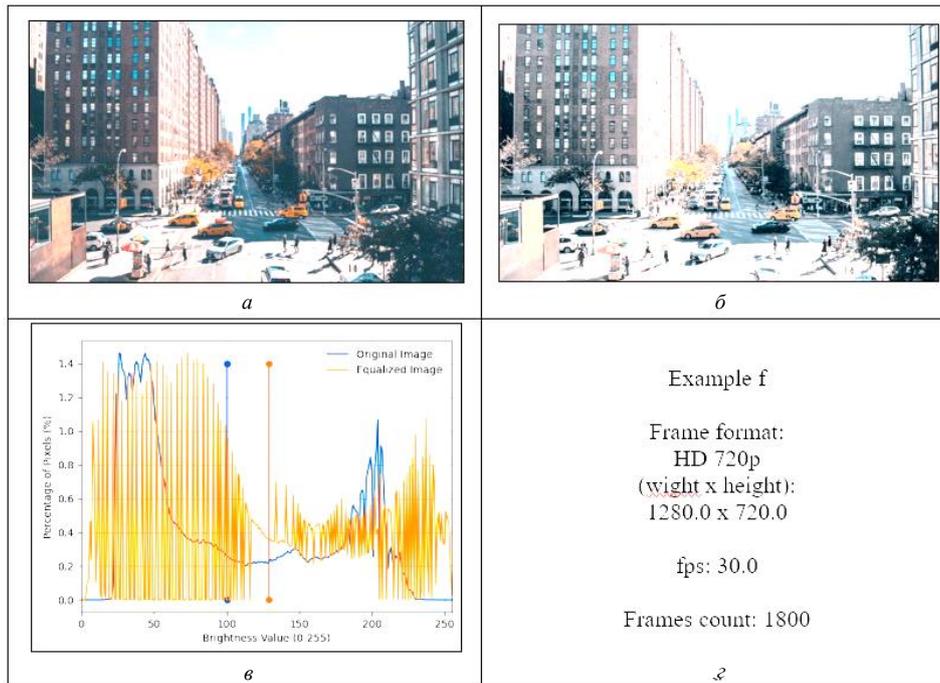


Рис. 8

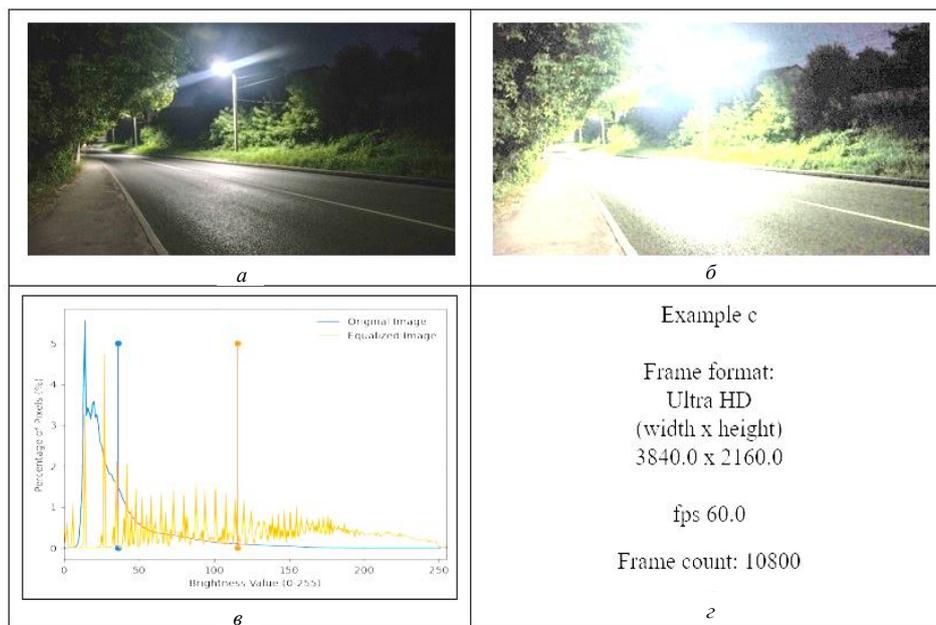


Рис. 9

Крім візуальних маркерів, для повноцінного аналізу розраховувалися і кількісні статистичні показники для всіх тестових відео, показаних на рис. 6, для алгоритму підвищення контрастності з переходом у простір HSV і алгоритму з використанням колірного простору YIQ. Результати розрахунків зведено в табл. 1 і 2 відповідно.

Аналізуючи кількісні показники, нескладно помітити, що використання алгоритму підвищення контрасту кадрів з застосуванням колірного простору HSV безумовно краще.

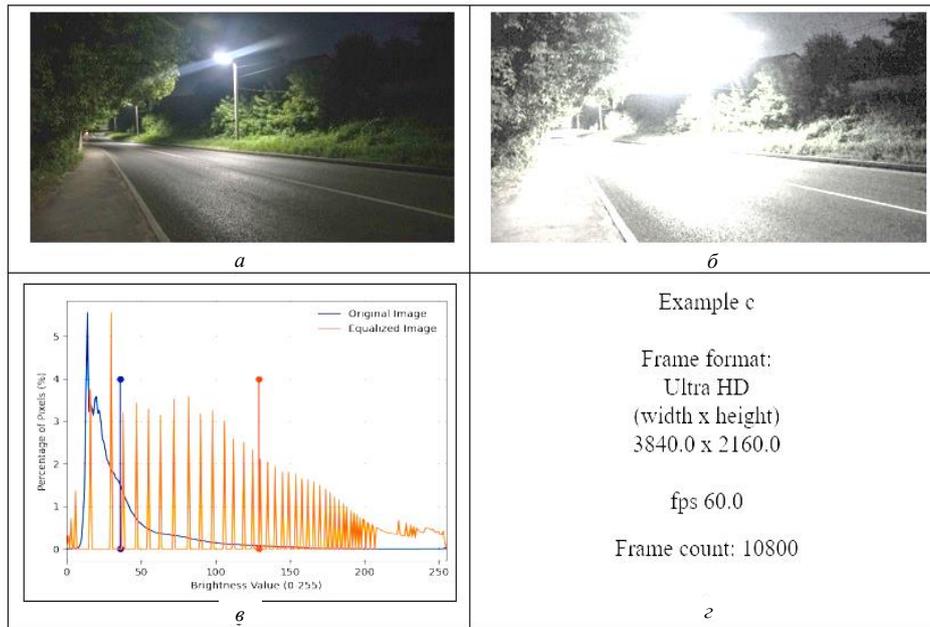


Рис. 10

Таблиця 1

Показники якості алгоритму покращення контрастності у просторі HSV							
Показник	A	B	C	D	E	F	Average
MSE	86,58	88,27	106,134	93,85	89,56	106,49	95,147
PSNR	28,76	28,67	27,87	28,41	28,61	27,86	28,36
NCC	0,9705	0,9766	0,8352	0,93	0,83	0,90	0,906
SSIM	0,87	0,88	0,35	0,77	0,41	0,58	0,643
AMBE	82,47	88,27	84,93	77,18	47,41	55,15	72,61

Таблиця 2

Показники якості алгоритму покращення контрастності у просторі YIQ							
Показник	A	B	C	D	E	F	Average
MSE	116,55	101,99	111,09	85,31	105,18	85	100,85
PSNR	27,47	28,04	27,67	28,82	17,91	28,84	26,458
NCC	0,96	0,9375	0,8	0,88	0,6755	0,8603	0,8522
SSIM	0,81	0,82	0,3	0,5	0,1	0,41	0,49
AMBE	116,55	93,26	99,09	75,11	100,88	75,55	93,4

Приклад 3. Розглянемо характеристики якості лінійного алгоритму стабілізації яскравості кадру з переходом у простір HSV. Візуальні результати оброблення показано на рис. 11, а кількісні оцінки якості — в табл. 3.

Для оцінки візуального сприйняття зображення кадру використано те саме тестове відео, що й у прикладі 1. Зверніть увагу, у цьому разі перетворене зображення кадру не викликає зорового дискомфорту. Зауважимо, що середня яскравість кадру трохи зменшилася (приблизно на п'ять одиниць шкали яскравості), але водночас істотно згладилася гістограма яскравості та зрушилася в бік зменшення яскравості. Це, що є дуже важливим в різних технічних додатках, свідчить про ефективну фільтрацію високочастотних коливань яскравості.

Наведемо ще один приклад обробки лінійним алгоритмом стабілізації яскравості кадрів зі заниженим рівнем освітлення сцени (тестове відео C). Результати розрахунків з переходом у простір HSV наведено на рис. 12.

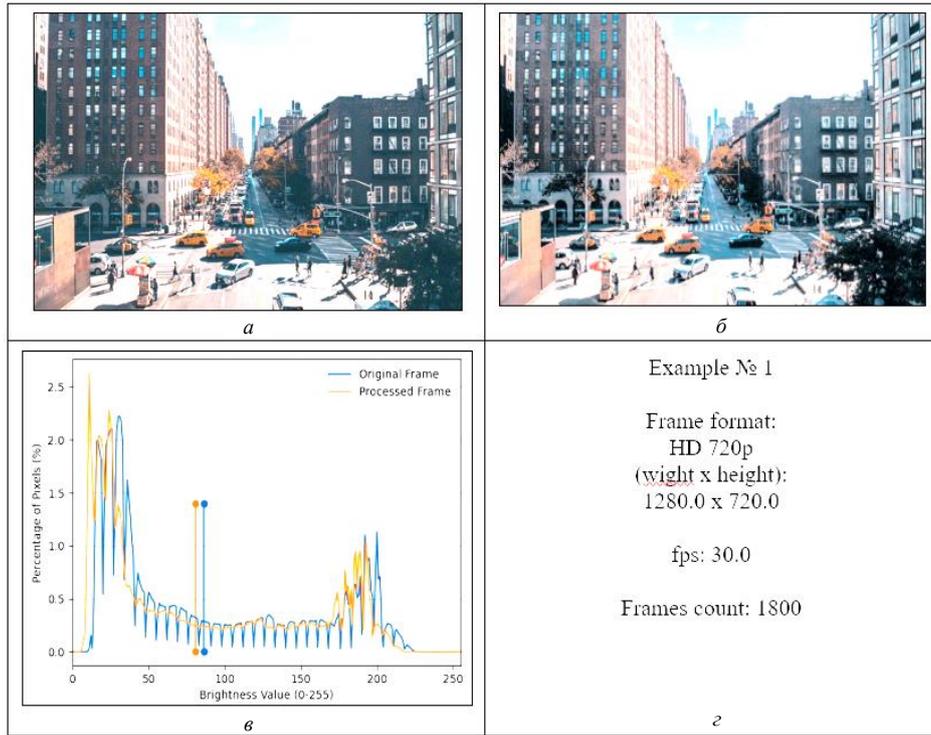


Рис. 11

Таблиця 3

Показники якості алгоритму лінійної стабілізації яскравості у просторі HSV							
Показник	A	B	C	D	E	F	Average
MSE	16,5710	14,40	18,49	12,85	11,10	15,40	14,80
PSNR	35,94	36,55	35,46	37,04	37,68	36,25	36,48
NCC	0,999	0,997	0,996	0,99	0,997	0,997	0,997
SSIM	1,0	1,1	0,99	1,0	0,95	0,98	0,97
AMBE	4,06	4,24	4,21	3,79	3,85	3,78	3,98

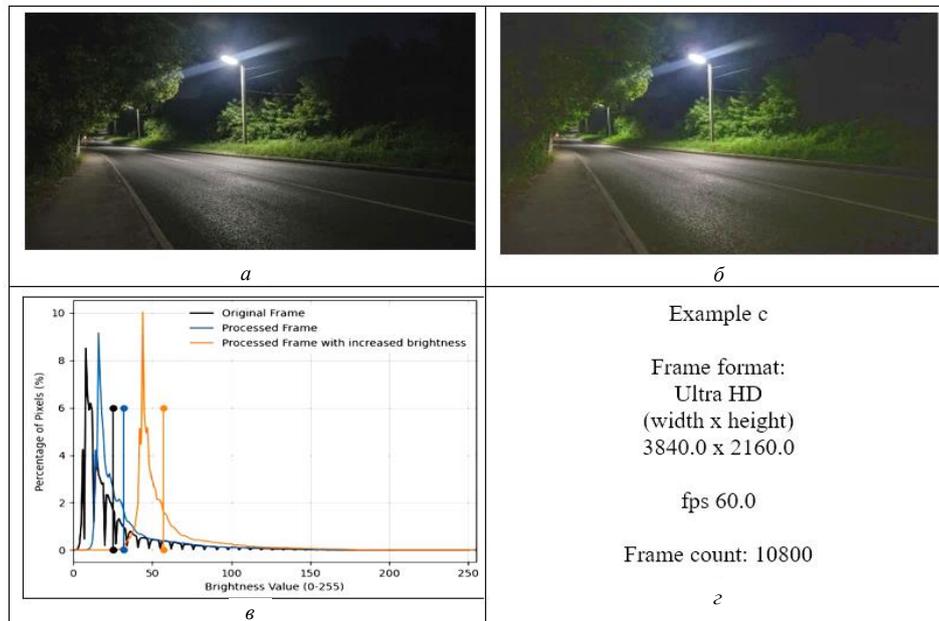


Рис. 12

Як і в попередньому прикладі, гістограма розподілу яскравості успішно згладилася, але форма її залишилася незмінною. Для фільтрації тренда середньої яскравості використано НЧ-фільтр з постійною часу 5 с. Відбулося невелике збільшення яскравості, яке складно помітити візуально. Але в цих випадках яскравість можна збільшити додаванням до кожного пікселя кількох градацій яскравості. У даному прикладі додано 30 одиниць. Зрозуміло, що ця добавка виконана із захистом від переповнення шкали яскравостей.

На жаль, обмежений обсяг публікації не дозволяє навести більшу кількість прикладів. Однак переваги запропонованого лінійного алгоритму стабілізації яскравості відеоданих, на думку авторів, цілком очевидні.

Аналіз можливості роботи алгоритму в реальному масштабі часу

Обробка відео в режимі реального часу є найважливішим показником якості використовуваних алгоритмів. Для цього застосовують різні методи і технології: ефективні алгоритми обробки зображень, апаратну підтримку, оптимізацію структури даних та архітектури програмного забезпечення. При цьому виникають певні складнощі — необхідно враховувати великий обсяг обчислень під час обробки кожного кадру. Паралельне опрацювання і використання спеціалізованих обчислювальних пристроїв допомагають впоратися з цією проблемою. Важливо враховувати і стійкість алгоритмів до змін параметрів відео. Час обробки має бути мінімізованим.

Для аналізу роботи алгоритму стабілізації яскравості в режимі реального часу розроблено окремий застосунок, інтерфейс якого представлено на рис. 13.

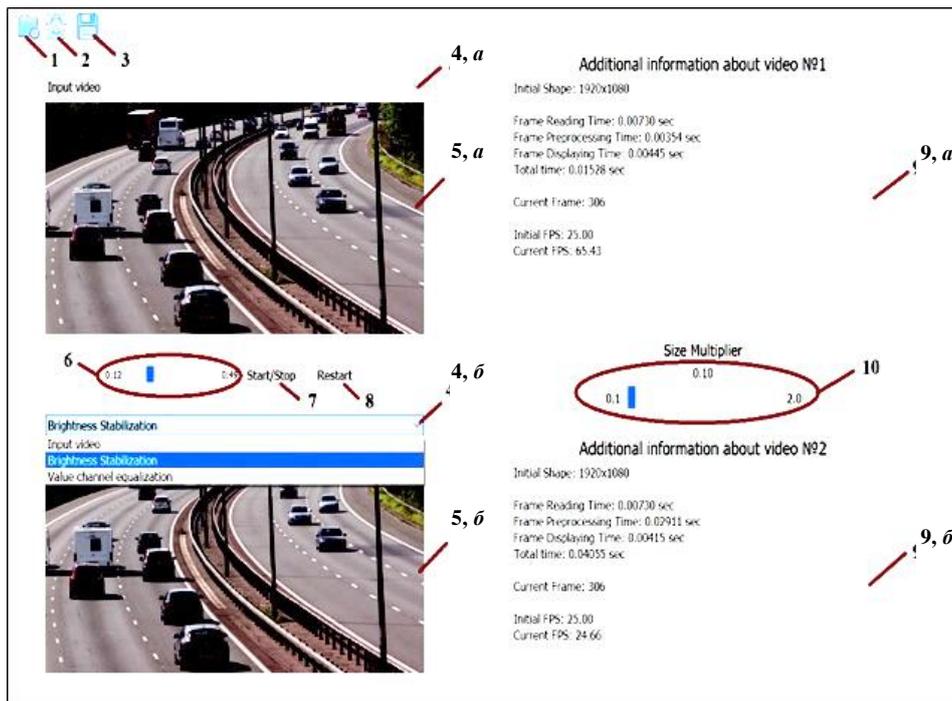


Рис. 13

Позначення на рис. 13: 1 — іконка для вибору поточного відеофайлу для роботи; 2 — іконка для початку роботи з відеоданими, отриманими з web-камери; 3 — іконка для побудови графіків залежностей параметрів відеопотоку; 4, а, б — списки, що випадають, для вибору варіанта відображення кадру; 5, а, б — вікна

відображення потокових кадрів відеопотоку; 6 — часова шкала для відеопотоку; 7 — кнопка для запуску/зупинки відеопотоку; 8 — кнопка для перезапуску відеопотоку; 9, а, б — інформаційні вікна для параметрів кожного з вікон відображення; 10 — повзунок адаптивного налаштування розмірного коефіцієнта алгоритму.

Цей додаток дозволив дослідити роботу лінійного алгоритму стабілізації яркості відеопотоку з метою вивчення швидкодії в порівнянні з алгоритмом еквалізації, зробити вибір оптимальних розмірів вихідних кадрів, провести аналіз впливу розмірного коефіцієнта ($Size_m$) на швидкість роботи. Як вихідні дані для досліджень використовувалися відеофайли з різною роздільною здатністю (1280x720, 1920x1080, 2560x1440, 3840x2160).

Спочатку досліджувалася залежність швидкодії алгоритму стабілізації яркості від значення розмірного коефіцієнта. Так, згідно з рис. 3 кожен кадр відеопотоку перетворюється у формат HSV і зменшується в розмірах для подальшої обробки. У [13, 14] показано, що робота алгоритму стійка до змін розмірності кадру, що дає змогу зменшити час обробки без втрати загальної якості. Отже, необхідно оцінити рівень впливу розмірного коефіцієнта на швидкодію алгоритму. Отримані результати показані на рис. 14, це графіки параметрів часу обробки та кількості кадрів за секунду від часу, вираженого в порядкових номерах кадрів.

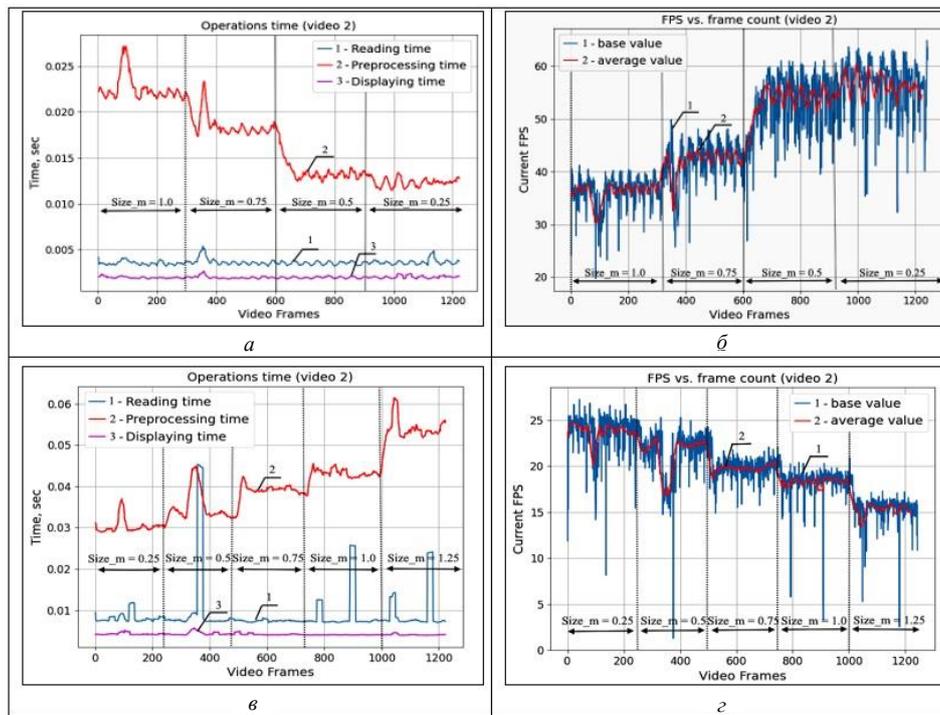


Рис. 14

На рис. 14, а і в показано часові витрати на повне виконання алгоритму з урахуванням зчитування окремих кадрів (Reading Time), обробки алгоритмом стабілізації яркості (Preprocessing Time) і відображення результату у вікні програми (Displaying Time). При цьому в обох прикладах виділено часові області, що належать до різних значень розмірного коефіцієнта ($Size_m$). На рис. 14, а зменшується значення коефіцієнта з 1,0 до 0,25, а на рис. 14, б — збільшується з 0,25 до 1,25. Проаналізувавши отримані графіки, можна помітити, що час читання і відображення кадрів не залежать від розмірного коефіцієнта $Size_m$.

Це повністю відповідає загальній ідеї, оскільки зміну розмірності виконують тільки в процесі алгоритму стабілізації яскравості для прискорення обчислювального етапу показника AFB, а читання і відображення виконують з кадром вихідного розміру. Головний висновок полягає в тому, що зменшення розмірного коефіцієнта ($Size_m$) призводить до зменшення часу опрацювання, однак при цьому яскраво спостерігається ефект насичення. Так, для першого прикладу під час зменшення розмірного коефіцієнта зі значення 1,0 до 0,75 час обробки зменшився на 17 %, а зі значення 0,5 до 0,25 — лише на 5 %. Відповідно надмірне зменшення розміру ($Size_m < 0,25$) не привнесе відчутного ($> 5\%$) прискорення роботи алгоритму.

Для оптимального значення розмірного коефіцієнта $Size_m = 0,25$ розраховано процентне значення прискорення роботи алгоритму. В результаті отримуємо, що для першого прикладу прискорення склало близько 35 %, для другого — 32 %. Зазначені тенденції можна побачити на рис. 14, б і рис. 14, з, оскільки значення Current FPS є зворотним значенням загального часу виконання. Отже, зменшення розмірного коефіцієнта призводить до збільшення FPS.

Крім значення розмірного коефіцієнта ($Size_m$), для оптимальної роботи алгоритму необхідно проаналізувати швидкість його роботи за різних розмірностей вихідного відеопотоку. Для цього проведено дослідження, під час якого для постійного значення $Size_m = 0,25$ використовувалися відеодані роздільних здатностей 1280x720, 1920x1080, 2560x1440, 3840x2160. Це стандартизовані розмірності, які часто зустрічаються під час запису в реальному масштабі часу. Отримані результати дослідження впливу розмірності вихідних кадрів на швидкодію представлені на рис. 15.

Можна помітити, що тільки для відеоданих розмірності 1280x720 вдалося одержати достатній рівень швидкодії, що відповідає значенню FPS понад 30. Однак під час роботи з відеоданими розмірності 1920x1080 середній рівень FPS перебуває на рівні 22 кадрів за секунду, що є прийнятним результатом для певного кола завдань, які не потребують максимальної швидкодії. Що стосується розмірів 2560x1440 і 3840x2160, швидкість роботи алгоритму абсолютно не дає змоги одержувати прийнятну якість вихідного відеопотоку, оскільки кількість кадрів за секунду перебуває на рівні менше 15 в обох випадках.

Зазначимо, що дослідження проводилися на ноутбучі з невисокими характеристиками (двоядерний процесор Intel(R) Core(TM) i5-6200U @ 2,3 ГГц, об'єм ОЗП — 8 ГБ, дискретна відеокарта GeForce 940MX), тому всі часові показники алгоритму можуть змінитися при виборі більш потужного і спеціалізованого обладнання.

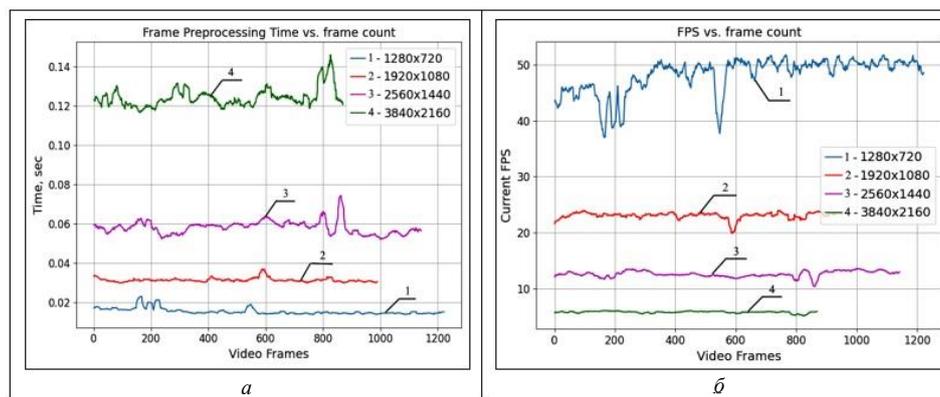


Рис. 15

Останнє дослідження становить собою порівняльний аналіз роботи лінійного алгоритму стабілізації яскравості та алгоритму еквалізації Value каналу в режимі реального часу. Для перевірки обрано відеофайл розмірності 1280x720 і розмірний коефіцієнт $Size_m = 0,25$. Отримані результати впливу розмірності вихідних даних на швидкість представлені на рис. 16 (video1 — стабілізація яскравості, а video2 — еквалізація).

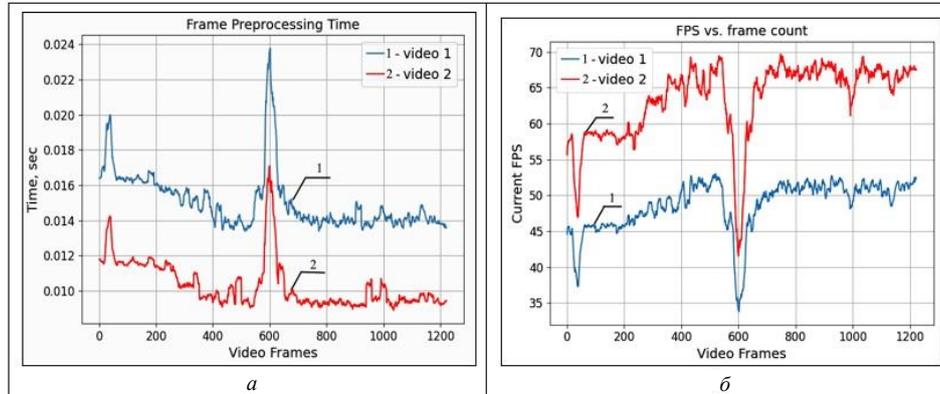


Рис. 16

Проаналізувавши рис. 16, *a* і *б*, можна зробити висновок, що алгоритм стабілізації яскравості поступається алгоритму еквалізації за показником швидкодії. Так, різниця в FPS становить у середньому 15 кадрів на секунду. Однак при цьому важливо зазначити, що порівняльна повільність роботи цінніша, ніж описані в попередніх розділах переваги. Також під час роботи з відеоданими розмірності 1280x720 або менше алгоритм стабілізації яскравості володіє достатньою швидкістю ($FPS \geq 30$) для оптимальної роботи, тому дана різниця у швидкості роботи з еквалізацією не впливає на виконання головного завдання, а саме, підвищення якості вихідних даних через стабілізацію яскравості.

Висновок

Результати досліджень дають змогу рекомендувати метод лінійної стабілізації яскравості відео для попереднього опрацювання даних у системах технічного зору під час розв'язання широкого кола прикладних задач. Показано, що перетворення вихідного відеопотоку не призводять до неконтрольованої зміни параметрів яскравості та контрастності і не порушують колірний баланс кадрів. Великою перевагою методу є можливість роботи в реальному масштабі часу. Вважаємо, що проведені дослідження актуальні та корисні. Надалі планується адаптувати отримані результати для використання їх в автономних мобільних системах технічного зору на базі мікрокомп'ютерів Raspberry Pi або їх аналогів.

V. Bilozerskyi, O. Bilous, K. Dergachov, L. Krasnov

PERFORMANCE EVALUATION OF A NOVEL METHOD FOR REAL-TIME BRIGHTNESS STABILIZATION OF VIDEO STREAMING

Vladyslav Bilozerskyi

National Aerospace University «Kharkiv Aviation Institute»,

v.o.bilozerskyi@khai.edu,

Oleksandr Bilous

National Aerospace University «Kharkiv Aviation Institute»,
aabelous@ukr.net

Konstantin Dergachov

National Aerospace University «Kharkiv Aviation Institute»,
k.dergachov@khai.edu

Leonid Krasnov

National Aerospace University «Kharkiv Aviation Institute»,
leonid.krasnov.1947@gmail.com

This article presents a comparative analysis of classical nonlinear methods for stabilizing the brightness and increasing the contrast of video data, and for the first time proposes an original method of linear brightness stabilization of a video stream and an algorithm that implements it, which works in real time. An objective assessment of the quality of video data brightness stabilization based on the results of pre-processing in comparison with nonlinear methods is given. The aim of the work is to create a methodology for analyzing the variability of video data parameters under the influence of negative factors and to develop effective algorithms for stabilizing video stream parameters; a thorough verification of the reliability of the new method is provided on real video recordings made in different conditions. To determine the most versatile, resistant to external influences and informative indicator necessary for an objective assessment of the quality of video data under different shooting conditions and scene lighting features; to develop and programmatically implement a linear algorithm that allows to effectively stabilize video brightness in real time. The development should be carried out on the basis of modern programming tools. Statistical analysis and pre-processing of video stream parameters as a random spatio-temporal process, algorithms for processing video data in real time by digital filtering and adaptive stabilization of video stream parameters. It is proposed and experimentally proved that the optimal indicator of video stream quality is the average frame brightness (AFB). A linear algorithm for spatio-temporal video data processing is proposed, which generates a sequence of AFB values from the original video stream. The paper also proposes digital algorithms for filtering and stabilizing the brightness of the video stream and investigates the effectiveness of their application. The scientific novelty of the obtained results lies in a new method for analyzing and evaluating the parameters of video surveillance data, algorithms for filtering and stabilizing the brightness of the video stream in real time. The performance of the proposed algorithms is tested on real data. The algorithms are implemented in the Python software environment using the functions of the OpenCV library.

Keywords: video stream, average frame brightness, video brightness trend, digital trend filtering, real-time brightness stabilization algorithms.

ПОСИЛАННЯ

1. Kim Y.T. Contrast enhancement using brightness preserving bi-histogram equation, *IEEE Transactions on Consumer Electronics*. 1997. Vol. 43, N 1. P. 1–8.
2. Ooi C.H., Mat Isa N.A. Adaptive contrast enhancement methods with brightness preserving. *IEEE Transactions on Consumer Electronics*. 2010. Vol. 56, N 4. P. 2543–2551.
3. Kim T.K., Paik J.K., Kang B.S. Contrast enhancement system using spatially adaptive histogram equalization with temporal filtering. *IEEE Transaction on Consumer Electronics*, 1998. Vol. 44, N 1. P. 82–86.
4. Sengee N., Choi H.K. Brightness preserving weight clustering histogram equalization. *IEEE Transactions on Consumer Electronics*. 2008. Vol. 54, N 3. P. 1329–1337.

5. Wang Q., Ward R.K. Fast image/video contrast enhancement based on weighted threshold histogram equalization. *IEEE transactions on Consumer Electronics*. 2007. Vol. 53, N 2. P. 757–764.
6. Chen S.-D., Ramli A.R. Minimum mean brightness error bi-histogram equalization in contrast enhancement. *IEEE Transactions on Consumer Electronics*. 2003. Vol. 49, N 4. P. 1310–1319.
7. Chen S.-D., Ramli A.R. Preserving brightness in histogram equalization based contrast enhancement techniques. *Digital Signal Processing*. 2004. Vol. 14. P. 413–428.
8. Chen S.-D. A new image quality measure for assessment of histogram equalization-based contrast enhancement technique. *Digital Signal Processing*. 2012. Vol. 22. P. 640–647.
9. Liang K., Ma Y., Xie Y., Zhou B., Wang R. A new adaptive contrast enhancement algorithm for infrared images based on double plateaus histogram equalization. *Infrared Physics & Technology*. 2012. Vol. 55. P. 309–315.
10. Raju A., Dwarakish G.S., Venkat D. Reddy a comparative analysis of histogram equalization based techniques for contrast enhancement and brightness preserving. *International Journal of Signal Processing. Image Processing and Pattern Recognition*. 2013. Vol. 6, N5, P. 353–366, DOI: <http://dx.doi.org/10.14257/ijsp.2013.6.5.31>
11. Su Y., Wu M., Yan Y. Image enhancement and brightness equalization algorithms in low illumination environment based on multiple frame sequences. *IEEE Access*. 2023. Vol. 11, P. 61535–61545.
12. Bilozerskyi V., Dergachov K., Krasnov L. Analysis and pre-processing of video data to improve the quality of computer vision systems. *International Scientific Technical Journal Problems of Control And Informatics*. 2023. Vol. 68, N 2, P. 50–66. DOI: <https://doi.org/10.34229/1028-0979-2023-2-4>. (In Ukrainian)
13. Bilozerskyi V., Dergachov K., Krasnov L. New method for video stream brightness stabilization: algorithms and performance evaluation. *Radioelectronic and computer systems*. 2023. N3. P. 125–135. DOI: <https://doi.org/10.32620/reks.2023.3.10>
14. Bilozerskyi V., Dergachov K., Krasnov L. New methods for video data pre-processing to improve the quality of computer vision systems. *IEEE 4th KhPI Week on Advanced Technology (KhPIWeek), Kharkiv, Ukraine*. 2023. P. 1–6. DOI: <https://doi.org/10.1109/KhPIWeek61412.2023.10312988>
15. Egiazarian K., Ponomarenko M., Lukin V., Ieremeiev O. Statistical evaluation of visual quality metrics for image Denoising. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, AB, Canada, 2018. P. 6752–6756. DOI: <https://doi.org/10.1109/ICASSP.2018.8462294>
16. Lukin V.V., Zriakhov M.S., Ponomarenko N.N., Krivenko S. S., Zhenjiang M. Lossy compression of images without visible distortions and its application. *IEEE 10th international conference on signal processing proceedings*. China : Beijing. 2010. P. 698–701. DOI: <https://doi.org/10.1109/ICOSP.2010.5655751>

Отримано 19.12.2023
Доопрацьовано 25.01.2024