КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 4 (32), 2026
ISSN 2663 – 4023
DOI 10.28925/2663-4023.2026.32.1105
УДК 004.45

Гришанович Тетяна Олександрівна
кандидат фізико-математичних наук, доцент
Волинський національний університет імені Лесі Українки, Луцьк, Україна
ORCID: 0000-0002-3595-6964
Hryshanovych.Tatiana@vnu.edu.ua

Буткевич Богдан Олександрович
здобувач освіти
Волинський національний університет імені Лесі Українки, Луцьк, Україна
ORCID: 0009-0009-6562-6745
butkevychb@gmail.com

КІЛЬКІСНА ОЦІНКА ЯКОСТІ ТА СТІЙКОСТІ ОКРЕМИХ СТЕГАНОГРАФІЧНИХ АЛГОРИТМІВ

Анотація. У роботі наведено результати порівняльного аналізу трьох поширених стеганографічних підходів, LSB (Least Significant Bit), DCT (Discrete Cosine Transform) і PVD (Pixel Value Differencing), що застосовуються для вбудовування текстових повідомлень у цифрові зображення, з різними параметрами розміру та роздільної здатності. Основною метою дослідження є виявлення сильних і слабких сторін кожного методу шляхом використання кількісних показників, які забезпечують об'єктивне оцінювання рівня якості та стійкості стеганографічних алгоритмів. Для проведення аналізу було обрано чотири метрики: середньоквадратичну похибку (MSE), пікове співвідношення сигнал/шум (PSNR), індекс структурної подібності (SSIM) та нормалізовану взаємну кореляцію (NCC). Зазначені показники дозволяють кількісно охарактеризувати ступінь спотворення зображень, а також рівень збереження їх структурних властивостей після процесу приховування інформації. Експериментальні результати засвідчили, що алгоритм LSB забезпечує найвищі показники якості зображень і практично не помітні візуальні відхилення. Водночас метод DCT характеризується більш значними спотвореннями, однак може бути доцільним у сценаріях, де допустиме зниження якості. Алгоритм PVD продемонстрував найменший час виконання та забезпечив збалансоване поєднання якості, стійкості й продуктивності, що свідчить про його перспективність для систем, орієнтованих на високу швидкість обробки даних. Проведене дослідження підтвердило ефективність застосування сукупності кількісних метрик для всебічного порівняння стеганографічних методів. Запропонований підхід дає змогу більш глибоко проаналізувати поведінку алгоритмів у різних умовах та формує підгрунтя для створення адаптивних стеганографічних рішень, спрямованих на підвищення рівня інформаційної безпеки.

Ключові слова: алгоритми; стеганографія; методи стеганографії; кількісні метрики; оцінка.

ВСТУП

Стеганографічні системи зазвичай оцінюються за трьома базовими якісними характеристиками: ємністю, непомітністю та стійкістю [1, 2]. Ємність визначає обсяг інформації, який може бути вбудований у контейнер без критичного погіршення його властивостей, і залежить як від обраного алгоритму приховування, так і від параметрів самого носія. Для кількісного опису цієї характеристики застосовуються різні показники, зокрема кількість бітів прихованого повідомлення, що припадає на один піксель зображення, або відношення розміру вбудованих даних до максимально допустимого обсягу для конкретного контейнера [4].

Непомітність є ключовим критерієм оцінювання якості стеганографічного методу, оскільки вона відображає ступінь відмінності між стегоконтейнером і вихідним зображенням. Аналіз цієї характеристики ґрунтується на використанні сенсорних і статистичних показників, які дозволяють кількісно оцінити рівень візуальних і структурних змін, що виникають у процесі вбудовування інформації.

Стійкість характеризує здатність стеганографічного алгоритму зберігати приховані дані після різних видів обробки контейнера, таких як стиснення, фільтрація, додавання шуму, зміна розміру, геометричні перетворення або конвертація в інші формати. Окрім цього, дана властивість передбачає протидію методам виявлення та несанкціонованого вилучення інформації, зокрема статистичним і криптографічним атакам. Зазначені параметри перебувають у взаємному протиріччі, а їх баланс визначається специфікою практичного застосування: збільшення ємності зазвичай супроводжується зростанням рівня спотворень контейнера. Отже, вибір оптимальних характеристик стеганографічної системи має здійснюватися з урахуванням поставлених вимог і пріоритетів.

Серед найбільш поширених методів приховування інформації в цифрових зображеннях виділяють алгоритми LSB (Least Significant Bit), DCT (Discrete Cosine Transform) і PVD (Pixel Value Differencing), які відрізняються механізмами вбудовування даних та рівнем стійкості до подальшої обробки зображень. Аналіз ефективності зазначених підходів є актуальним науковим завданням, спрямованим на підвищення надійності та прихованості передавання інформації в графічних файлах.

Мета статті. Метою даного дослідження є вивчення результативності стеганографічних алгоритмів LSB, DCT і PVD із використанням сукупності кількісних показників. Оцінювання ефективності зазначених підходів має принципове значення для встановлення їхньої практичної доцільності в умовах реального застосування. Обґрунтований вибір стеганографічного методу повинен враховувати не лише здатність вбудовувати заданий обсяг інформації, але й рівень збереження візуальних характеристик зображення, стійкість до змін формату, складність виявлення прихованого сигналу та надійність передавання даних.

Використання об’єктивних кількісних метрик, зокрема MSE (Mean Squared Error), PSNR (Peak Signal-to-Noise Ratio), SSIM (Structural Similarity Index Measure) і NCC (Normalized Cross-Correlation), забезпечує отримання точних і відтворюваних результатів, що створює підґрунтя для коректного порівняння різних алгоритмів. Такий підхід є особливо важливим під час розроблення систем інформаційної безпеки, у яких вибір стеганографічного механізму безпосередньо впливає на рівень захищеності критично важливих даних. Аналіз зазначених методів здійснюється на основі результатів їх застосування до зображень сформованої вибірки.

Аналіз останніх досліджень і публікацій. Метод найменш значущого біта (Least Significant Bit, LSB) ґрунтується на модифікації найменш значущих бітів значень пікселів цифрового зображення, що зумовлює мінімальний вплив на його візуальні характеристики. [8] Це пояснюється тим, що молодші біти мають незначний внесок у формування інтенсивності яскравості та кольору, а отже їх зміна, як правило, є непомітною для системи зорового сприйняття людини.

Нехай контейнер-зображення описується матрицею

$$I = \{p_{i,j}\}, p_{i,j} \in \{0, 1, \dots 255\},$$

де $p_{i,j}$ – значення пікселя у позиції (i, j)(для кожного кольорового каналу окремо).

Секретне повідомлення подається у вигляді бітової послідовності

$$M = \{m_k\},\ m_k \in \{0,1\}.$$

Процес вбудовування інформації полягає у заміні найменш значущого біта кожного вибраного пікселя відповідним бітом повідомлення і може бути формалізований таким чином:

$$p'_{i,j} = \left(p_{i,j}\&\overline{1}\right)|m_k,$$

де $p'_{i,j}$ – модифіковане значення пікселя,
& – побітова операція AND,
$\overline{1}$ – маска для занулення молодшого біта,
| – побітова операція OR.

Алгоритм приховування даних методом LSB складається з таких етапів:

1. Зчитування цифрового зображення-контейнера.
2. Перетворення секретного повідомлення у двійкову послідовність.
3. Визначення найменш значущих бітів пікселів для кожного кольорового каналу.
4. Послідовна заміна цих бітів на біти прихованого повідомлення.

Вилучення інформації здійснюється шляхом зчитування стегоконтейнера та екстракції молодших бітів пікселів:

$$m_k = p'_{i,j}\&1,$$

після чого отримана бітова послідовність декодується у початкове текстове повідомлення.

Метод дискретного косинусного перетворення (Discrete Cosine Transform, DCT) базується на поданні зображення у частотній області шляхом розкладання просторового сигналу на сукупність спектральних коефіцієнтів. [9] Отримані коефіцієнти характеризують внесок окремих частотних компонент у формування зображення, що створює передумови для приховування інформації з урахуванням особливостей зорового сприйняття людини.

Для двовимірного цифрового зображення $f(x,y)$ дискретне косинусне перетворення обчислюється для кожної двовимірної просторової області – блока розміром $N \times N$ пікселів (як правило, $N = 8$). Значення DCT-коефіцієнтів визначається за формулою:

$$C(u,v) = \alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left(\frac{(2x+1)u\pi}{2N}\right)\cos\left(\frac{(2y+1)v\pi}{2N}\right),$$

де $C(u,v)$ – коефіцієнт перетворення для частот (u,v),
$\alpha(u), \alpha(v)$ – нормувальні коефіцієнти.

У спектральному представленні зображення розрізняють низькочастотні, середньочастотні та високочастотні компоненти. Низькочастотна область містить основну енергію сигналу та визначає ключові візуальні характеристики зображення, тоді як високочастотні складові є більш чутливими до втрат під час стиснення та

впливу шумів. З огляду на це, вбудовування секретної інформації здійснюється шляхом модифікації коефіцієнтів середньочастотного піддіапазону, що забезпечує компроміс між непомітністю та стійкістю методу.

Після завершення процесу вбудовування застосовується зворотне дискретне косинусне перетворення (Inverse Discrete Cosine Transform, IDCT), яке переводить модифіковані частотні коефіцієнти назад у просторову область [3].

Алгоритм приховування даних із використанням методу DCT реалізується у такій послідовності:

1. Зчитування зображення-контейнера.
2. Подання секретного повідомлення у вигляді двійкової послідовності.
3. Розбиття зображення на блоки розміром 8×8 пікселів та зміщення значень пікселів шляхом віднімання 128 для центрування діапазону ([-128;127]).
4. Застосування DCT до кожного блока.
5. Квантування отриманих коефіцієнтів із використанням таблиці квантування.
6. Заміна найменш значущих бітів вибраних DC-коефіцієнтів відповідними бітами секретного повідомлення.
7. Виконання зворотного DCT з метою відновлення зображення у просторовій області.

Вилучення прихованої інформації здійснюється шляхом виконання перших етапів алгоритму, ідентичних процедурі вбудовування, з подальшою екстракцією найменш значущих бітів DC-коефіцієнтів. Отримана бітова послідовність декодується у початковий текстовий формат.

Метод різниці значень пікселів (Pixel Value Differencing, PVD) ґрунтується на аналізі локальних змін яскравості зображення та забезпечує високу непомітність стегоконтейнера за рахунок адаптивного вбудовування інформації [10]. Основна ідея полягає у виборі пар суміжних пікселів та визначенні максимально допустимого обсягу прихованих даних залежно від величини різниці їх значень. Такий підхід дозволяє збільшити ємність методу, зберігаючи при цьому візуальні й статистичні характеристики зображення після вбудовування повідомлення.

Цифрове зображення розбивається на неперекривні блоки, кожен з яких складається з двох послідовних пікселів. Для кожної пари обчислюється абсолютна різниця

$$d = |p_1 - p_2|, d \in [0, 255].$$

Невеликі значення d відповідають однорідним (гладким) ділянкам зображення, де допустимий обсяг прихованих даних є обмеженим. Натомість великі значення d характерні для крайових або текстурованих областей, що дозволяє вбудовувати більшу кількість секретних бітів без помітного погіршення якості. Визначення ємності для кожної пари пікселів здійснюється за допомогою таблиці діапазонів квантування, яка містить безперервні інтервали в межах $[0, 255]$. Кількість бітів, що вбудовуються у кожну пару пікселів, визначається відповідним діапазоном цієї таблиці.

Алгоритм приховування даних методом PVD реалізується таким чином:

1. Зчитується зображення-контейнер та розділяється на окремі кольорові канали (RGB); у межах кожного каналу формується послідовність неперекривних пар пікселів із використанням зигзагоподібного сканування, після чого для кожної пари обчислюється значення різниці d.

2. За таблицею квантування для обчисленого d визначається відповідний інтервал $[l, u]$, де l і u – нижня та верхня межі діапазону, а також кількість бітів n, які можуть бути приховані у даному блоці.

3. З бітової послідовності секретного повідомлення зчитується n бітів та перетворюється у десяткове число b.

4. Обчислюється нове значення різниці

$$d' = l + b,$$

де виконується умова $d' \in [l, u]$.

5. Нові значення пікселів p_1' та p_{12}' визначаються таким чином, щоб зберегти середнє значення пари та забезпечити різницю d.

Зазначені кроки повторюються доти, доки вся бітова послідовність секретного повідомлення не буде вбудована у зображення.

Процедура вилучення прихованої інформації передбачає виконання початкових етапів, аналогічних процесу вбудовування. Для кожної пари пікселів у стегозображенні обчислюється різниця d', після чого за таблицею діапазонів квантування визначається відповідний інтервал $[l, u]$. Значення

$$b = d' - l,$$

перетворюється у двійковий формат, а отримані біти послідовно відновлюють початкове секретне повідомлення.

Оцінювання ефективності стеганографічних методів здійснюється із використанням кількісних показників, що обчислюються на основі аналізу піксельних значень зображень. Зазначені метрики дозволяють об’єктивно визначити рівень спотворень, які виникають у контейнері після вбудовування прихованої інформації. У даній роботі для порівняльного аналізу застосовуються найбільш поширені критерії якості [9].

PSNR (Peak Signal-to-Noise Ratio) – пікове відношення сигналу до шуму, яке вимірюється в децибелах і характеризує співвідношення максимально можливої потужності корисного сигналу до потужності шуму, внесеного в процесі модифікації зображення [7]. Даний показник широко використовується для оцінки візуальної стійкості стеганографічних алгоритмів. Великі значення PSNR свідчать про те, що зміни, внесені в контейнер, є малопомітними для людського зору, а якість стегозображення залишається високою.

Значення PSNR визначається на основі середньої квадратичної помилки (MSE) за формулою

$$PSNR = 10 \log_{10} \left(\frac{MAX^2}{MSE} \right)$$

де MAX – максимальне можливе значення пікселя зображення, а MSE – середня квадратична похибка між оригінальним зображенням і стегозображенням. Дана метрика відображає ступінь відхилення піксельних значень після вбудовування повідомлення: чим меншим є значення MSE, тим меншими є спотворення зображення і вищою є його якість.

Для оригінального зображення розміру $W \times H$ середня квадратична помилка

$$MSE = \frac{1}{WH}\sum_{i=1}^{W}\sum_{j=1}^{H}\big(I(i,j) - I'(i,j)\big)^2,$$

де $I(i,j)$ та $I'(i,j)$ – значення пікселів оригінального та стеганографічного зображень, відповідно.

SSIM (Structural Similarity Index Measure) – індекс структурної схожості, що використовується для оцінювання подібності між стегоконтейнером і початковим зображенням з урахуванням особливостей людського зорового сприйняття [19].

Значення SSIM знаходиться в діапазоні від -1 до 1, де значення близьку до 1 відповідає майже повній ідентичності зображень [11]. Метрика SSIM базується на аналізі трьох складових: яскравості, контрасту та структури.

Яскравість (luminance) визначається як середнє значення пікселів області зображення:

$$\mu = \frac{1}{N}\sum_{k=1}^{N} x_k,$$

де x_k – значення k-го пікселя області, а N – загальна кількість пікселів.

Контраст (contrast) описується середньоквадратичним відхиленням піксельних значень та обчислюється за формулою

$$\sigma = \sqrt{\frac{1}{N-1}\sum_{k=1}^{N}(x_k - \mu)^2},$$

де μ — середнє значення яскравості області.

Функція порівняння яскравості для двох зображень x та y має вигляд

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_2}$$

де $C_1 = 6.5025$ – стабілізаційна константа [8].

Функція порівняння контрасту визначається як

$$c(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2},$$

де $C_2 = 58.5225$.

Функція порівняння структури має вигляд

$$s(x,y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3},$$

де σ_{xy} — коваріація між зображеннями, а $C_3 = \frac{C_2}{2}$.

Загальний індекс структурної схожості визначається формулою

$$SSIM(x,y) = [l(x,y)]^{\alpha}[c(x,y)]^{\beta}[s(x,y)]^{\gamma},$$

де $\alpha > 0, \beta > 0, \gamma > 0$ – коефіцієнти вагомості окремих складових. За умови рівних ваг $\alpha = \beta = \gamma = 1$ отримуємо стандартну форму метрики SSIM [10].

NCC (Normalized Cross-Correlation) – нормована взаємна кореляція, що характеризує ступінь подібності між початковим контейнером і модифікованим зображенням [12]. Значення NCC, близьке до 1, вказує на високу кореляцію між зображеннями та незначні спотворення, спричинені вбудовуванням даних.

Нормована взаємна кореляція визначається за формулою

$$NCC = \frac{\sum_{i,j}(I(i,j)-\bar{I})(I'(i,j)-\bar{I}')}{\sqrt{\sum_{i,j}(I(i,j)-\bar{I})^2}\sqrt{\sum_{i,j}(I'(i,j)-\bar{I}')^2}},$$

де $I(i,j)$ та $I'(i,j)$ – значення пікселів оригінального та стегозображення відповідно, а $\bar{I}$ і $\bar{I}'$ – їх середні значення.

Окрім наведених показників, у літературі існують і інші кількісні критерії оцінювання якості та ефективності стеганографічних алгоритмів [5], [7], однак у межах даної роботи.

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для виконання порівняльного аналізу стеганографічних методів було створено програмний інструмент StegoProg, функціональні можливості якого орієнтовані на дослідження ефективності вбудовування прихованих даних у цифрові зображення.

Розроблений програмний засіб забезпечує реалізацію таких основних функцій:

- вбудовування секретної інформації з високою пропускною здатністю за умови збереження належної візуальної якості зображення;
- зниження рівня візуальних спотворень та небажаних артефактів, що можуть виникати внаслідок процесу приховування даних;
- підтримку роботи з різними форматами графічних файлів, зокрема JPEG, PNG, BMP та іншими поширеними типами зображень;
- забезпечення визначеного рівня стійкості стегоконтейнера до випадкових або ненавмисних модифікацій.

Реалізація програмного засобу виконана мовою програмування Python з використанням спеціалізованих бібліотек Pillow, OpenCV, NumPy та Ctkinter, які забезпечують обробку зображень, чисельні обчислення та створення графічного інтерфейсу користувача [6], [16], [17], [18].

Далі наведено реалізації алгоритмів приховування даних на основі методів LSB, DCT та PVD.

Насамперед розглянемо програмну реалізацію методу LSB, призначеного для вбудовування текстової інформації у графічні зображення.

```
# Визначення кількості каналів (3 для RGB, 4 для RGBA)
channels = 4 if image.mode == "RGBA" else 3
pixels = img_arr.size // channels
# Перетворення повідомлення у бінарний формат
byte_message = ''.join(f"{ord(c):08b}" for c in message_to_hide)
bits = len(byte_message)
# Перевірка, чи повідомлення не занадто довге для приховування
    if bits > pixels:
        result_label.configure(text="The message is too long to encode!")
    else:
        index = 0
        for i in range(pixels):
            for j in range(0, 3):
            if index < bits:
        img_arr[i][j] = int(bin(img_arr[i][j])[2:-1] + byte_message[index], 2)
        index += 1
# Перетворення масиву назад у зображення
    img_arr = img_arr.reshape((height, width, channels))
    result = PIL.Image.fromarray(img_arr.astype('uint8'), image.mode)
    dct_encoded_image_file_path = "./imgs/encoded/lsb_" + original_image_file
```

Реалізація методу DCT:

```
# DCT, квантування і вставка секретного повідомлення для кожного каналу (R, G, B)
for channel in range(3):
    for i in range(0, height, N):
        for j in range(0, width, N):
            block = img_arr[i:i + N, j:j + N, channel]
            dct_block = dct(block)
            quantized_block = np.round(dct_block / QUANTIZATION_MATRIX)
    if index < len(bit_message):
    quantized_block[4, 4] += -(int(quantized_block[4, 4]) % 2) +
int(bit_message[index]) # зміни коефіцієнтів піддіапазону середніх частот
index += 1
img_arr[i:i + N, j:j + N, channel] = quantized_block * QUANTIZATION_MATRIX
img_arr[i:i + N, j:j + N, channel] = idct(img_arr[i:i + N, j:j + N, channel])
img_arr += 128
img_arr = np.clip(img_arr, 0, 255)
encoded_img = np.uint8(img_arr)
encoded_img = Image.fromarray(encoded_img, 'RGB')
dct_encoded_image_file_path = "./imgs/encoded/dct_" + original_image_file
encoded_img.save(dct_encoded_image_file_path)
```

Реалізація методу PVD:

```
# Повідомлення конвертується у двійковий формат. Потім до нього додаються нулі
на початку, щоб довжина була кратною 8.
data = '0' * (8 - len(data) % 8) + data #Додається  довжина повідомлення (у 32-
бітовому двійковому форматі) на початок, щоб можна було витягти його при
розшифровці.
data_len = bin(len(data))[2:].zfill(32)
data = data_len + data
i = capacity = 0
while i < height:
    for j in range(0, width, 2):
    for k in range(3):
        print("img max = ", max(img[i, j + 1, k], img[i, j, k]))
        print("img min = ", min(img[i, j + 1, k], img[i, j, k]))
        print("img = ", img[i, j])
        print("img + 1 = ", img[i, j+1])
        dif = max(img[i, j + 1, k], img[i, j, k]) - min(img[i, j + 1, k], img[i, j, k])
        print("dif = ", dif )
        emb, n, maxr = embed_number(dif)
        print("emb, n, maxr - ", emb, " ", n," ", maxr)
#перевіряється, чи можна змінити різницю між пікселями на максимальну
допустиму.
res, _, _ = change_diff(maxr - dif, min(img[i, j + 1, k], img[i, j, k]), max(img[i, j + 1, k],
img[i, j, k]))
print("change_dif = ", res)
if not res:
    continue
#Зчитуються біти з повідомлення і додаються до різниці, після чого знову
використовується change_diff для корекції значень пікселів
    bits = data[capacity:capacity + n]
    capacity += len(bits)
#Зчитуємо біти секретного повідомлення і перетворюємо в десяткове значення
    new_dif = emb + int(bits, 2)
#Рахуємо нову різницю
    _, img[i, j, k], img[i, j + 1, k] = change_diff(new_dif - dif, img[i, j, k], img[i, j + 1, k])
    print("new pixels values - ", img[i, j, k], " ", img[i, j + 1, k])
    print("----------------------------------")
    if capacity == len(data): #Якщо всі біти повідомлення вбудовані, цикл
завершується
```

Застосування кількісних показників, дає змогу об'єктивно визначити ступінь візуальної непомітності змін, що вносяться у зображення в процесі приховування інформації, що є одним із ключових критеріїв для стеганографічних систем. Проведення експериментального тестування дозволяє виявити наявні недоліки реалізованих алгоритмів, удосконалити їх та підвищити загальну надійність і якість програмного продукту, забезпечуючи його коректне функціонування в умовах практичного використання.

Для виконання експериментів було сформовано тестовий набір графічних зображень із роздільною здатністю 512×512 та 1024×1024 пікселів. Як секретні повідомлення використовувався текст, згенерований за допомогою сервісу Lorem Ipsum [13], обсягом 100 байт і 10 000 байт.

Після етапу підготовки та завантаження відповідних наборів зображень і текстових даних тестування здійснювалося за наступною схемою: у кожне зображення різного розміру вбудовувалися повідомлення різної довжини та змісту із застосуванням різних стеганографічних алгоритмів. Загалом було виконано чотири серії експериментів, у межах кожної з яких проводилися наступні дії.

Для кожного зображення здійснювалося приховування секретного повідомлення заданого розміру з використанням реалізованих методів LSB, DCT та PVD. Оцінювання впливу процесу вбудовування інформації на якість зображення виконувалося на основі метрик. Додатково проводився аналіз ступеня подібності між оригінальним та модифікованим зображеннями за показниками SSIM (Structural Similarity Index Measure) і NCC (Normalized Cross-Correlation).

Результати проведених експериментів подані у відповідних таблицях нижче та відображають ефективність трьох розглянутих методів стеганографії при вбудовуванні текстових повідомлень різного обсягу у зображення з різною просторовою роздільною здатністю.

Method	File Name	Text Size	Image Resolution	MSE	PSNR	Embedding Time	Extraction Time	SSIM	NCC
LSB	flowers.png	100byte	512px	0.000565847	80.60381527	0.363	0.568	0.999941984	0.999999956
LSB	house.png	100byte	512px	0.000574748	80.53603103	0.348	0.616	0.999996011	0.999999921
LSB	lenna.png	100byte	512px	0.000536601	80.83429087	0.34	0.505	0.999996011	0.999999921
LSB	motorbike.png	100byte	512px	0.00055186	80.72253641	0.409	0.587	0.999998917	0.999999959
LSB	pillar.png	100byte	512px	0.000550588	80.72253641	0.512	0.649	0.999999995	0.999999859
DCT	flowers.png	100byte	512px	39.36485545	32.179717	0.347	0.173	0.911309368	0.992959424
DCT	house.png	100byte	512px	28.77057648	33.54131797	0.323	0.198	0.69014166	0.996491594
DCT	lenna.png	100byte	512px	20.4274203	35.02866836	0.304	0.203	0.665944524	0.996365964
DCT	motorbike.png	100byte	512px	17.24373118	35.76449117	0.33	0.295	0.397755611	0.998088456
DCT	pillar.png	100byte	512px	12.24219894	37.25220928	0.284	0.68	0.321593651	0.996330874
PVD	flowers.png	100byte	512px	0.003037771	73.30525388	0.06	0.068	0.999924442	0.99999996
PVD	house.png	100byte	512px	0.000507355	81.07768642	0.077	0.101	0.999995733	0.999999893
PVD	lenna.png	100byte	512px	0.000740051	79.43818553	0.032	0.073	0.999998503	0.999999917
PVD	motorbike.png	100byte	512px	0.001140594	77.55949094	0.021	0.061	0.999999995	0.999999879
PVD	pillar.png	100byte	512px	0.00048701	81.25542764	0.037	0.054	0.999989339	0.999999983

Рис 1. Тестування 1. Набір зображень розміром 512x512 пікселів і текст розміром 100 байт

Method	File Name	Text Size	Image Resolution	MSE	PSNR	Embedding Time	Extraction Time	SSIM	NCC
LSB	butterfly.png	100byte	1024px	0.000146866	86.46159553	1.283	2.458	0.999991979	0.999999982
LSB	flowers.png	100byte	1024px	0.000137011	86.76324259	1.626	2.66	0.999999953	0.999999982
LSB	house.png	100byte	1024px	0.00014623	86.48043697	1.378	2.1	0.99998878	0.999999989
LSB	motorbike.png	100byte	1024px	0.000137011	86.76324259	1.314	1.94	0.999999873	0.999999989
LSB	pillar.png	100byte	1024px	0.000129382	87.0120712	1.24	2.55	0.999962271	0.999999968
DCT	butterfly.png	100byte	1024px	25.24402396	34.10921777	1.181	0.735	0.505038639	0.998444288
DCT	flowers.png	100byte	1024px	7.537981351	39.35825302	1.15	0.7	0.90044616	0.996441004
DCT	house.png	100byte	1024px	23.9254233	34.34220731	1.161	0.763	0.667432267	0.997358346
DCT	motorbike.png	100byte	1024px	18.630874	35.42847132	1.161	0.68	0.410741555	0.998106587
DCT	pillar.png	100byte	1024px	20.63357417	34.98505897	1.21	0.695	0.522605866	0.994722043
PVD	butterfly.png	100byte	1024px	0.000139236	86.69327418	0.049	0.044	0.999999926	0.999999952
PVD	flowers.png	100byte	1024px	0.000130335	86.98017672	0.046	0.039	0.99998218	0.99999999
PVD	house.png	100byte	1024px	0.000372887	82.41503517	0.073	0.054	0.999999835	0.999999982
PVD	motorbike.png	100byte	1024px	0.0002683	83.84459082	0.062	0.055	0.999978088	0.999999969
PVD	pillar.png	100byte	1024px	0.000122388	87.25340799	0.049	0.096	0.999804706	0.999969695

Рис 2. Тестування 2. Набір зображень розміром 1024x1024 пікселів і текст розміром 100 байт

Method	File Name	Text Size	Image Resolution	MSE	PSNR	Embedding Time	Extraction Time	SSIM	NCC
LSB	flowers.png	10000byte	512px	0.051183065	61.03954074	0.438	0.564	0.999937812	0.999993579
LSB	house.png	10000byte	512px	0.05132548	61.0274734	0.515	0.653	0.977641046	0.99999612
LSB	lenna.png	10000byte	512px	0.050991058	61.05586335	0.414	0.748	0.998105111	0.999992675
LSB	motorbike.png	10000byte	512px	0.050912221	61.06258315	0.522	0.539	0.992334326	0.999996197
LSB	pillar.png	10000byte	512px	0.050977071	61.05705481	0.438	0.548	0.997191366	0.999987251
DCT	flowers.png	10000byte	512px	39.97412872	32.11301354	0.431	0.207	0.853069194	0.98907782
DCT	house.png	10000byte	512px	29.50245285	33.43222236	0.382	0.295	0.564422945	0.99424037
DCT	lenna.png	10000byte	512px	21.35538101	34.83573037	0.36	0.22	0.531130748	0.991884579
DCT	motorbike.png	10000byte	512px	18.12255478	35.5486094	0.357	0.223	0.314177419	0.995782575
DCT	pillar.png	10000byte	512px	13.23407491	36.91386772	0.372	0.301	0.185147635	0.988600307
PVD	flowers.png	10000byte	512px	0.24180603	54.29613234	0.879	0.787	0.999804706	0.999969695
PVD	house.png	10000byte	512px	0.14181623	58.43222236	0.382	0.456	0.990902732	0.999995628
PVD	lenna.png	10000byte	512px	0.166381042	61.23523304	0.67	0.588	0.999495628	0.999495622
PVD	motorbike.png	10000byte	512px	0.061421712	60.24758442	1.681	0.9	0.990902732	0.999995628
PVD	pillar.png	10000byte	512px	0.122554779	56.51860339	0.357	0.312	0.990902732	0.999995628

Рис 3. Тестування 3. Набір зображень розміром 512x512 пікселів і текст розміром 10000 байт

Method	File Name	Text Size	Image Resolution	MSE	PSNR	Embedding Time	Extraction Time	SSIM	NCC
LSB	butterfly.png	10000byte	1024px	0.012884458	67.0301421	1.514	2.251	0.996605896	0.999998402
LSB	flowers.png	10000byte	1024px	0.012705167	67.09099978	1.526	2.339	0.99997782	0.999998409
LSB	house.png	10000byte	1024px	0.012810071	67.05528814	1.474	2.209	0.992775122	0.999999036
LSB	motorbike.png	10000byte	1024px	0.012752851	67.0747308	1.441	1.999	0.999730996	0.999999053
LSB	pillar.png	10000byte	1024px	0.012678464	67.1001371	1.375	2.173	0.999624259	0.999996872
DCT	butterfly.png	10000byte	1024px	8.543396632	40.12234106	1.524	0.747	0.255644128	0.994340158
DCT	flowers.png	10000byte	1024px	34.18298492	38.81449792	1.351	0.737	0.785060176	0.992358984
DCT	house.png	10000byte	1024px	24.81886228	34.18298492	1.391	0.746	0.528318162	0.995052606
DCT	motorbike.png	10000byte	1024px	33.97496353	35.21948806	1.586	0.695	0.318771425	0.99568295
DCT	pillar.png	10000byte	1024px	21.65505377	34.77521094	1.312	0.773	0.350719888	0.986645683
PVD	butterfly.png	10000byte	1024px	0.012839953	63.92496451	1.972	1.778	0.996697009	0.999998403
PVD	flowers.png	10000byte	1024px	0.038779895	67.04516923	1.079	0.931	0.999953221	0.999995188
PVD	house.png	10000byte	1024px	0.032581212	62.24473737	1.011	1.01	0.999739292	0.999998541
PVD	motorbike.png	10000byte	1024px	0.019740105	65.1773091	1.495	1.335	0.999739292	0.999998541
PVD	pillar.png	10000byte	1024px	0.051053725	64.67511014	1.122	0.897	0.999953221	0.999995188

Рис 4. Тестування 4. Набір зображень розміром 1024x1024 пікселів і текст розміром 10000 байт

Проведений експериментальний аналіз трьох стеганографічних підходів (LSB, DCT та PVD) дав змогу сформулювати низку узагальнень щодо їхньої ефективності та впливу на якість зображень під час приховування текстових даних. Зокрема, метод LSB (Least Significant Bit) показав найкращі показники збереження візуальної якості після вбудовування інформації. Для нього характерні низькі значення середньоквадратичної помилки (MSE) та високі значення відношення сигнал/шум (PSNR), що вказує на практично непомітні для користувача зміни у зображенні. Метрики SSIM і NCC також підтверджують мінімальні відмінності за яскравістю, контрастом і структурою між оригінальним та стегозображенням. Таким чином, LSB доцільно застосовувати в задачах, де пріоритетом є збереження максимальної якості зображень. Водночас цей метод характеризується підвищеними часовими витратами на вбудовування та вилучення інформації, особливо у випадку зображень великої роздільної здатності.

Метод DCT, навпаки, супроводжується більш вираженими спотвореннями зображень, що проявляється у зростанні значень MSE та зниженні PSNR. Менші значення індексу SSIM порівняно з іншими методами свідчать про зменшення візуальної подібності між початковим і модифікованим зображеннями, тобто про помітнішу втрату якості після приховування інформації. Разом з тим DCT може бути прийнятним у ситуаціях, де допустимий певний рівень деградації якості. За часовими характеристиками цей метод займає проміжне положення: він працює швидше за LSB, але поступається за швидкодією методу PVD.

Метод PVD забезпечує прийнятний баланс між якістю та швидкістю обробки, демонструючи добрі результати збереження візуальних характеристик, особливо для зображень з високою роздільною здатністю. Достатньо високі значення PSNR, а також майже ідеальні показники SSIM і NCC свідчать про незначний вплив на структуру та сприйняття зображення. Окрім цього, PVD є найшвидшим серед розглянутих методів,

оскільки час вбудовування та вилучення даних для нього є мінімальним, що робить цей підхід привабливим для задач, чутливих до продуктивності.

Отже, кожен із проаналізованих методів стеганографії має власні сильні та слабкі сторони, які визначають доцільність їх використання в конкретних умовах. LSB забезпечує найвищу якість зображень ціною збільшення часу обробки, DCT може застосовуватися за умови допустимості втрати якості, тоді як PVD поєднує високу швидкодію з прийнятним рівнем візуальної якості. Остаточний вибір методу визначається вимогами до якості зображення та обмеженнями за часом обробки.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

У ході дослідження було проведено порівняльний аналіз трьох поширених методів стеганографії – LSB (Least Significant Bit), DCT (Discrete Cosine Transform) та PVD (Pixel Value Differencing) – на основі кількісних метрик MSE, PSNR, SSIM та NCC. Отримані результати дозволили об’єктивно оцінити ефективність кожного методу з точки зору якості зображень після вбудовування інформації, часу виконання та стійкості до змін.

Результати тестування показали:

- метод LSB забезпечує найвищу якість зображень, що підтверджується низькими значеннями MSE та високими значеннями PSNR, SSIM та NCC. Це робить його придатним для застосувань, де критично важливим є збереження візуальної якості контейнера;
- метод DCT демонструє найбільші спотворення, що знижує його придатність для завдань, де важлива непомітність, але може бути корисним у сценаріях, коли допустима втрата якості;
- метод PVD поєднує високу швидкість вбудовування з прийнятною якістю та стійкістю, що робить його ефективним для задач, орієнтованих на продуктивність.

Порівняння показало, що жоден із методів не є універсальним: вибір алгоритму має залежати від конкретних вимог до ємності, якості зображення, швидкості та рівня захисту. Метрики MSE, PSNR, SSIM і NCC виявилися ефективними інструментами кількісного оцінювання якості та стійкості методів стеганографії, що дозволяє забезпечити відтворюваність результатів і коректне порівняння алгоритмів.

Перспективи подальших досліджень передбачають розширення аналізу за рахунок інших стеганографічних методів (наприклад, на основі хвильових перетворень або комбінованих гібридних моделей); дослідження впливу попередньої обробки зображень (фільтрація, стиснення, масштабування) на стійкість алгоритмів; розробку адаптивних методів вибору алгоритму залежно від типу зображення та вимог до ємності й непомітності; застосування методів машинного навчання для автоматизації процесу вибору оптимального способу вбудовування інформації. Отримані результати можуть бути використані для побудови більш ефективних систем прихованого обміну даними та підвищення рівня захисту інформації у цифровому середовищі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Horpeniuk, A., & Storozhenko, A. (2012). Research and comparative analysis of steganographic methods for embedding data in digital files. *Bulletin of Lviv Polytechnic National University*, (741), 2–4.
2. Denysiuk, V. (2017). Steganographic algorithm for data protection using image files. *Efektyvna Ekonomika*, (5). http://www.economy.nayka.com.ua/?op=1&z=5584

3. Klymenko, L. A., & Hordiienko, A. Y. (2019). Analysis of image compression methods based on discrete cosine transform. *Information and Control Systems on Railway Transport*, (5), 24–31.
4. Shvidchenko, I. (2012). Methods for detecting steganographic information hiding in images. *Bulletin of Lviv Polytechnic National University*, (741), 1–6.
5. Zeng, Q., et al. (2025). A method combining discrete cosine transform with attention for multi-temporal remote sensing image matching. *Sensors, 25*(5), 1345. https://doi.org/10.3390/s25051345
6. Python Software Foundation. (n.d.). *Graphical user interfaces with Tk (tkinter).* https://docs.python.org/3/library/tkinter.html
7. Helmrich, C., Bosse, S., Schwarz, H., Marpe, D., & Wiegand, T. (2020). A study of the extended perceptually weighted peak signal-to-noise ratio (XPSNR) for video compression with different resolutions and bit depths. *ITU Journal: ICT Discoveries, 3*(1).
8. Hidayasari, N., & Yanto, F. (2020). Analysis of least significant bit method using sequential encoding-decoding in steganography digital image. In *Proceedings of the International Conference on Science and Engineering* (Vol. 3, pp. 201–205). https://doi.org/10.14421/icse.v3.498
9. Wahab, O. F. A., et al. (2019). Hiding data in images using steganography techniques with compression algorithms. *TELKOMNIKA (Telecommunication Computing Electronics and Control), 17*(3), 1168.
10. Alade, O. M., et al. (2021). Image steganography using pixel value differencing (PVD) technique based on firefly algorithm. *Journal of Scientific Research and Reports*, 80–86. https://doi.org/10.9734/jsrr/2021/v27i730414
11. Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing, 13*(4), 600–612.
12. Luo, J., & Konofagou, E. E. (2010). A fast normalized cross-correlation calculation method for motion estimation. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, 57*(6), 1347–1357. https://doi.org/10.1109/TUFFC.2010.1554
13. Lorem Ipsum. (n.d.). *Lorem Ipsum generator.* https://www.lipsum.com/
14. Mara, D. S. (2022). Computer forensics in image steganography. *International Journal for Research in Applied Science and Engineering Technology, 10*(5), 3831–3842.
15. NumPy. (n.d.). *NumPy package.* https://pypi.org/project/numpy/
16. OpenCV. (n.d.). *opencv-python package.* https://pypi.org/project/opencv-python/
17. Pillow. (n.d.). *Pillow package.* https://pypi.org/project/pillow/
18. Tseng, H.-W., & Leng, H.-S. (2013). A steganographic method based on pixel-value differencing and the perfect square number. *Journal of Applied Mathematics*, 2013, 1–8.
19. Milovic, C., et al. (2024). XSIM: A structural similarity index measure optimized for MRI QSM. *Magnetic Resonance in Medicine.* https://doi.org/10.1002/mrm.30271

КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 4 (32), 2026
ISSN 2663 – 4023

Tetiana Hryshanovych
Ph.D.,Associate Professor, Head of the Department of Computer Science and Cybersecurity
Lesya Ukrainka Volyn National University, Lutsk, Ukraine
ORCID: 0000-0002-3595-6964
Hryshanovych.Tatiana@vnu.edu.ua

Bohdan Butkevych
student
Lesya Ukrainka Volyn National University, Lutsk, Ukraine
ORCID: 0009-0009-6562-6745
butkevychb@gmail.com

QUANTITATIVE ASSESSMENT OF THE QUALITY AND ROBUSTNESS OF SELECTED STEGANOGRAPHIC ALGORITHMS

Abstract. This paper presents the results of a comparative analysis of three widely used steganographic approaches – LSB (Least Significant Bit), DCT (Discrete Cosine Transform), and PVD (Pixel Value Differencing) – applied to embedding textual messages into digital images with varying sizes and resolutions. The main objective of the study is to identify the strengths and weaknesses of each method through the use of quantitative metrics that provide an objective evaluation of the quality and robustness of steganographic algorithms.

Four metrics were selected for the analysis: Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), and Normalized Cross-Correlation (NCC). These indicators make it possible to quantitatively assess the degree of image distortion as well as the preservation of structural properties after the information-hiding process.

Experimental results demonstrate that the LSB algorithm achieves the highest image quality with virtually imperceptible visual distortions. In contrast, the DCT method exhibits more noticeable degradation, although it may be suitable in scenarios where a reduction in image quality is acceptable. The PVD algorithm showed the shortest execution time and provided a balanced combination of quality, robustness, and performance, indicating its potential suitability for systems focused on high-speed data processing.

The conducted study confirms the effectiveness of using a set of quantitative metrics for comprehensive comparison of steganographic methods. The proposed approach enables a deeper analysis of algorithm behavior under various conditions and forms a foundation for the development of adaptive steganographic solutions aimed at enhancing information security.

Keywords: Algorithms; steganography; steganographic methods; quantitative metrics; evaluation.

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. Horpeniuk, A., & Storozhenko, A. (2012). Research and comparative analysis of steganographic methods for embedding data in digital files. *Bulletin of Lviv Polytechnic National University*, (741), 2–4.
2. Denysiuk, V. (2017). Steganographic algorithm for data protection using image files. *Efektyvna Ekonomika*, (5). http://www.economy.nayka.com.ua/?op=1&z=5584
3. Klymenko, L. A., & Hordiienko, A. Y. (2019). Analysis of image compression methods based on discrete cosine transform. *Information and Control Systems on Railway Transport*, (5), 24–31.
4. Shvidchenko, I. (2012). Methods for detecting steganographic information hiding in images. *Bulletin of Lviv Polytechnic National University*, (741), 1–6.
5. Zeng, Q., et al. (2025). A method combining discrete cosine transform with attention for multi-temporal remote sensing image matching. *Sensors, 25*(5), 1345. https://doi.org/10.3390/s25051345
6. Python Software Foundation. (n.d.). *Graphical user interfaces with Tk (tkinter)*. https://docs.python.org/3/library/tkinter.html
7. Helmrich, C., Bosse, S., Schwarz, H., Marpe, D., & Wiegand, T. (2020). A study of the extended perceptually weighted peak signal-to-noise ratio (XPSNR) for video compression with different resolutions and bit depths. *ITU Journal: ICT Discoveries, 3*(1).

8. Hidayasari, N., & Yanto, F. (2020). Analysis of least significant bit method using sequential encoding-decoding in steganography digital image. In *Proceedings of the International Conference on Science and Engineering* (Vol. 3, pp. 201–205). https://doi.org/10.14421/icse.v3.498
9. Wahab, O. F. A., et al. (2019). Hiding data in images using steganography techniques with compression algorithms. *TELKOMNIKA (Telecommunication Computing Electronics and Control), 17*(3), 1168.
10. Alade, O. M., et al. (2021). Image steganography using pixel value differencing (PVD) technique based on firefly algorithm. *Journal of Scientific Research and Reports*, 80–86. https://doi.org/10.9734/jsrr/2021/v27i730414
11. Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing, 13*(4), 600–612.
12. Luo, J., & Konofagou, E. E. (2010). A fast normalized cross-correlation calculation method for motion estimation. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, 57*(6), 1347–1357. https://doi.org/10.1109/TUFFC.2010.1554
13. Lorem Ipsum. (n.d.). *Lorem Ipsum generator.* https://www.lipsum.com/
14. Mara, D. S. (2022). Computer forensics in image steganography. *International Journal for Research in Applied Science and Engineering Technology*, *10*(5), 3831–3842.
15. NumPy. (n.d.). *NumPy package.* https://pypi.org/project/numpy/
16. OpenCV. (n.d.). *opencv-python package.* https://pypi.org/project/opencv-python/
17. Pillow. (n.d.). *Pillow package.* https://pypi.org/project/pillow/
18. Tseng, H.-W., & Leng, H.-S. (2013). A steganographic method based on pixel-value differencing and the perfect square number. *Journal of Applied Mathematics*, 2013, 1–8.
19. Milovic, C., et al. (2024). XSIM: A structural similarity index measure optimized for MRI QSM. *Magnetic Resonance in Medicine.* https://doi.org/10.1002/mrm.30271

Отримано редакцією журналу / Received: 14.01.26
Прорецензовано / Revised: 02.02.26
Схвалено до друку / Accepted: 26.03.26