

3. Захаров А.Н. Методика фазування антенної решітки активної ГАС на основі попарної кореляційної обробки [Текст] / А.Н. Захаров // Збірник наукових праць, СВМІ. Севастополь, -2006. - вип. 2 (10), - С. 172-180.
4. Захаров А.Н., Миронов И.Я. Фазування антенної решітки активної ГАС по алгоритму самоподавлення перешкоди [Текст] / А.Н. Захаров, И.Я. Миронов // Збірник наукових праць, СВМІ. Севастополь, -2008. - вип. 1 (11), - С. 133-140.
5. Ольшевский, В. В. Статистические свойства морской реверберации [Текст] / В. В. Ольшевский. - М. : Наука, 1965. - 360 с.
6. Справочник по гидроакустике [Текст] : / А. П. Евтютов, А. Е. Колесников, Е. А. Корепин и др. ; - 2-е изд. — Л. : Судостроение, 1988. - 552 с.

*Розглянуто дослідження розпаралелених алгоритмів сегментації зображень з використанням обчислень на графічному процесорі. Для порівняння алгоритмів за якістю та швидкодією обрано такі два підходи сегментації – аналіз гістограм та кластеризація. Наведено результати досліджень та практичні рекомендації до їх використання*

*Ключові слова: аналіз гістограм, сегментація, кластеризація, обчислення на графічному процесорі, опрацювання зображень, паралельні обчислення, порівняння алгоритмів*

*Рассмотрены исследования параллельных алгоритмов сегментации изображений с использованием вычислений на графическом процессоре. Для сравнения алгоритмов по качеству и быстродействию избрано такие два подхода сегментации - анализ гистограмм и кластеризация. Приведены результаты исследований и практические рекомендации к их использованию*

*Ключевые слова: анализ гистограмм, сегментация, кластеризация, вычисления на графическом процессоре, обработка изображений, параллельные вычисления, сравнение алгоритмов*

УДК 004.272.2+004.932

# ДОСЛІДЖЕННЯ РОЗПАРАЛЕЛЕНИХ АЛГОРИТМІВ СЕГМЕНТАЦІЇ ЗОБРАЖЕНЬ З ВИКОРИСТАННЯМ ОБЧИСЛЕНЬ НА ГРАФІЧНОМУ ПРОЦЕСОРІ

**Р. Б. Тушницький**

Кандидат технічних наук, старший викладач\*

E-mail: ruslan.tushnytskyy@gmail.com

**І. Я. Кожух**

Аспірант\*

E-mail: ill.kozh@gmail.com

\*Кафедра програмного забезпечення

Національний університет «Львівська політехніка»

вул. С. Бандери, 12, м. Львів, Україна, 79013

## 1. Вступ

Сегментація є одним з процесів обробки зображень, тісно пов'язаним з такими задачами, як розпізнавання об'єктів і пошук зображень за вмістом, та знаходить своє застосування у різних галузях науки. Завданням сегментації є розбиття зображення на деяку кількість частин – сегментів, що часто є одним з перших етапів обробки.

На даний час для сегментації зображень розроблений ряд алгоритмів з використанням різних математичних підходів, які дають змогу адаптувати її до конкретних задач галузі. У багатьох сферах розроблено платні та безкоштовні програмні засоби, що здійснюють сегментацію в контексті інших задач, а також більш вузькоспеціалізовані програми.

При залученні до розпаралелення графічного процесора очікується підвищення продуктивності робо-

ти, тобто суттєве пришвидшення сегментації завдяки одночасному виконанню операцій в багатьох потоках. Однак, не завжди теоретично придатний до розпаралелення алгоритм буде значно швидшим за послідовний на практиці.

Тому якість та швидкість сегментації, і як наслідок доцільність розпаралелення можуть бути визначені коректно лише при реалізації алгоритмів і експериментальному порівнянні їх роботи.

Таке дослідження дасть змогу не тільки підвищити швидкість роботи сегментації, а й встановити як найбільш ефективні з точки зору паралельної роботи алгоритми, так і найменш придатні до розпаралелення. Його результати можуть бути корисні для розробників програмного забезпечення та науковців, що працюють в області сегментації і розпаралелення, а також для всіх тих, хто стикається з потребою вибору підходів до сегментації зображень в інших галузях.

---

## 2. Постановка задачі дослідження алгоритмів сегментації

---

Використання сегментації у будь-якій галузі і для будь-яких задач має зміст для великого об'єму даних, оскільки при малій їх кількості користувач зможе виконувати дії вручну. Крім того, існують тенденції до збільшення як об'ємів колекцій зображень, так і розмірів самих зображень.

Тобто, застосування сегментації доцільне і актуальне саме для великої кількості зображень, але це вимагає значних затрат часу, особливо при високому їх розширенні.

Тому виникає одна з основних проблем – підвищення швидкодії.

Для підвищення швидкодії чи якості розроблено алгоритми, що базуються на різних підходах [1–9]. Однак, при цьому з'являється додаткова задача – їх практичне порівняння в однакових умовах, оскільки дослідники використовують різні критерії порівняння і різні засоби.

Хоч алгоритми сегментації зображень є ресурсоемкими, однак над кожним пікселем зображення виконуються схожі дії, і їх результат не завжди залежить від результатів обробки інших пікселів. Це дає змогу розпаралелити роботу, доцільним рішенням для чого є використання графічного процесора, який за будовою призначений саме для паралельних обчислень. Зокрема, в роботі [10] розглянуто практичне застосування розпаралелених алгоритмів сегментації з використанням обчислень на графічному процесорі для системи пошуку зображень за їх вмістом.

Однак, практичний приріст швидкодії неможливо точно спрогнозувати лише на основі аналізу існуючих опублікованих результатів, оскільки він залежить не тільки від самого алгоритму, а й від використовуваних апаратних і програмних засобів, які теоретично можуть дати вигоду у швидкодії, потрібно реалізувати і порівняти практично, провівши експериментальні дослідження в однакових умовах.

---

## 3. Методика проведення досліджень

---

### 3.1. Принципи порівняння якості

Вибір критерію порівняння якості є однією з складних проблем сегментації, для якої немає однозначного рішення. Зазвичай, під якістю сегментації розуміється ступінь відповідності отриманих сегментів реальним об'єктам, представленим на зображенні [11]. Тобто, виділені алгоритмом сегменти повинні якнайточніше відповідати об'єктам, які б виділила людина.

Всі методи оцінки якості можна розділити на дві категорії: суб'єктивні та об'єктивні. До суб'єктивних належать ті методи, де використовується людський фактор; до об'єктивних – всі інші, зокрема з використанням метрик [11].

Оскільки метою даного дослідження є порівняння швидкодії, порівняння якості носить допоміжний характер.

Тому порівняння якості в даній роботі здійснюється за допомогою сегментації одного зображення різними алгоритмами зі збереженням результатів в графічному вигляді. Користувач на основі їх візуального співставлення проводитиме повністю суб'єктивну оцінку якості.

### 3.2. Принципи порівняння швидкодії

Хоча технологія NVIDIA CUDA надає засоби для обчислення часу роботи як паралельних, так і послідовних алгоритмів, їх коректне практичне порівняння залишається складною задачею. Причиною цього є ряд важливих чинників, нехтування якими призводить до значних похибок в результатах.

Швидкодія залежить, перш за все, від апаратних засобів – центрального процесора, оперативної пам'яті та відеокарти, – а також від їх завантаження на момент роботи. Суттєвий вплив здійснює і те, як повно використовує алгоритм можливості CPU або GPU, чи враховано при адаптації їх переваги і недоліки.

Крім того, важливим є сам метод порівняння. Часто дослідники досягають багатократного приросту швидкодії, враховуючи лише час роботи ядра GPU [12]. Але для обробки на GPU необхідний ряд додаткових операцій обміну даних, які в сукупності можуть перевищувати час роботи самого ядра від 2 до 50 раз [12].

Тому, в межах даної роботи, під порівнянням швидкодії розуміється порівняння часу сегментації одного і того ж набору зображень, з врахуванням всіх операцій обміну даними. Це дозволить відобразити реальний вигоду у часі обробки, який отримує користувач при використанні алгоритму для своїх цілей.

Цей вигоду у часі відобразатиметься за допомогою приросту швидкодії – відношення часу обробки набору зображень послідовним алгоритмом до часу обробки того самого набору зображень паралельним алгоритмом.

---

## 4. Вибір алгоритмів сегментації для розпаралелення

---

Після проведеного аналізу різних підходів до сегментації зображень і з урахуванням будови GPU та особливостей технології CUDA, стає можливим коректний вибір алгоритмів для розпаралелення. Крім того, додатково розглянуто вже існуючі дослідження з використанням CUDA, що дає змогу оцінити придатність до розпаралелення і, як наслідок, доцільності розробки взагалі.

В результаті обрано такі підходи до сегментації:

- аналіз гістограм. Існують результати CUDA-реалізації, яка дає 30-кратний приріст швидкодії для побудови гістограм [13],

- кластеризація. Існують результати CUDA-реалізації, яка дає приріст швидкодії від 13 до 68 раз в залежності від типу і серії відеокарти [14].

Однак, крім обраних алгоритмів для розпаралелення в межах даної роботи, слід відзначити і деякі інші. Перспективними є і такі алгоритми, як виділення границь [15] та представлення графом [16], для яких теж існують реалізації на CUDA. Розростання областей вимагає окрім додаткового вводу, алгоритми автоматизації часто застосовують рекурсію, яка не підтримується в CUDA.

Генетичні алгоритми чи нейронні мережі використовуються, в основному, лише для виділення границь і є досить складними.

Зазначені результати досліджень [13–16] проаналізовано лише для оцінки можливості розпаралелення, вони не можуть бути використані для коректного співставлення підходів сегментації, оскільки умови їх отримання надто різні.

### 5. Апаратні засоби та їх конфігурація

Для проведення досліджень алгоритмів аналізу гистограми і кластеризації з врахуванням обраної методики, розроблено програмний продукт з використанням технології CUDA.

Характеристики апаратного забезпечення, з використанням якого отримано всі подальші результати, наведено у табл. 1.

Таблиця 1

Характеристики апаратного забезпечення

	Host-характеристики	Device-характеристики
Процесор	Intel Pentium 4 HT (Prescott): 1 фізичне ядро / 2 віртуальні, частота – 3,2 ГГц	NVIDIA GeForce 9600 GT: частота ядра – 600 МГц, кількість ядер CUDA – 64, частота шейдерів – 1500 МГц.
Пам'ять	DDR2 (об'єм – 2,5 Гб): сер. частота доступу – 266 МГц	GDDR3 (об'єм – 512 Мб): частота доступу – 900 МГц.

### 6. Порівняння алгоритмів за якістю

Для алгоритмів обох підходів проведено експерименти на різних типах зображень – тестовій базі зображень [17] та інших фотографіях і графіці, а також з різними параметрами роботи (від 2 до 20 сегментів). На основі порівнянь виявлено такі особливості:

- алгоритм аналізу гистограми базується лише на кольорі (гістограма не несе інформації про координати), виділяючи всі області з перепадами кольорів. Частіше за все межі сегментів, особливо для фотографій, нечіткі,

- алгоритм кластеризації враховує і колір, і координати, виділені області менш залежні від перепаду кольорів. Межі сегментів, в порівнянні з гістограмою, зазвичай більш чіткі.

Ці особливості, в залежності від конкретного зображення, можуть як позитивно, так і негативно впливати на якість.

Далі наведено основні виявлені переваги кожного алгоритму в порівнянні з іншим.

Алгоритм аналізу гистограми ефективний для виділення фону та об'єкту (тільки 2 сегментів), особливо для зображень, де їх можна явно визначити візуально. Сегментація здійснюється якісно майже з повною відповідністю сегментів об'єктам, як наведено на рис. 1. В даному прикладі об'єкт (квітка) чітко і точно відділений від фону (листя).

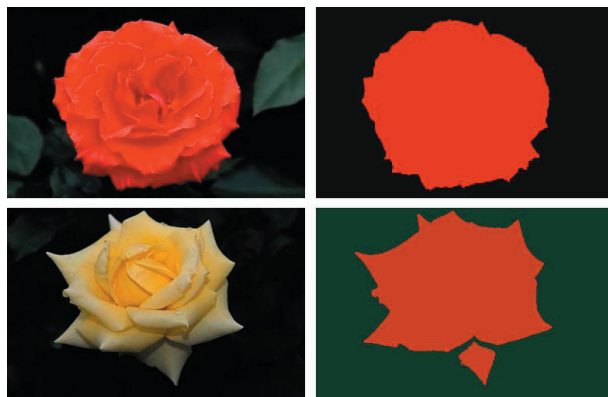


Рис. 1. Виділення фону і об'єкту аналізом гистограми

Якісною є обробка аналізом гистограми зображень комп'ютерної графіки, оскільки для них, зазвичай, межі об'єктів і кольори є більш чіткими і насиченими. Тому для них ефективно виділяються і більш, ніж 2 сегменти.

Алгоритм кластеризації якісно працює і з ще більшою кількістю сегментів для фото і графіки, де зображені різні кольорові об'єкти, дозволяючи виділити окремі їх деталі. Розбиття на сегменти наведено на рис. 2.



Рис. 2. Сегментація за допомогою кластеризації

В даному прикладі основні деталі (частини автобусів та споруд) чітко відокремлені, тоді як несуттєві (елементи дорожнього покриття, цегли, листя) об'єднані.

### 7. Порівняння алгоритмів за швидкістю

#### 7.1. Залежність від розміру зображень

Для визначення залежності приросту швидкодії від розміру зображень проведено ряд дослідів з використанням режиму порівняння швидкодії.

Спочатку здійснено порівняння алгоритмів для тестових зображень, розмір яких складає 384×256 (~0,1Мп).

Програма запущена 10 раз, на різних їх наборах (по 100 штук), але з одними і тими ж параметрами – 2 сегменти, 1-разовий прохід по файлах без збереження читання і запису. Результати наведено у табл. 2.

**Таблиця 2**

Порівняння алгоритмів для тестових зображень (384×256)

Алгоритм	Загальний час виконання, сек		Приріст швидкодії
	Послідовний	Паралельний	
Аналіз гістограми	0,676	0,759	0,89
Кластеризація	51,075	3,692	13,84

Час роботи паралельного алгоритму аналізу гістограми вищий, ніж послідовного – через надто малий час самої обробки в порівнянні з копіюванням даних, або незначуще розпаралелення при записі в вихідний масив. Час роботи паралельної кластеризації суттєво нижчий за послідовну, приблизно у 14 раз.

Крім того помічено, що алгоритми аналізу гістограми практично не залежать від категорій зображень (їх вмісту) – відхилення часу обробки 100 зображень будь-якої категорії від середнього часу не перевищувало 3%, – як для послідовного, так і паралельного алгоритму.

Натомість алгоритми кластеризації дають різні результати у залежності від категорії зображення. Час їх роботи коливався в межах 28-68 сек. для послідовного і 2,2-4,8 сек. для паралельного, що складає 51,075 ± 45% і 3,692 ± 40% відповідно. Однак, відхилення приросту становило не більше 10%. Така залежність від вмісту зображень простежувалась і в подальших дослідях.

Після цього проведено дослідження роботи алгоритмів для 10 зображень розміром 1600×1200 (~2Мп). Програма запущена 10 раз з одними і тими ж параметрами – 2 сегменти, 10-разовий прохід по файлах без врахування читання і запису. Результати наведено в табл. 3.

**Таблиця 3**

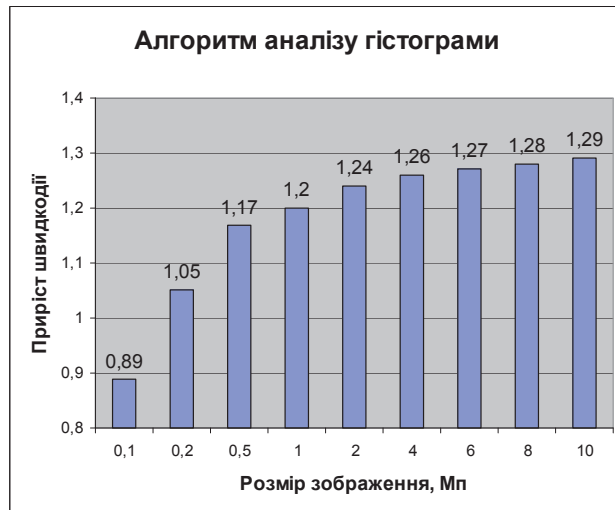
Порівняння алгоритмів для тестових зображень (1600×1200)

Алгоритм	Загальний час виконання, сек		Приріст швидкодії
	Послідовний	Паралельний	
Аналіз гістограми	13,002	10,441	1,24
Кластеризація	767,189	52,152	14,71

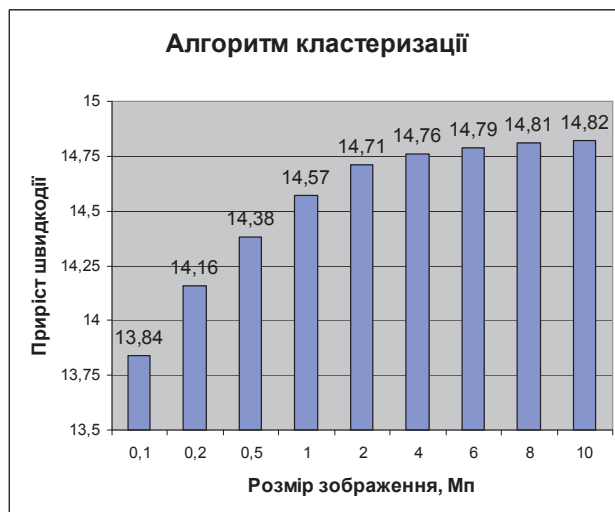
Зі збільшенням розміру зображень паралельний алгоритм аналізу гістограми стає швидшим за послідовний, приріст складає 1,24. Це спричинено ростом кількості потоків під час запису, тобто основний ефект алгоритму полягає саме у паралельному записі у вихідний масив, а не паралельною обробкою гістограм. Приріст швидкодії кластеризації збільшується майже до 15, також завдяки залученню більшої кількості потоків, але вже на двох етапах.

Отже, зі збільшенням розміру зображень приріст швидкодії зростає, тобто розпаралелення більш доцільне під час роботи з великими за розміром зображеннями. Для виявлення точної залежності проведено ще ряд дослідів з зображеннями різних розмірів. Узагальнені графіки залежності приросту швидкодії

від розміру зображень, для обох підходів з кількістю сегментів 2, наведено на рис. 3.



а)



б)

Рис. 3. Залежність приросту швидкодії від розміру зображень: а) алгоритм аналізу гістограми; б) алгоритм кластеризації

Як видно з графіків, функція залежності від розміру зображень для обох підходів сегментації є досить схожою.

**7.2. Залежність від кількості сегментів**

Для визначення залежності приросту швидкодії від кількості сегментів проведено ще ряд дослідів з використанням режиму порівняння швидкодії. Для цього обрано 100 різних зображень розміром 800×600 (~0,5Мп), також кількість запусків зменшена до 2-3 штук через досить тривалий час обробки.

Спочатку здійснено порівняння алгоритмів з такими ж параметрами, що використовувались в попередніх дослідях – 2 сегменти, 1-разовий прохід по файлах без врахування читання і запису. Після цього кількість сегментів збільшено до 5, інший параметр залишено без змін – 1-разовий прохід по файлах без врахування читання і запису.

Зі збільшенням кількості сегментів час роботи паралельного алгоритму аналізу гістограми починає зростати, що свідчить про неефективність паралельної реалізації пошуку порогів.

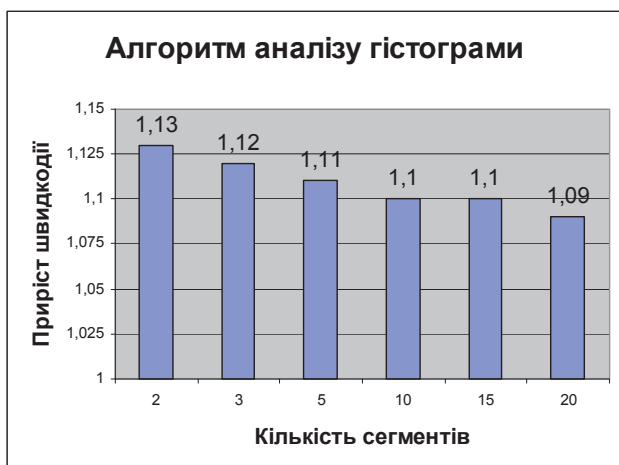
Час роботи послідовного алгоритму практично не міняється, тому приріст швидкодії дещо знижується – з 1,13 до 1,11.

Час роботи кластеризації зростає значно сильніше, оскільки з підвищенням кількості сегментів підвищується і кількість ітерацій, алгоритми довше збігаються.

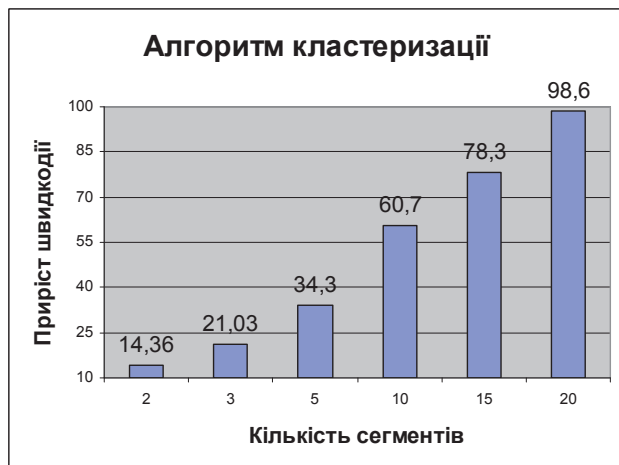
Особливо значним стає час роботи послідовного алгоритму, що спричиняє різке підвищення приросту швидкодії з 14,36 до 34,30.

Отже, зі збільшенням кількості сегментів приріст швидкодії зростає лише для для кластеризації, її розпаралелення стає не просто доцільним, а й необхідним, оскільки час роботи послідовного алгоритму надто високий. Для виявлення точної залежності проведено ще ряд дослідів з різною кількістю сегментів.

Узагальнені графіки залежності приросту швидкодії від кількості сегментів, для обох підходів на зображеннях розміру  $\sim 0,5$ Мп, наведено на рис. 4.



а)



б)

Рис. 4. Залежність приросту швидкодії від кількості сегментів: а) алгоритм аналізу гістограми; б) алгоритм кластеризації

Як видно з графіків, функція залежності від кількості сегментів дещо спадає для аналізу гістограми та різко зростає для кластеризації.

## 8. Результати досліджень

В результаті досліджень виявлено, що аналіз гістограми дає якісніші результати при виділенні 2 сегментів, аніж багатьох, досить якісно обробляє комп'ютерну графіку, однак іноді виділяє шум на зображеннях і нечіткі межі сегментів. Кластеризація, навпаки, краще працює з великою кількістю сегментів, їх межі є порівняно чіткішими, однак при цьому іноді розбиває однотонні об'єкти.

Час роботи аналізу гістограми виявився меншим за час кластеризації у 4-18 раз, без врахування операцій з файлами. Крім того, аналіз гістограми стабільний, а час роботи кластеризації коливається в залежності від вмісту зображення.

Приріст швидкодії при розпаралеленні аналізу гістограми незначний і не перевищує 1,3, а в деяких випадках менший за 1. Повільно зростає зі збільшенням розміру зображень (від 0,9 до 1,3 для 2 сегментів), і дуже повільно спадає зі збільшенням кількості сегментів (від 1,13 до 1,09 для зображень  $\sim 0,5$ Мп).

При розпаралеленні кластеризації приріст суттєвий, і навіть з врахуванням всіх операцій з файлами майже досягає 5.

В умовах багаторазової обробки одного зображення він є значно більшим. Повільно зростає зі збільшенням розміру зображень (від 13,8 до 14,8 для 2 сегментів) і дуже швидко зростає зі збільшенням кількості сегментів (від 14 до 100 для зображень  $\sim 0,5$ Мп). В крайніх випадках час роботи послідовної кластеризації був надто високим.

## 9. Висновки

Існує велика кількість опублікованих праць, що описують розпаралелені алгоритми сегментації і їх приріст швидкодії. Однак порівнювати різні підходи до сегментації лише на основі аналізу відомих результатів некоректно, оскільки апаратні засоби дослідників та самі методи обчислення часу роботи є надто різними. В даній роботі всі алгоритми виконувались в однакових умовах, що дало змогу порівняти їх швидкодю. Завдяки врахуванню всіх операцій під час обчислення часу виконання алгоритмів, встановлено, що аналіз гістограми є дуже швидким навіть у послідовній версії і його розпаралелення не приносить для користувача відчутного ефекту.

Алгоритм кластеризації є значно повільнішим, однак ефективно розпаралелюється з багатократним приростом швидкодії.

Проведені дослідження підтвердили особливості щодо якості алгоритмів, зокрема більшу ефективність аналізу гістограми для виділення фону та об'єкту, вищу чіткість меж сегментів при кластеризації.

Результати можуть бути корисні для розробників програмного забезпечення та науковців, що працюють в галузі сегментації зображень та розпаралелення алгоритмів.

## Література

1. Kurugollu, F. Color image segmentation using histogram multithresholding and fusion [Текст] / F. Kurugollu, B. Sankur, A. E. Harmanci // *Image and Vision Computing*, Vol. 19, Issue 13. – 2001. – P. 915-928.
2. Lopes, N. V. Automatic histogram threshold using fuzzy measures [Текст] / N. V. Lopes, P. A. Mogadouro do Couto, H. Bustince, P. Melo-Pinto // *IEEE Transactions on Image Processing*, Vol. 19, Issue 1. – 2010. – P. 199-204.
3. Lezoray, O. Color image segmentation using morphological clustering and fusion with automatic scale selection [Текст] / O. Lezoray, C. Charrier // *Pattern Recognition Letters*, Vol. 30, Issue 4. – 2009. – P. 397-406.
4. Chitade, A. Z. Colour based image segmentation using k-means clustering [Текст] / A. Z. Chitade, S. K. Katiyar // *International Journal of Engineering Science and Technology*, Vol. 2, Issue 10. – 2010. – P. 5319-5325.
5. Wang, H. Generalizing edge detection to contour detection for image segmentation [Текст] / H. Wang, J. Oliensis // *Computer Vision and Image Understanding*, Vol. 114, Issue 7. – 2010. – P. 731-744.
6. Felzenszwalb, P. F. Efficient graph-based image segmentation [Текст] / P. F. Felzenszwalb, D. P. Huttenlocher // *International Journal of Computer Vision*, Vol. 59, Issue 2. – 2004. – P. 167-181.
7. Espindola, G. M. Parameter selection for region-growing image segmentation algorithms using spatial autocorrelation [Текст] / G. M. Espindola, G. Camara, I. A. Reis, L. S. Bins, A. M. Monteiro // *International Journal of Remote Sensing*, Vol. 27, Issue 14. – 2006. – P. 3035-3040.
8. Senthilkumar, N. Edge detection techniques for image segmentation – A survey of soft computing approaches [Текст] / N. Senthilkumar, R. Rajesh // *International Journal of Recent Trends in Engineering*, Vol. 1, Issue 2. – 2009. – P. 250-254.
9. Chen, J. Adaptive perceptual color-texture image segmentation [Текст] / J. Chen, T. N. Pappas, A. Mojsilovic, B. E. Rogowitz // *IEEE Transactions on Image Processing*, Vol. 14, Issue 10. – 2005. – P. 1524-1536.
10. Kozhuh, I. CBIR System Using CUDA Technology [Текст] / I. Kozhuh, R. Tushnytskyy // *Proceedings of the VIIIth International Conference “Perspective Technologies and Methods in MEMS Design” (MEMSTECH’2012)*. – Lviv-Polyana, 2012. – P. 60-61.
11. Zhang, H. Image segmentation evaluation: A survey of unsupervised methods [Текст] / H. Zhang, J. E. Fritts, S. A. Goldman // *Computer Vision and Image Understanding*, Vol. 110, Issue 2. – 2008. – P. 260-280.
12. Gregg, C. Where is the data? Why you cannot debate CPU vs. GPU performance without the answer [Текст] // C. Gregg, K. Hazelwood. – *ISPASS*, Austin, 2011. – P. 134-144.
13. Shams, R. Efficient histogram algorithms for NVIDIA CUDA compatible devices [Текст] / R. Shams, R. A. Kennedy. – *ICSP-CS*, Gold Coast, 2007. – P. 418-422.
14. Farivar, R. A parallel implementation of k-means clustering [Текст] / R. Farivar, D. Rebodello, E. Chan, R. Campbell. – *International Conference on PDPTA*, Las Vegas, 2008. – P. 340-345.
15. Catanzaro, B. Efficient, high-quality image contour detection [Текст] / B. Catanzaro, B. Y. Su, N. Sundaram, Y. Lee, M. Murphy, K. Keutzer. – *IEEE International Conference on Computer Vision*, Kyoto, 2010. – P. 2381-2388.
16. Vineeth, V. CUDA cuts: Fast graph cuts on the GPU [Текст] / V. Vineeth, P. J. Narayanan. – *Workshop on Visual Computer Vision on GPUs*, Anchorage, 2008. – P. 1-8.
17. Test database (1000 test images) [Электронный ресурс] / J. Z. Wang Research Group]. – Режим доступа: <http://wang.ist.psu.edu/~jwang/test1.tar> (2013).