УДК 004.056.53

МАРКОВСЬКИЙ О.П.,
НЕВДАЩЕНКО М.В.,
БІЛАШЕВСЬКА А.М.

ЗАХИЩЕНА РЕАЛІЗАЦІЯ ФІЛЬТРАЦІЇ ЗОБРАЖЕНЬ В GRID-СИСТЕМАХ

В статті пропонуються методи захищеної реалізації медіанної та середньоарифметичної фільтрації на віддалених процесорних засобах GRID-систем. Методи забезпечують захист від доступу до зображень при їх передачі та під час їх обробки на віддалених комп'ютерних системах. Метод базується на використанні інтервального шифрування для медіанної фільтрації і використання адитивного маскування точок зображення для середньоарифметичної фільтрації. Детально описані процедури шифрування зображень і їх дешифрування після фільтрації. Наведено числові приклади шифрування та дешифрування зображень для обох типів фільтрації.

This paper proposes a method for protected implementation of image digital median and averaging filtering at remote processors of GRID-systems. Proposed methods ensures protection of image against unauthorized access while image network transmission and while image processing on remote computer systems. The methods for median filtering image protection is based on interval encryption and developed method for averaging filtering image protection is based on using of additional masking. The proposed procedure for image encryption and decryption are described in details. A numerical example for procedure image encryption and decryption for both type of image filtering are given.

Вступ

Спіралевидний характер розвитку комп'ютерних технологій призвів на початку нового тисячоліття до повернення на якісно новому рівні до колективного використання обчислювальних потужностей: віддаленої обробки задач користувачів на потужних багатопроцесорних комп'ютерних системах.

Динамічний прогрес способів телекомунікації та мережевих технологій дозволив широкому колу різнорідних користувачів прямо використовувати значні обчислювальні ресурси сучасних кластерних систем для вирішення наукових та прикладних задач. Це суттєво прискорило обробку прикладних задач, допомогло підвищити рівень їх розв'язування, а також, як наслідок, залучити до комп'ютерної обробки більш широке коло нових задач, які, в силу їх значної ресурсоємності, не могли бути вирішені на персональних комп'ютерах.

Технології віддаленого надання розподілених обчислювальних ресурсів активно розвиваються в рамках GRID-систем [1]. Особливістю таких систем є використання рознесених у просторі ресурсів. При цьому GRID-система бере на себе функцію забезпечення ефективного використання комп'ютерних систем шляхом динамічного розподілення потоку задач користувачів. Відповідно, звертаючись до GRID-системи, користувач в принципі не знає на якій з комп'ютерних систем буде розв'язуватися його задача. Відкритість GRID-систем зумовлює їх основний недолік – незахищеність від несанкціонованого доступу до їх задач користувачів. В сучасних умовах доступ до даних та задач користувачів в GRID-системах суттєво обмежує коло їх застосування і як результат помітно знижує ефективність віддаленого надання обчислювальних ресурсів [2].

Однією з найбільш поширених прикладних задач є обробка зображень. Висока ресурсоємність процедур обробки зображень, що містять мільйони пікселів диктує доцільність використання потужних багатопроцесорних комп'ютерних систем. Разом з цим, характер обмежень доступу до зображення не дозволяє, в більшості випадків, їх обробку у відкритих системах, в тому числі GRID-системах. Відповідно, виникає практична потреба в організації захищеної обробки зображень в сучасних відкритих розподілених комп'ютерних системах великої потужності [3].

Таким чином, наукова задача створення методів і засобів захищеної обробки зображень і, зокрема їх фільтрації, в GRID-системах є актуальною та важливою для практики.

Аналіз існуючих методів фільтрації зображень та її захищеної реалізації

Одним з основних процедур обробки зображень є їх фільтрація, яка дозволяє позбутися імпульсних завад [3]. Будь-яка фільтрація виконується над зображенням, яке можна подати

в наступному вигляді: $I= \{f(x,y);\ x \in [1;l],\ y \in [1;h]\}$ де I– зображення; $f(x,y)$– функція значення інтенсивності кольору в точці з координатами(x,y); l та h– відповідно ширина та висота зображення.

Якщо вважати, що будь-яке зображення має вади, то функцію $f(x,y)$ буде сумою значень інтенсивностей оригінального зображення $g(x,y)$ та вад $v(x,y)$, які виникли під час формування зображення: $f(x,y)= g(x,y)+ v(x,y)$.

Отже, метою фільтрації є зменшення значення інтенсивності вад $v(x,y)$.

Найбільшого поширення вважаються медіанна та середньоарифметична фільтрації [3]. Операція полягає в тому, що зображення сканується апертурою, що містить непарну кількість рядків та стовпців. В процесі сканування центральний елемент замінюється медіаною або середнім арифметичним точок апертури. Операція починається з верхнього лівого кутка і , поступово зміщуючись, охоплює весь об'єм зображення.

Операція медіанної фільтрації полягає в тому, що зображення сканується квадратною апертурою з непарною кількістю рядків та стовбців. Під час сканування значення інтенсивності точок апертури сортуються та серед ним вибирається елемент, що стоїть посередині, тобто являється медіаною. Значення інтенсивності замінюється на значення медіани і апертура переходить до обробки наступної точки. Обробка розпочинається з лівого верхнього кута, далі апертура переміщується по рядку, а в кінці рядка переходить на першу точку наступного рядка.

Приклад на рисунку 1 ілюструє принцип медіанної фільтрації. Для зручності взято за приклад зображення 4x4 із застосуванням апертури 3x3. Зліва показано початкове зображення. Далі зліва направо приведена послідовність зміни зображення в процесі його медіанної фільтрації. На малюнку виділено апертури, які використовуються в процесі фільтрації.

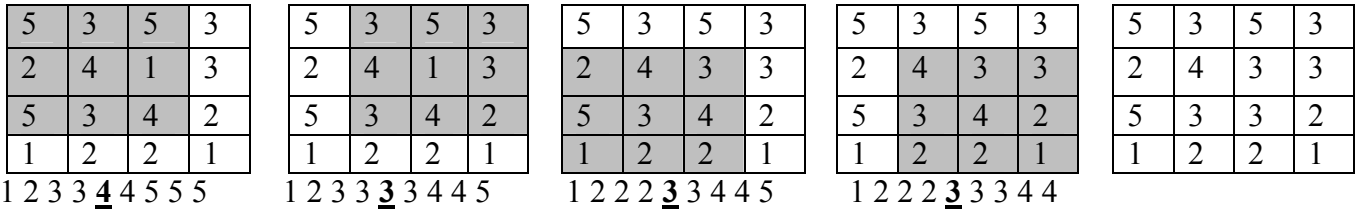

Рис.1. Приклад медіанної фільтрації

Операція середньоарифметичної фільтрації полягає в тому, що зображення сканується квадратною апертурою з непарною кількістю рядків та стовбців. Під час сканування знаходиться середнє значення інтенсивності точок апертури, яке замінює значення інтенсивності точки. Обробка розпочинається з лівого верхнього кута, далі апертура переміщується по рядку, а в кінці рядка переходить на першу точку наступного рядка.

Приклад на рисунку 2 ілюструє принцип середньоарифметичної фільтрації. За приклад взято зображення 4x4 із застосуванням апертури 3x3. Зліва показано початкове зображення. Далі, зліва направо приведена послідовність зміни зображення в процесі його середньо арифметичної фільтрації. На малюнку виділено апертури, які використовуються в процесі фільтрації.

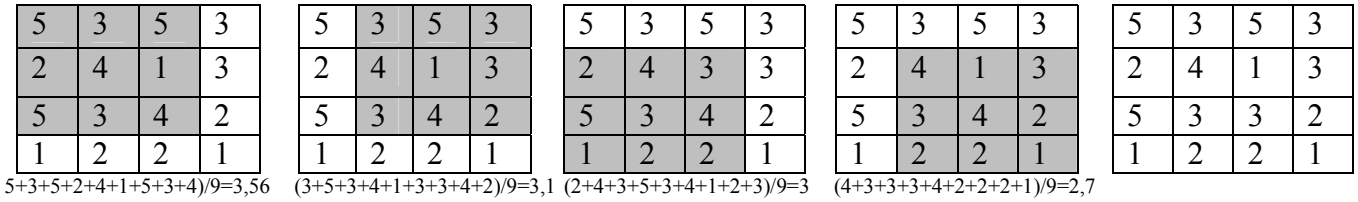

Рис.2. Приклад середньоарифметичної фільтрації

Описані вище методи фільтрації зображень не орієнтовані на захищену реалізацію на відкритих віддалених комп'ютерних системах. Разом з тим, як зазначалося вище, така задача з появою і початком широкого застосування GRID-систем набуває дедалі більшої актуальності.

До теперішнього часу, на відміну від захисту інформації при її передачі та зберіганні, проблема захисту даних в процесі їх безпосередньої обробки в загальному вигляді не вирішена [4]. Якщо в процесі обробки над даними a виконується функціональне перетворення $s = f(a)$ то захищена реалізація передбачає попереднє

функціональне перетворення $b=\varphi(a,m)$, де m-секретний ключ і передачу для безпосередньої обробки коду b. Головною проблемою захищеної реалізації є одержання функції відновлення $\psi(x,m)$, $s=\psi(f(\varphi(a,m)),m)=f(a)$. При цьому обчислювальна складність функціональних перетворень φ та ψ має бути суттєво меншою в порівнянні з f.

Практично, до теперішнього часу для кожної процедури (функції f) віддаленої обробки даних вибирається свої перетворення φ та ψ, які забезпечують вказані вище умови ефективної захищеної реалізації.

Ціллю досліджень є розробка ефективних методів захисту даних зображень безпосередньо під час їх віддаленої фільтрації шляхом отримання процедур шифрування точок зображень та дешифрування результатів фільтрації.

Захищена реалізація медіанної фільтрації

Оскільки в основі медіанної фільтрації лежить операція порівняння, то при шифруванні має зберігатись відносний порядок між точками зображення : якщо для точок зображення a_i і a_j виконується $a_i > a_j$, то і для зашифрованих представлень відповідних точок b_i і b_j має так само виконуватись $b_i > b_j$. Для того, щоб ця умова виконувалась, пропонується використати інтервальне шифрування. Основою такого методу є те, що розрядність d представлення кодів точок зображення менша за розрядність u зашифрованих представлень цих точок. При цьому, якщо кількість можливих значень точок становить r, то діапазон 0 до 2^u-1 можливих значень зашифрованих точок випадковим чином розділяється на r інтервалів з границями : $g_1, g_2, \ldots, g_r$, причому $g_1 < g_2 < \ldots < g_r$. Якщо позначити точки зображення через $a_1, a_2, \ldots, a_N$, де $N=h \cdot l$-кількість точок зображення, то запропонований спосіб захищеної реалізації медіанної фільтрації на відкритих GRID-системах полягає в наступному.

1) Створюється впорядкований масив, присутніх в зображенні значень точок, що a'_1, a'_2, a'_3, a'_4, ... a'_k , де k – загальна кількість можливих інтенсивностей зображення(для прикладу, показаного на рис. 1, k =4).

2) Обирається інтервал X=(0, 2^u-1), d'-розрядність представлення точок зображення у шифрованому вигляді.

3) Інтервал X розбивається випадковим чином на k проміжків: Xj, де j=1,k . Кожному значенню a'_j відповідає інтервал X_j .

4) Для кожної точкою виконується операція її шифрування: $\forall\ i \in \{1 \ldots N\}\, b_i = r_i$. Де r_i – випадкове число, що належить інтервалу Xj : $a_i = a'_j$.

5) Масив чисел b_1, b_2, b_3, b_4, ... b_N передається на обробку системі, яка потім повертає масив значення точок відфільтрованого шифрованого зображення q_1, q_2, q_3, q_4, ... q_N .

6) Користувач виконує дешифрацію точок $q_1, q_2, q_3, q_4, q_5, \ldots, q_N$ отриманого зображення: $\forall i \in \{1,2,\ldots,N\}$: $w_i = a'_j$, який відповідає інтервалу Xj так, що $q_i \in Xj$,де w_i . реальні відфільтровані значення .

Запропонований спосіб шифрування цілком відповідає зазначеним вище теоретичним умовам. Впорядкувавши вибірку кольорів і поставивши їм у відповідність проміжки поступово розбитого інтервалу, виконується головна вимога – збереження порядку числа, що робить можливим порівняння шифрованих аналогічне порівнянню реальних значень. Це забезпечує коректність дешифрування.

Запропонований спосіб захищеної реалізації медіанної фільтрації може бути ілюстровано наступним прикладом фільтрації для прикладу, зображеного на рис. 1. Спершу впорядкується вибірка можливих кольорів:

$a'_1 = 1$, $a'_2 = 2$, $a'_3 = 3$, $a'_4 = 4$, $a'_4 = 5$.

Оскільки розрядність реальних чисел – 3, то для шифрованих можна обрати будь-яке більше, з розрахунком на те, що більший проміжок для позначення числа дає більший захист за рахунок збільшення діапазону випадкових варіантів. Для розглянутого прикладу було обрано розрядність u = 7. Тоді основний інтервал – X=(0, 127).

Співвідношення між можливими значеннями та випадковими інтервалами ілюструє таблиця 1, наведена нижче:

Табл1. Розподіл можливих значень за інтервалами

a'_j	X_j
1	0...26
2	27...53
3	54...84
4	85...111
5	112...127

Приклад на рисунку 3 ілюструє принцип фільтрації зашифрованих значень. Зліва показано початкове зображення, що утворюється за допомогою обраного методу шифрування. Далі зліва направо приведена послідовність зміни зображення в процесі його медіанної фільтрації. На малюнку виділено апертури, які використовуються в процесі фільтрації.

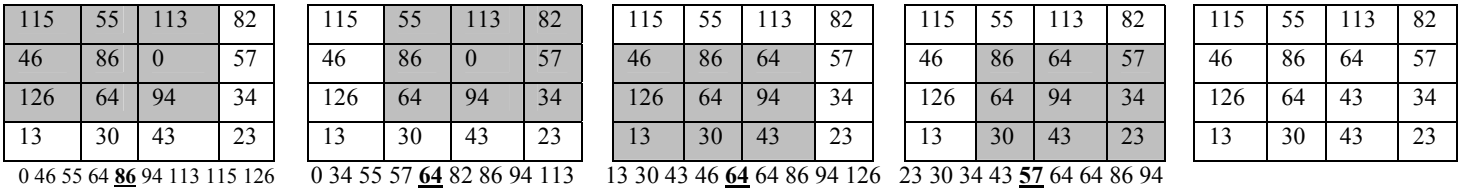

Рис.3. Приклад захищеної медіанної фільтрації

Дешифровані медіани:

1) 86-4
2) 64-3
3) 64-3
4) 57-3

Одержані результати співпадають з представленими рис.1 результатами звичайної реалізації медіанної фільтрації.

Метод захищеної реалізації середньоарифметичної фільтрації.

В основі середньоарифметичної фільтрації лежить операція обчислення середнього арифметичного точок поточної апертури. Ця операція включає арифметичне додавання та ділення. Найпростішим способом шифрування операндів арифметичного додавання є використання адитивного маскування, тобто додавання до кожного із операнду випадкового цілого, кратного певному модулю m. Зняття такої маски може бути доволі просто виконане шляхом знаходження залишку від ділення суми на модуль m, за умови, що модуль m більший за суму. Зняття адитивної захисної маски після операції ділення можливе за умови, що ділене буде кратним як подільнику, так і модулю m.

З урахуванням наведеного, спосіб шифрування, що пропонується зводиться до накладання на точки реального зображення адитивної складової кратної простому числу m, яке за значенням більше будь-якої точки зображення. Якщо позначити значення n реальних точок апертури як a_1, a_2, a_3, a_4, ... a_n, а через b_1, b_2, b_3, ... b_n - відповідно захищені від читання точки цієї апертури, то формально процес шифрування точок апертури можна представити у вигляді: $\forall j \in \{1,..,n\}$: $b_j = a_j + r_j \cdot m$, де r_j – випадкове ціле число. Для того, щоб результат фільтрації q, що формується як середнє арифметичне значень b_1, b_2, b_3, ... b_n, міг бути відтворений користувачем, його значення також має бути сумою середнього арифметичного s точок a_1, a_2, a_3, a_4, ... a_n та адитивної компоненти кратної m: $q = s + d \cdot m$, де d - ціле число. Таким чином, значення q фактично обчислюється у вигляді:

$$q = \frac{1}{n} \cdot \sum_{j=1}^{n} b_j = \frac{1}{n} \cdot \sum_{j=1}^{n} a_j + \frac{m}{n} \cdot \sum_{j=1}^{n} r_j = s + d \cdot m \qquad (1)$$

Відновлення фільтрованої точки s початкового зображення по зашифрованому коду q, як слідує з (1) реалізується через обчислення залишку від ділення q на m: $s = q \bmod m$. Коректність такого перетворення базується на тому, що $s<m$ (в силу того, що середнє арифметичне менших за m також менше за m), та на тому, що d - ціле.

Так як d має бути цілим, то з (1) слідує, що сума $r_1+r_2+...+r_n$ має бути кратною числу n точок апертури. В процесі фільтрації обчислене значення q замінює центральну $(n+1)/2$-ту точку $b_{(n+1)/2}$ апертури. Так як випадкові коефіцієнти генеруються користувачем до передачі зображення в GRID-систему з урахуванням зазначеної вище умови кратності суми коефіцієнтів апертури її розміру, то для того, щоб при заміні центральної точки апертури середнім значенням q ця умова не порушувалась, коефіцієнт $r_{(n+1)/2}$ центральної. $(n+1)/2$-ї точки має дорівнювати середньому арифметичному коефіцієнтів точок апертури:

$$r_{(n+1)/2} = \frac{1}{n} \cdot \sum_{j=1}^{n} r_j \qquad (2)$$

Таким чином, для захищеної реалізації середньоарифметичної фільтрації на віддалених комп'ютерних системах зображення, що містить N точок: $a_1,a_2,...,a_N$, пропонується спосіб, який зводиться до наступної послідовності дій користувача.

1) Генеруються випадковим чином цілі числа $r_1,r_2,...,r_N$, по кількості N точок зображення

таким чином, щоб для кожної апертури з n точок їх сума була кратна n і виконувалася умова (2). Вказані дії не прив'язані до конкретного зображення і можуть виконуватися заздалегідь.

2) Для зображення обирається модуль m, що є простим числом, більшим, за значення будь-якої точки зображення: $\forall i \in \{1,2,...,N\}: m > a_i$.

3) Для кожної з точок зображення виконується її шифрування накладанням адитивної маски: $\forall i \in \{1,2,...,N\}: b_i = a_i + r_i \cdot m$.

4) Зашифровані точки $b_1,b_2,...,b_N$ зображення передаються на GRID-систему для віддаленої фільтрації.

5) В GRID-системі виконується віддалена фільтрація на вільних обчислювальних потужностях і повертається користувачеві в вигляді кодів $s_1,s_2,....,s_N$ точок відфільтрованого зображення.

6) Користувач виконує дешифрацію точок $s_1,s_2,....,s_N$ отриманого зображення шляхом обчислення залишку від ділення: $\forall i \in \{1,2,...,N\}: v_i = s_i \bmod m$.

Запропонований метод захищеної реалізації середньоарифметичної фільтрації в GRID-системах може бути ілюстровано прикладом його застосування для фільтрації зображення на рис.1.

Згідно в викладеною вище методикою, випадковим чином формується N=16 цілих коефіцієнтів $r_1,r_2,...,r_{16}$ таких, що їх сума в рамках будь-якої з 4-х апертур розміром 3 х 3 ділиться на 9 і при цьому частка від ділення дорівнює коефіцієнту центрального елементу апертури. Сформовані випадкові коефіцієнти для прикладу наведені в таблиці 1. Легко пересвідчитись, що, наприклад, для правої нижньої апертури, що включає точки 6,7,8, 10,11,12,14,15 і 16, $r_6 + r_7 + r_8 + r_{10} + r_{11} + r_{12} + r_{14} + r_{15} + r_{16} =$ 5+6+5+8+9+10+5+5+28= 81; тобто сума ділиться на 9 і коефіцієнт r_{11} центрального елементу апертури дорівнює середньому значенню її коефіцієнтів: $r_{11} = 9 = 81/9$.

Відповідно до п.2 запропонованої методики для заданого зображення випадковим чином обирається модуль m: більше за будь-яке значення просте число , за значенням більше ніж будь-яка точка зображення. Для прикладу, що розглядається m=7. Таблиця 2 відображає шифрування значень точок зображення.

Приклад на рисунку 4 ілюструє принцип фільтрації зашифрованих значень. Зліва показано початкове зображення, що утворюється за допомогою обраного методу шифрування, та поступове обчислення середніх значень. Далі зліва направо приведена послідовність зміни зображення в процесі його середньоарифметичної фільтрації. На малюнку виділено апертури, які використовуються в процесі фільтрації.

Номер точки	a_i	r_i	b_i
1	5	1	12
2	3	2	17
3	5	3	26
4	3	6	45
5	2	4	30
6	4	5	39
7	1	6	43
8	3	5	38
9	5	7	54
10	3	8	59
11	4	9	67
12	2	10	72
13	1	23	162
14	2	5	37
15	2	5	37
16	1	28	197

Табл2. Шифрування точок

12	17	26	45
30	39	43	38
54	59	67	72
162	37	37	197

12	17	26	45
30	38,5	43	38
54	59	67	72
162	37	37	197

12	17	26	45
30	38,5	45	38
54	59	67	72
162	37	37	197

12	17	26	45
30	38,5	45	38
54	58,8	67	72
162	37	37	197

115	55	113	82
46	86	0	52
126	64	43	34
13	30	43	23

(12+17+26+30+39++43+54+59+67)/9=38,56
(17+26+45+38,5+43+38+59+67+72)/9=45
(30+38,5+45+54+59+67+162+37+37)/9=58,8
(38,5+45+38+58,8+67+72+37+37+197)/9=65,59

Рис.4. Приклад захищеної середньоарифметичної фільтрації

В результаті фільтрації, від системи користувач отримує наступні значення відфільтрованих точок 4-х апертур: s_6=38,5; s_7=45; s_{10}=58,8; s_{11}=65,59. Інші точки зображення повертаються системою в незмінному вигляді. Дешифрування змінений точок, згідно з викладеним вище, виконується користувачем наступним чином.

$v_6 = s_6 \, mod \, 7 = 38.56 \, mod \, 7 = 3.56 = 4$;

$v_7=s_7$ *mod* 7=45 *mod* 7=3;

$v_{10}=s_{10}$ *mod* 7=58.8 *mod* 7=2.8=3;

$v_{11}=s_{11}$ *mod* 7=65.59 *mod* 7=2.59=3;

Порівняння наведених значень з прикладом на рис.2 свідчить проте, що результат захищеної за запропонованим методом фільтрації повністю співпадає з результатами незахищеної середньоарифметичної фільтрації.

Аналіз ефективності

Основною перевагою запропонованих методів є можливість виконання масових операцій обробки зображень на відкритих GRID-системах з захистом самого зображення від несанкціонованого читання під час передачі та обробки. Відповідно, ефективність запропонованих методів оцінюється двома критеріями: рівнем захищеності, що досягається при їх застосуванні та часом, потрібний на їх реалізацію.

Задачею порушення захисту є відновлення істинного зображення, тобто встановлення за кодами $b_1,b_2,...,b_N$ значення невідомих кодів $a_1,a_2,...,a_N$ точок істинного зображення. При застосуванні запропонованого методу для захисту віддаленої реалізації медіанної фільтрації для цього зловмисник має виявити границі k інтервалів, на які розділяють діапазон представлення 2^u чисел закодованого зображення. Очевидно, що число ν варіантів виділення k інтервалів в комбінаторному плані дорівнює числу вибору k предметів з 2^u, яке згідно [5] визначається як $\upsilon = C_{2^u}^k = \frac{2^u!}{(2^u-k)!k!}$. При умові $k<<2^u$, що найчастіше має місце на практиці, значення $\upsilon \approx 2^{u\cdot k}/k! \approx (2^u \cdot e/k)^k \cdot \sqrt{2\cdot\pi\cdot k}$. Очевидно, що для типових для практики значень u та k число ν є доволі великим, для того, щоб гарантувати неможливість порушення захисту. Наприклад, при використанні 32-розрядних чисел (u=32) для кодування точок зображень, інтенсивність яких представляється 16-розрядним кодом (k=2^{16}) значення числа ν близьке до 10^{314572}, що практично повністю виключає порушення захисту шляхом перебору.

Час виконання операцій шифрування та дешифрування точок зображення значною мірою залежить від організації виконання цих операцій користувачем.

При обробці серії однотипних зображень, що найчастіше зустрічається на практиці, можна попередньо виконати допоміжні обчислення, пов'язані з розділенням на інтервали і генерацією випадкових чисел в цих інтервалах. За цих умов можна використовувати технології безколізійного хешування (perfect hashing) [5], які дозволяють уникнути використання бінарного інтервального пошуку. Тоді, запропонований спосіб шифрування та дешифрування зображень при віддаленій медіанній фільтрації зводиться до звертання до пам'яті при обробці кожної із точок зображення, тобто обчислювальна складність становить O(N). Обчислювальна складність фільтрації становить O(N·n^2) [3]. Таким чином, обчислювальна складність шифрування в n^2 раз менша за складність операції фільтрації. Наприклад, для розміру апертури 7 x 7 (n=49), складність операції шифрування складає менше одного відсотка від складності операцій медіанної фільтрації.

При виконанні середньоарифметичної фільтрації порушення захисту зображення, що передається для фільтрації в віддалену обчислювальну систему полягає в підборі модуля m, що лежить в інтервалі від k від 2^u. Технологія підбору полягає, що для вибраної точки перебираються всі k можливі її значення. Для кожного з цих значень перебираються можливі значення модуля m з вказаного вище інтервалу. Таким чином, об'єм перебору становить $k\cdot(2^u-k)$. Наприклад, при k=2^{16} і u=32, об'єм вказаного перебору становить $10^{14.4}$, що свідчить, що рівень захищеності запропонованого методу шифрування приблизно відповідає рівню захищеності відомого алгоритму DES [4].

Складність виконання операції фільтрації визначається тим, що для кожної з N точок зображення потрібно виконати n операцій додавання і одну операцію ділення. Шифрування точок може бути зведено до однієї операції додавання за умови, що масив добутків випадкових чисел на модуль сформовано заздалегідь. Дешифрування кожної точки зводиться до однієї операції ділення. Якщо вважати, що операція ділення по часу виконання відповідає приблизно 44-м операціям додавання [6], то можна вважати, що складність дешифрування в 1+n/44 раз менша складності фільтрації.

Висновки

В результаті проведених досліджень запропоновано методи захищеної реалізації масових операцій обробки зображень - медіанної та середньоарифметичної фільтрації. На основі проведеного аналізу операцій фільтрації запропо-

новано інтервальний метод шифрування точок зображення перед передачею його для обробки в GRID-системи. Метод забезпечує виконання медіанної фільтрації на віддалених відкритих комп'ютерних системах, закриваючи при цьому доступ до справжнього зображення.

Доведено, що за рівнем захищеності та часом реалізації запропонований метод захищеної реалізації медіанної фільтрації значно ефективніший в порівнянні з розробленим методом середньоарифметичної фільтрації. Сама медіанна фільтрація виконується швидше ніж середньоарифметична фільтрація.

З проведеного аналізу можна зробити висновок, що ефективність запропонованих методів захищеної реалізації фільтрації збільшується при зменшенні кількості градацій точок зображень. Найбільша ефективність захисту досягається при віддаленій обробці бінарних зображень.

Список посилань

1. Петренко А. Національна GRID інфраструктура для забезпечення наукових досліджень і освіти / А.Петренко Системні дослідження та інформаційні технології.- 2008 - № 4. - С.79-92.
2. Hennessy, John L. Computer architecture: a quantitative approach/ John L Hennessy, David A Patterson. – Amsterdam, Noord-Hooland, Netherlands : Elsevier, 2011. – 824 p.
3. Гонсалес Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс; пер. с англ. В.В. Харитонова.- М.:Техносфера, 2005. - 1072 с.
4. Шнаер Б. Прикладная криптография / Б. Шнаер.; пер.с англ. П.В. Семьянов.- М.:Издательство Триумф, 2003.- 816 с.
5. Кнут Д. Искусство программирования для ЭВМ / Д. Кнут.; пер.с англ. Н.И.Вюковой, В.А. Галатенко, А.Б.Ходулева.- М.:Мир, 1978.- 840 с.
6. Брэй Б. Микропроцессоры Intel. Архитектура, программирование и интерфейсы. Шестое издание / Б. Брэй; перю с анг. А.В.Жукова.- Санк-Петербург: БХВ-Петербург, 2005.-1328 с.