УДК 681.3

ШИФРУВАННЯ ЗОБРАЖЕНЬ ЗА ДОПОМОГОЮ ГЕНЕТИЧНОГО АЛГОРИТМУ

П. М. Носко[1, a], О. А. Орєхов[1]

[1] *Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Фізико-технічний інститут*

Анотація

Стаття присвячена створенню нового методу шифрування зображень за допомогою еволюційних моделей, а саме – генетичного алгоритму. На довільній виборці було протестовано його шифрувальну здатність. Результати випробувань довели його спроможність надійно та швидко шифрувати зображення.

Ключові слова: шифрування, генетичний алгоритм, зображення

Вступ

У світі інформаційних технологій підтримання високого рівня безпеки інформації супроводжується великими проблемами. Для комфортної, а головне безпечної передачі даних потрібно дотримуватись багатьох параметрів: прийнятною швидкістю шифрування та розшифрування, цілісністю при їх модифікації, підтримуванням достатнього рівню захисту, використанням методів стискання для зберігання інформації тощо. Більша частина незручностей виникає при роботі із графічними зображеннями, відео, анімаціями. Цей вид зберігання даних є у всіх сферах нашого життя: від спілкування в Інтернеті та застосування у медицині [1], до секретних матеріалів воєнного спрямування та справ державної важливості. За останній час було чимало дискредитованих алгоритмів, тож розробка більш ефективних методів кодування зображень не зупиняється. Більшою мірою, методи шифрування засновані на використанні потужностей системи та не підійдуть для більшості користувачів, які хочуть максимальний рівень захисту, але мають низьку пропускну спроможність. Тож потрібно шукати якісь альтернативи.

1. Огляд генетичного алгоритму

Ще здавна люди шукали можливість передавати конфіденційні дані щоб ті не потрапили до чужих рук. З розвитком науки методи шифрування даних тільки вдосконалювались [2]. Від простих перестановок ми еволюціонували до асиметричної та квантової криптографії [3]. Ще у XX столітті математики та біологи задалися питанням, чи можна математично змоделювати процес життєвого циклу, щоб застосовувати його на прикладних задачах. Це і було початком створення генетичного алгоритму [4]. Щоб вирішити якусь проблему за допомогою ГА завдання формалізується таким чином, щоб її рішення могло бути закодовано у вигляді вектора («генотипу») генів, де кожен ген може бути бітом, числом або якимось іншим об'єктом. Потім, випадково створюється безліч генотипів початкової популяції. Спочатку вони оцінюються з використанням «функції пристосованості», в результаті чого з кожним генотипом асоціюється деяке значення. Це значення можна вважати мірою пристосованості, яке визначає наскільки добре фенотип, їм описуваний, вирішує поставлене завдання. З отриманого безлічі рішень, «поколінь» з урахуванням значення «пристосованості» вибираються рішення, до яких застосовуються «генетичні оператори» («схрещування» або «мутація»), результатом чого є отримання нових рішень. Для них також обчислюється значення пристосованості, і потім проводиться добір («селекція») кращих рішень в наступне покоління. Так повторюється ітеративно, тим самим моделюється «еволюційний процес», що триває кілька життєвих циклів (поколінь), поки не буде виконано умову припинення дії алгоритму: знаходження рішення, вичерпання числа поколінь, часу тощо. Все як у житті, але за допомогою нього ще можна вирішувати багато прикладних задач на оптимізацію, ігрові стратегії, використовувати у біоінформатиці та тренувати штучні нейроні мережі. Ніхто б і не міг подумати, але й такий алгоритм можна використовувати у повсякденному житті, а саме для шифрування інформації, в тому числі і зображень [5], [6].

2. Запропонований метод шифрування

Для шифрування зображення проводимо наступні кроки

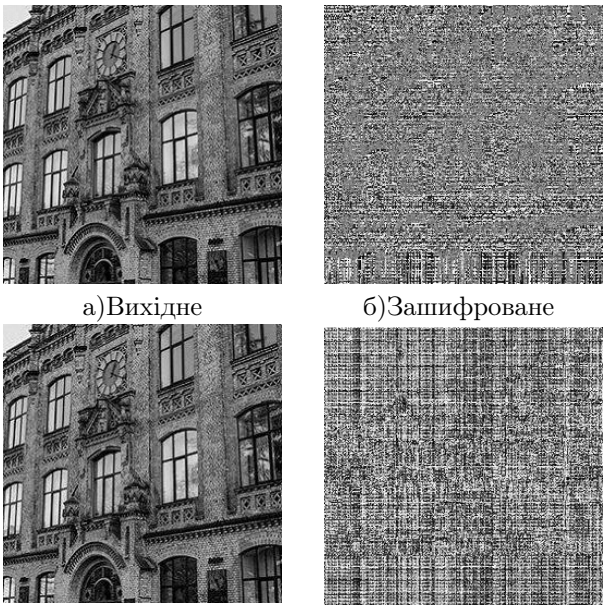
1) Завантажити зображення.
2) Визначити його висоту та ширину.
3) Перевірити отримані данні. Якщо вони не кратні 8 – округлити до більшого значення.

[a] adamovsskin@gmail.com

4) Розділити усе зображення на блоки розміром 8×8.
5) Для кожного блоку зі своєю довжиною та шириною робимо наступні дії.
   - Проводимо схрещування у 3 етапи:
     - Беремо будь-який вертикальний та горизонтальний рядки.
     - Схрещуємо їх випадковим чином.
     - Змінюємо між собою частини рядків. Секретний ключ використовується для кросовера. У цьому випадку, секретний ключ має два атрибути, що називаються a, b та лежать у проміжку від 1 до 8. Кросовер включає в себе зміну a на b у кожному векторі.
   - До кожного елементу векторів проводимо мутацію шляхом впливу новою секретною змінною.
   - Будуємо зашифрований блок з набору N векторів, який утворюється в результаті мутації.
6) Беремо наступний зашифрований блок.
7) Повторюємо пункти 4 та 5 для усіх частин.
8) На виході отримуємо зашифроване зображення.

Кроки для дешифрування зображення:

1) Завантажуємо зашифроване зображення.
2) Також розбиваємо його на блоки.
3) Проводимо мутацію.
4) Проводимо схрещування.
5) Отримуємо дешифрований блок.
6) Робимо так з кожним блоком і на виході отримуємо дешифроване зображення.



а)Вихідне б)Зашифроване

в)Розшифроване з вірним ключем г)Розшифроване з хибним ключем

Рис. 1. Демонстрація роботи алгоритму.

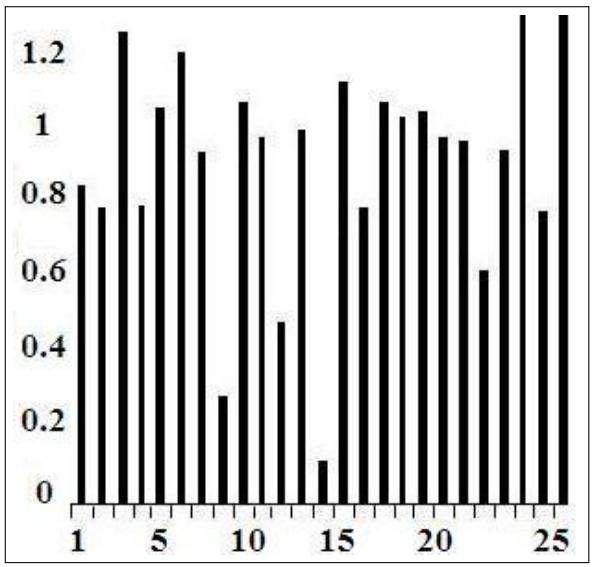


Рис. 2. Індекс SSIM(%) для оригінального та розшифрованого зображення.

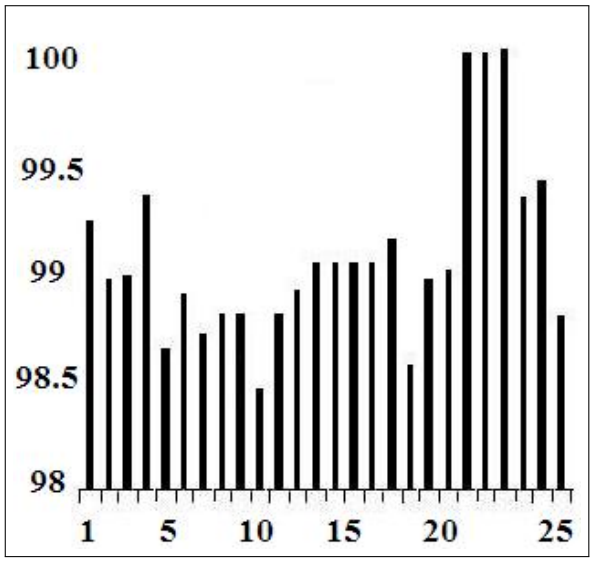


Рис. 3. Індекс SSIM(%) для оригінального та зашифрованого зображення.

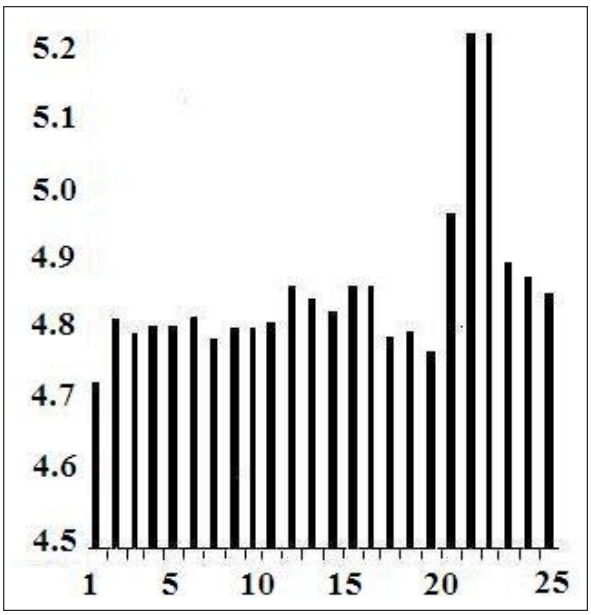
3. Аналіз результатів роботи алгоритму

Для тестування спроможності шифрування будемо використовувати датасет із 25-ти зображень. Усі вони мають різні ключові характеристики.

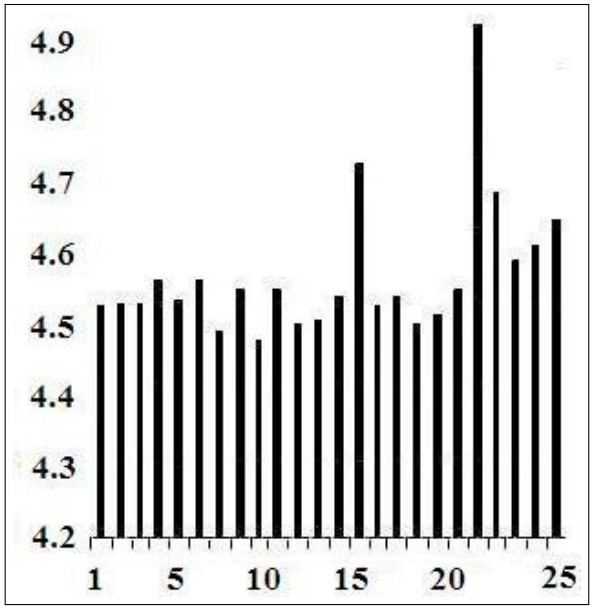
Результати шифрування та розшифрування можна побачити на Рис. 1.

Для оцінки ефективності кодування зображення будемо використовувати індекс структурної схожості — SSIM (structure similarity). Особливість цього методу, полягає в тому, що він враховує «можливість помилки» завдяки врахуванню структурної зміни інформації.[7].

SSIM для розшифрованого зображення (Рис. 3) доходить майже до 100%,а для зашифрованого (Рис.2) знаходиться в межах 0 – 2%. Також, для будь-якого алгоритму кодування важливим показником доцільності його використання є його швидкодія. (Рис.4)



(а) Шифрування



(б) Розшифрування

Рис. 4. Час роботи алгоритму.

Висновки

У цій статті було розглянуто метод шифрування зображень на основі генетичного алгоритму. Він був протестован на вибірці зображень різного формату, розміру та розширення. Запропонований метод ефективно справляється із поставленою задачою. Час шифрування збільшується пропорційно розміру зображення, але все одно залишається незначним і підходить для будь-яких систем. Шифрування відбувається із мінімальним рівнем індексу цифрової ідентичності. Показники цілосності даних після маніпуляцій із зображеннями майже 100%. Алгоритм працює навіть на слабких системах та повністю підходить для використання у повсякденному житті.

Перелік використаних джерел

1. Puech W. Image Encryption and Compression for Medical Security // International Workshops on Image Processing Theory. — 2009. — Vol. 1, no. 30.
2. Brief History of Encryption / P. Dwiti, R. N. Khushboo, T. Sneha et al. // International Journal of Computer Applications. — 2015. — Vol. 131, no. 10.
3. Seema S. K., Chitra G. D. Quantum Cryptography: A Review // Indian Journal of Science and Technology. — 2012. — Vol. 10, no. 3.
4. Genetic Algorithm: A Tutorial Review / M. M. Deep, O. B. Maricel, F. A. Alisherov et al. // International Journal of Computer Science and Mobile Computing. — 2009. — Vol. 2, no. 3.
5. Afarin R., Mozaffari S. Image encryption using genetic algorithm // MVIP and IEEE. — 2013.
6. Rakhi C., Pawanesh A. Genetic Algorithm Based Image Cryptography to Enhance Security // International Journal of Advanced Research in Computer Engineering and Technology. — 2017. — Vol. 6, no. 6.
7. Yusra A., Soong D. Chen. Comparison of image quality assessment: PSNR, HVS, SSIM, UIQI // International Journal of Scientific and Engineering Research. — 2012. — Vol. 3, no. 8.