УДК 01.05.02; 05.13.06; 05.13.21

А. Ковальчук, Д. Пелешко, А. Шкодин, О. Троян
Нціональний університерт “Львівська політехніка”,
кафедра автоматизованих систем управління

ПРО ОДИН АЛГОРИТМ ШИФРУВАННЯ-ДЕШИФРУВАННЯ ЗОБРАЖЕНЬ З ВИКОРИСТАННЯМ ПОРОЗРЯДНИХ ОПЕРАЦІЙ

Побудовано стійкий алгоритм шифрування та дешифрування зображень, який задовольняє вимоги: забезпечення широкого діапазону ключа, забезпечення захисту від основних методів криптоаналізу, виключення можливості візуального розпізнавання деталей зображення.

Ключові слова: стійкий алгоритм, діапазон ключа, зображення, криптоаналіз.

An algorithm is resistant encryption and decryption of images that satisfy the requirements of: providing a wide range of key protection of the basic methods of cryptanalysis, removing the possibility of visual recognition of image details.

Keywords: sustainable algorithm, a range of key, image, cryptanalysis.

Вступ

У сучасному світі інформація має також матеріальну цінність, як у комерційній сфері, так і у сфері державної безпеки, де збереження її конфіденційності є критичним. Зокрема, такою інформацією можуть бути наукові розробки, бізнес-стратегії, військові плани тощо. Велику частку такої інформації становлять зображення.

З розвитком технологій та нарощуванням потужності комп’ютерної техніки застосування методів криптоаналізу стає менш складним, як і безпека інформації, зашифрованої «застарілими» методами. Потреба у розробленні нових алгоритмів вкрай висока. Крім того, виникають тенденції створення криптометодів, спеціалізованих для конкретних типів даних, таких як текст, зображення, звук, відео тощо.

Цифрові зображення потребують особливої уваги [1], оскільки вони містять візуальну інформацію, що потрібно враховувати з погляду шифрування. Алгоритм шифрування повинен, окрім криптостійкості, ще й унеможливлювати візуальне розпізнавання деталей зображення.

Оскільки зображення можна розуміти як дискретний сигнал, кожен відлік якого є значенням кольору або інтенсивності одного пікселя, а число відліків відповідно дорівнює кількості пікселів, то можна розглядати шифрування зображення як деформацію сигналів [2].

Для поставленої мети добре підходять тригонометричні функції, оскільки вони чутливі до зміни свого значення при зміні значення аргумента, внаслідок чого забезпечується певна стійкість результату та відповідно ускладнюється можливість аналізу в зв’язку із нелінійністю перетворень значень інтенсивностей кожного пікселя.

Мета роботи

Побудова стійкого алгоритму шифрування та дешифрування зображень, а саме:

- забезпечення широкого діапазону ключа;
- захист від основних методів криптоаналізу;
- унеможливлення візуального розпізнавання деталей зображення.

Характеристики зображення

Розглянемо зображення як послідовність дискретних сигналів з числом пікселів *w*h*, де *w* – ширина зображення , а *h* відповідно висота [3-4]. Тоді представимо зображення у вигляді вектора послідовних пікселів:

$$\vec{p} = (a_0, a_1, \dots, a_{w*h-1}) \quad (1)$$

де a_i відповідно значення інтенсивності *i*-го пікселя,

Залежно від зображення розмір a_i може складати 1 (для чорно-білих) або 3 (кольорові зображення) байти. Згідно зі сказаним вище, максимальне значення, котре може набути a_i, позначимо як *maxa*, значення 255 для чорно-білих та 16777215 для кольорових.

Опис алгоритму шифрування-дешифрування

Під ключем будемо розуміти символьний рядок довільної довжини *m* і позначимо його як вектор $\overrightarrow{key}$. Довільне початкове значення ключа шифрування позначимо F_0, для прикладу виберемо $F_0 = 23f3cba2_{16}$.

Функцію перетворення ключа шифрування визначимо так:

$$F_i(\overrightarrow{key}, F_{i-1}) = \sum_{j=0}^{m-1}(key[j] * \cos^j(F_{i-1}) + key[m-j] * \sin^j(F_{i-1}) + F^2_{i-1})$$

де F_{i-1} - попереднє значення функції, при $i = 1$ у функцію передається F_0.

Алгоритм шифрування

1. Обчислюється значення функції від F_0:
$$f = F(\overrightarrow{key}, F_0).$$
2. Отримане значення додається за модулем 2 до поточного значення
$$a'_i = a_i \oplus (f \ \& \ maxa),$$
де a'_i – зашифроване значення.
3. Обчислюється наступне значення функції
$$f = F(\overrightarrow{key}, f)$$
4. Повторюються кроки 2, 3 поки усі елементи вектора не будуть опрацьовані, тобто *w*h* разів.

Дешифрування

Для здійснення дешифрування, через симетрію операції додавання за модулем 2, до зашифрованого зображення достатньо застосувати пункти 1-4 алгоритму шифрування. На виході отримаємо дешифроване зображення

Наведений алгоритм шифрування можна представити у вигляді схеми.

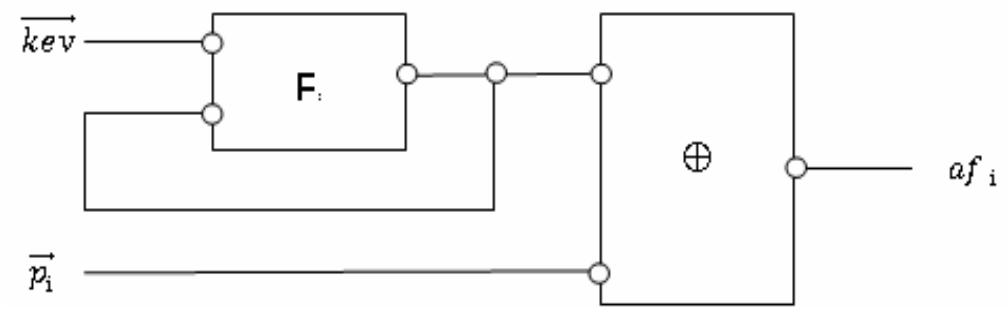

Рис. 1. Схема шифрування

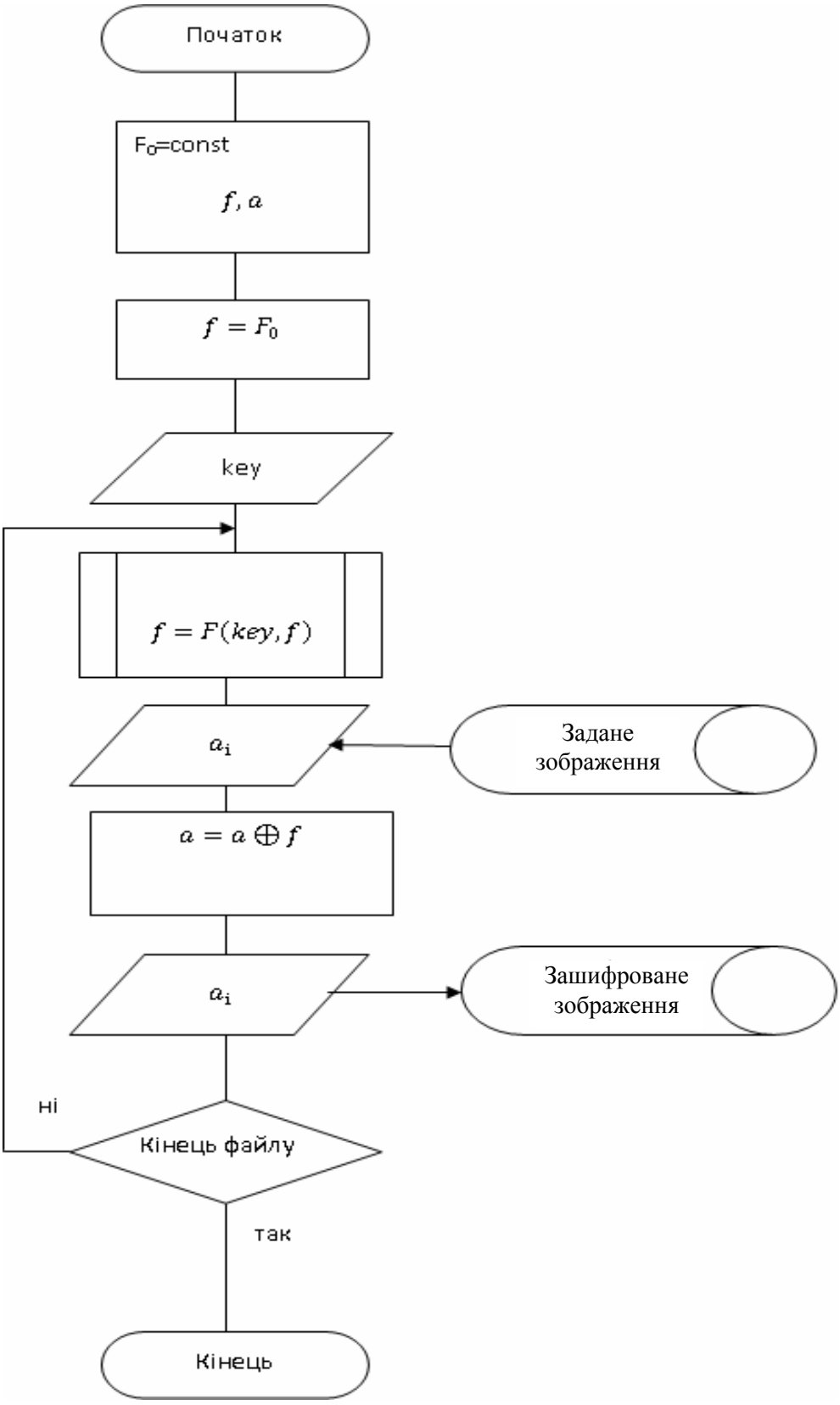

Рис. 2. Блок-схема шифрування-дешифрування

Результати роботи алгоритму над чорно-білим зображенням показано на рис. 3–5.

Рис. 3. Початкове зображення

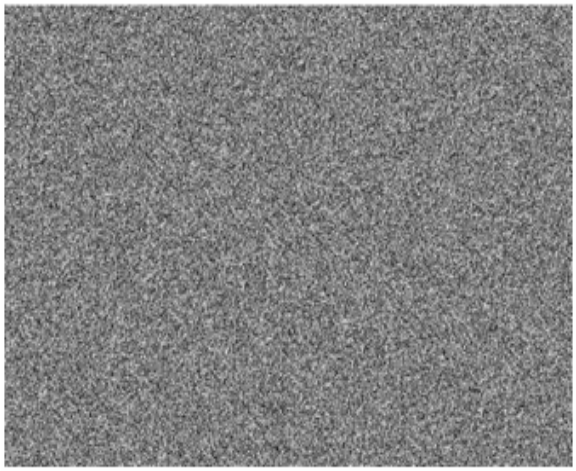

Рис. 4. Зашифроване зображення

Рис. 5. Дешифроване зображення

Результат роботи алгоритму над кольоровими зображеннями показано на рис. 6–8.

Рис. 6. Початкове зображення

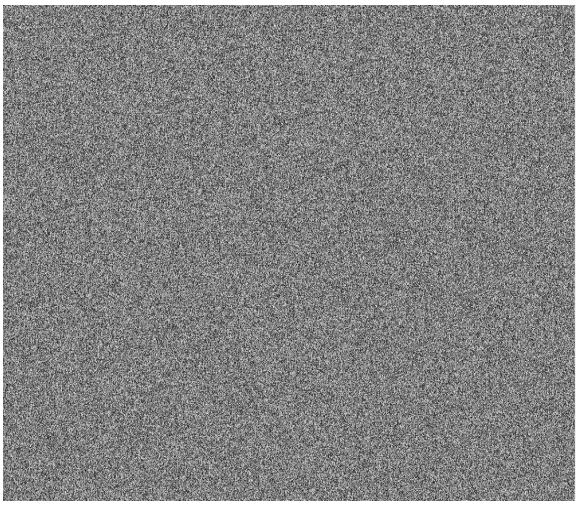

Рис. 7. Зашифроване зображення

Рис. 8. Дешифроване зображення

Як видно з рис. 4 та рис. 7, результат шифрування візуально є доволі добрим, зашумленість результуючого зашифрованого зображення не дає змоги виділити будь-яку закономірність чи візуально розпізнати деталі зображення. Початкове та дешифроване зображення ідентичні, тобто виконується символічна рівність (піксельна)

$$\forall i,\ 0 \leqslant i \leqslant (w * h),\ a_i = a_i'.$$

Фрагменти цифрових значень матриць інтенсивностей пікселів кольорових зображень у текстовому поданні з використанням ключа *key* = 7A8F145F і F_0 = 23f3cba216 показано на рис. 9–11.

```
-4143656   -4077863   -4077350   -4077350   -4142630   -4142630
-4208936   -4143143   -4142630   -4142630   -4142630   -4076837
-4208423   -4208423   -4142630   -4142630   -4141860   -4141860
-4207653   -4273446   -4273446   -4207653   -4207396   -4141603
-4207653   -4207653   -4273189   -4207396   -4141603   -4075810
-4207908   -4207908   -4207396   -4141603   -4075810   -4010017
-4207908   -4142115   -4076067   -4010274   -4075810   -4010017
-4142115   -4076322   -4010274   -3944481   -4010017   -4010017
-4207653   -4207653   -4207396   -4141603   -4140833   -4140833
-4141860   -4141860   -4207396   -4075810   -4140833   -4075040
```

Рис. 9. Фрагмент початкового зображення

```
-16056164   -2297837    -15067338   -13361709   -7653040   -13766677
-14153740   -5548310    -13579969   -2342897    -4738912   -13461124
-3569513    -3646628    -9480201    -11760728   -7021085   -897643
-7064671    -13988266   -9855316    -11481083   -3983554   -4872665
-5931359    -1318154    -12877869   -3423922    -9075996   -9673922
-7923552    -9902616    -8548116    -16617881   -2209510   -1944432
-3825148    -3778995    -14236582   -195220     -10264851  -1304354
-6366077    -10792703   -11283858   -10049522   -9575885   -3591965
-8313304    -2592583    -2793186    -127967     -15627448  -3180121
-15221428   -2306473    -5439553    -5021955    -8044724   -4149800
```

Рис. 10. Фрагмент зашифрованого зображення

-4143656	-4077863	-4077350	-4077350	-4142630	-4142630
-4208936	-4143143	-4142630	-4142630	-4142630	-4076837
-4208423	-4208423	-4142630	-4142630	-4141860	-4141860
-4207653	-4273446	-4273446	-4207653	-4207396	-4141603
-4207653	-4207653	-4273189	-4207396	-4141603	-4075810
-4207908	-4207908	-4207396	-4141603	-4075810	-4010017
-4207908	-4142115	-4076067	-4010274	-4075810	-4010017
-4142115	-4076322	-4010274	-3944481	-4010017	-4010017
-4207653	-4207653	-4207396	-4141603	-4140833	-4140833
-4141860	-4141860	-4207396	-4075810	-4140833	-4075040

Рис. 11. Фрагмент дешифрованого зображення

Висновок

Порівнявши рис. 4 і рис. 7, бачимо, що шифрування чорно-білих і кольорових зображень візуально майже не відрізняється. Контури в обох зашифрованих зображеннях відсутні. Вказаний алгоритм можна використати при передаванні графічних зображень. Запропоновані модифікації можна використати стосовно будь-якого типу зображень, але найбільші переваги досягаються у разі використання зображень, які дають змогу чітко виділяти контури.

Для збільшення швидкодії можна здійснювати шифрування вектора з n елементів (за блоками пікселів, оскільки поелементний доступ до файла займає більше машинного часу).

1. Шнайер Б. Прикладная криптография. – М.: Триумф, 2003. – 815 с. 2. Яне Б. Цифровая обработка изображений. – М.: Техносфера, 2007. – 583 с. 3. Рашкевич Ю.М., Пелешко Д.Д., Ковальчук А.М., Пелешко М.З. Модифікація алгоритму RSA для деяких класів зображень // Технічні вісті 2008/1(27), 2(28). – С. 59–62. 4. Rashkevych Y., Kovalchuk A., Peleshko D., Kupchak M. Stream Modification of RSA Algorithm For Image Coding with precize contour extraction. Proceedings of the X-th International Conference CADSM, 2009. 24–28 February 2009, Lviv–Polyana, Ukraine. – P. 469–473.