DOI: 10.18372/2410-7840.22.14801
УДК 004.056; 004.415.24

ДОСЛІДЖЕННЯ ОСНОВНИХ КОМПОНЕНТІВ СИСТЕМ JPEG-СТЕГАНОАНАЛІЗУ НА БАЗІ МАШИННОГО НАВЧАННЯ

Наталія Кошкіна

Для побудови ефективних стеганоаналітичних систем у заданих практичних умовах необхідно здійснити аналіз та оцінку якості існуючих методів та компонентів. Для вибору оптимальних складових системи необхідно порівняти оцінки базових характеристик наявних кандидатів. Проте здійснити таке порівняння, базуючись на даних з наукових публікацій, досить складно через відмінності в умовах чисельних експериментів. В основі даного дослідження лежить принцип створення рівних умов для всіх досліджуваних статистичних моделей формування характеристичних векторів для стеганоаналізу JPEG-зображень за методами на базі машинного навчання. Проаналізовано швидкодію та точність детектування чотирьох різних варіантів приховування даних у частотній області, що були отримані з використанням таких статистичних моделей як CHEN, CC-CHEN, LIU, CC-PEV, CC-C300, GFR та DCTR, а також SVM з лінійним чи гаусівським ядром або ансамблевого класифікатора. Основними результатами здійсненого дослідження є таблиці, що відображають чисельні оцінки швидкодії основних етапів стеганоаналізу та точності класифікації пустих і заповнених контейнерів.

Ключові слова: *інформаційна безпека, стеганоаналіз, пасивна протидія, методи з навчанням та класифікацією, порівняльний аналіз, моделі характеристичних векторів, SVM, ансамблевий класифікатор.*

Вступ. Однією з актуальних проблем інформаційної безпеки є боротьба з незаконною прихованою передачею інформації. Дисципліна про методи протидії стеганографічному приховуванню отримала назву стеганоаналіз. Найпершою задачею стеганоаналізу є визначення того, містить досліджуваний об'єкт приховану інформацію чи ні. Подальші етапи базуються на достовірності припущення про застосування стеганографії. Сам процес стеганоаналізу досить складний, щоб в повній мірі застосовувати його до усіх наявних об'єктів, тому правильне визначення класу об'єкту (пустий чи заповнений) дозволяє суттєво скоротити час та обчислювальні ресурси, потрібні для експертизи різних можливих об'єктів, зокрема цифрових аудіосигналів, зображень, відеороликів.

Останнім часом значну увагу дослідників отримали стеганоаналітичні методи з навчанням і класифікацією, так як вони є універсальними та можуть покращуватися шляхом застосування новітніх здобутків теорії машинного навчання. Загальна схема таких методів відображена на рис. 1.

Ці методи кожному об'єкту (контейнеру) ставлять у відповідність його характеристичний вектор, що чутливо реагує на стеганоперетворення і разом з тим не є залежним від вмісту контейнера. Стеганодетектором виступає класифікатор, на вхід якого подаються характеристичні вектори. Він будується шляхом контрольованого навчання на контейнерах, для який відома мітка класу – «пустий» чи «заповнений». Таким чином, двома ключовими компонентами методів з навчанням і класифікацією є модель характеристичних векторів та метод класифікації.

У наукових публікаціях пропонується широкий вибір як різних моделей характеристичних векторів, так і класифікаторів, придатних для вирішення задач стеганоаналізу (частина існуючих варіантів описана, наприклад, у роботі [1]). Але зробити найкращий вибір, зокрема скомпонувати модель характеристичних векторів та метод класифікації так, щоб отримати найбільш точну та прийнятно швидку стеганоаналітичну систему, спираючись тільки на результати наявні у статтях дуже важко, зокрема через розбіжності у проведенні чисельних експериментів. Тому, **метою роботи** є практичне дослідження в рівних умовах показників якості різних існуючих моделей векторів та класифікаторів для стеганоаналізу найпоширеніших стеганографічних контейнерів – JPEG-зображень.

Ці методи кожному об'єкту (контейнеру) ставлять у відповідність його характеристичний вектор, що чутливо реагує на стеганоперетворення і разом з тим не є залежним від вмісту контейнера. Стеганодетектором виступає класифікатор, на вхід якого подаються характеристичні вектори. Він будується шляхом контрольованого навчання на контейнерах, для який відома мітка класу – «пустий» чи «заповнений». Таким чином, двома ключовими компонентами методів з навчанням і класифікацією є модель характеристичних векторів та метод класифікації.

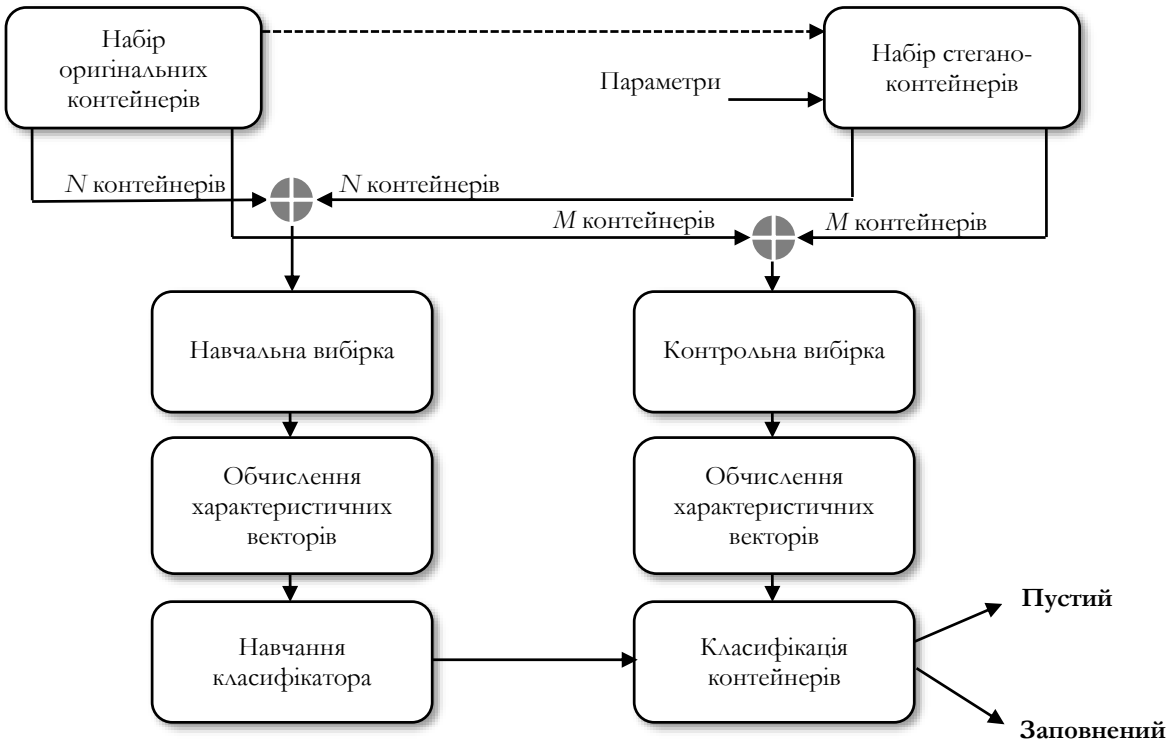

Рис. 1. Загальна схема стеганоаналізу на базі машинного навчання

У наукових публікаціях пропонується широкий вибір як різних моделей характеристичних векторів, так і класифікаторів, придатних для вирішення задач стеганоаналізу (частина існуючих варіантів описана, наприклад, у роботі [1]). Але зробити найкращий вибір, зокрема скомпонувати модель характеристичних векторів та метод класифікації так, щоб отримати найбільш точну та прийнятно швидку стеганоаналітичну систему, спираючись тільки на результати наявні у статтях дуже важко, зокрема через розбіжності у проведенні чисельних експериментів. Тому, **метою роботи** є практичне дослідження в рівних умовах показників якості різних існуючих моделей векторів та класифікаторів для стеганоаналізу найпоширеніших стеганографічних контейнерів – JPEG-зображень.

Вибір стеганографічних програм чи перетворень, яким потрібно протидіяти. Аналіз стеганоконтейнерів. Як вихідний набір було обрано 1330 кольорових JPEG-зображень розмірами 384×512 пікселів, стиснених з коефіцієнтом якості 75 (розміри файлів у наборі – від 8 до 82 кБайт). Стеганоконтейнери для дослідження створювалися за допомогою трьох програмних продуктів, що реалізують НЗБ приховування в частотній області – Jsteg, Jphide, Steganos Privacy Suite 2012 (модуль Crypt&Hide) та Матлаб реалізації алгоритму нового покоління – J-UNIWARD. У кожен вихідний контейнер вкраплявся 1 кБ даних. Разом з тим кількість змінених ДКП коефіцієнтів помітно відрізняється в залежності від алгоритму приховування, що відображено на рис. 2.

Так, найбільших змін зазнали стеганоконтейнери, створені програмою Jsteg – 5398±456 коефіцієнтів. Зазначимо, що на відміну від інших досліджуваних варіантів Jsteg здійснює не тільки приховування, а й стиснення, тому змінені коефіцієнти в цілому відповідають як прихованим бітам, так і артефактам стиснення. Зокрема, на рис. 3б, в верхній частині зазначені місцеположення змінених ДКП коефіцієнтів і за рахунок приховування, і за рахунок стиснення, а в нижній частині можна побачити виключно зміни, спричинені похибками заокруглення алгоритму стиснення Jsteg (програма реалізує послідовне НЗБ вкраплення).

Середня кількість ДКП коефіцієнтів, змінених Jphide – 3150±60. Відносно мале середньоквадратичне відхилення пояснюється тим, що для визначення порядку зміни ДКП коефіцієнтів Jphide за допомогою фіксованої таблиці ділить всі придатні коефіцієнти на класи за порядком і НЗБ приховування продовжується у поточному класі навіть після вкраплення всього повідомлення. Перший клас у таблиці складають DC-коефіцієнти (нульова частота), далі йдуть класи AC-коефіцієнтів (всі інші частоти), абсолютні значення яких більші за відповідні табличні величини. В результаті біти повідомлення будуть розподілені по всьому контейнеру та міститимуться в найбільших за модулем ДКП коефіцієнтах (див. рис. 3в).

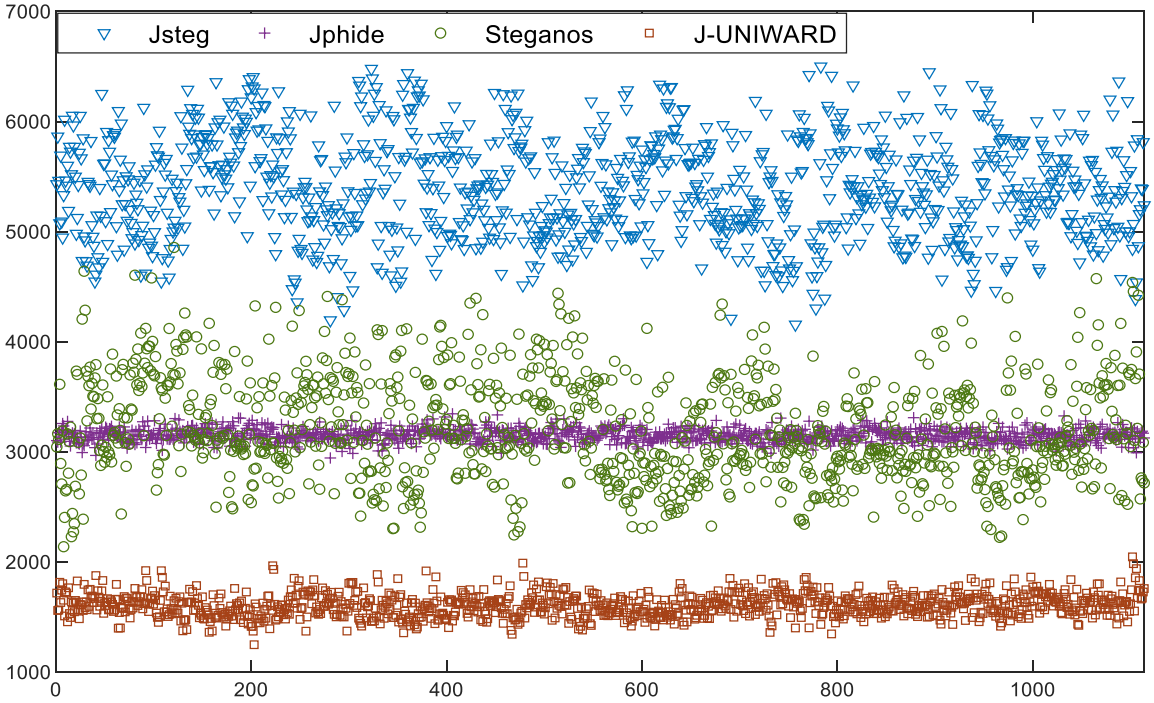

Рис. 2. Кількість змінених ДКП коефіцієнтів для зображень тестового набору, після вкраплення в них 1 кБ даних за допомогою Jsteg, Jphide, Steganos та J-UNIWAR

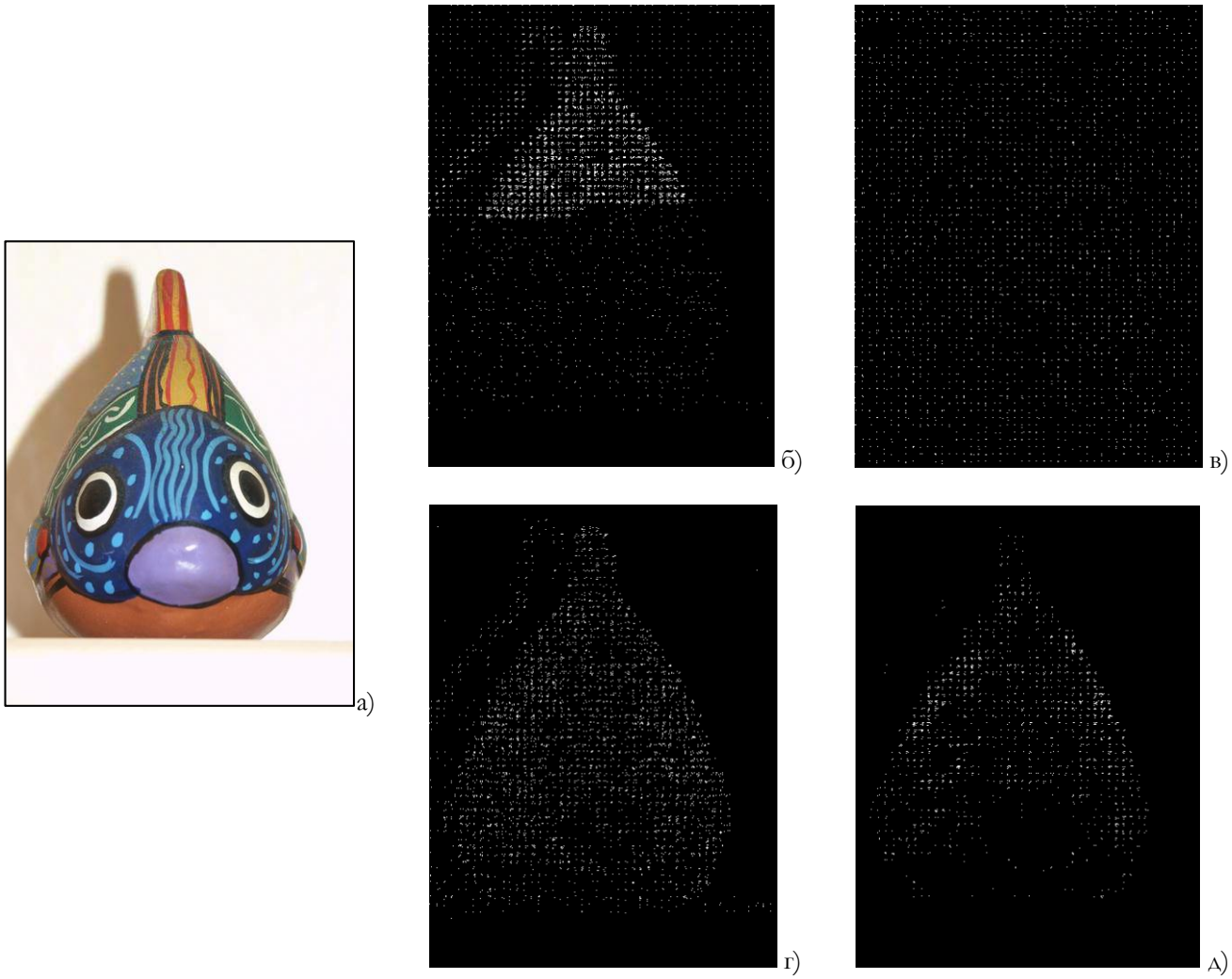

Рис. 3. Місцеположення змінених ДКП коефіцієнтів в одному з тестових зображень (а) після вкраплення в нього 1 кБ даних за допомогою Jsteg (б), Jphide (в), Steganos (г) та J-UNIWARD (д)

Програма Steganos Privacy Suite, а саме компонент Crypt&Hide для JPEG-контейнерів реалізує послідовне НЗБ вкраплення в квантовані ДКП коефіцієнти за виключенням статичної компоненти кожного блоку (DC). Цей крок направлений на збільшення візуальної непомітності стеганографічного втручання. Середня кількість ДКП коефіцієнтів, змінених Steganos – 3205±468. Уявлення про місцеположення змінених коефіцієнтів можна отримати з рис. 3г. Зважаючи на те, що загальна кількість ДКП коефіцієнтів в досліджуваних зображеннях рівна 196608, а кількість DC-коефіцієнтів – 3072, можна прорахувати середній відсоток змін коефіцієнтів області приховування. Для Steganos він складає 1.66%, в той час як для Jsteg 2.75%. Тож можна зробити висновок, що на артефакти стиснення для Jsteg стеганоконтейнерів припадає близько 1% змін ДКП коефіцієнтів.

Алгоритму нового покоління J-UNIWARD притаманна найменша кількість змін: 607±105. Такі цифри пояснюються в першу чергу застосуванням під час стеганоперетворення синдромних гратчастих кодів (Syndrome Trellis Codes), запропонованих як спосіб мінімізації впливу вкраплення у роботі [2]. Крім того, на відміну від попередніх варіантів UNIWARD здійснює адаптивне вкраплення, коли біти повідомлення з більшою вірогідністю будуть приховані у тих частинах зображення, де їх складніше виявити, тобто в першу чергу в текстурованих та високошумних (див. рис. 3д).

Відмітимо, що для подальших експериментів у кожному з наявних наборів було залишено по 1114 зображень – саме скільки виявилося таких, для яких з пустого контейнера всіма обраними стеганопрограмами було створено заповнені (Jphide та Steganos не створюють стеганоконтейнер при завеликому для нього повідомленні).

Опис моделей формування характеристичних векторів. Для дослідження було обрано сім існуючих статистичних моделей формування характеристичних векторів для JPEG-контейнерів, які коротко описані нижче.

1. CHEN – модель, заснована на процесах Маркова, яка використовує внутрішньоблокові та міжблокові кореляції між ДКП коефіцієнтами природних зображень. Запропонована у роботі [3]. Для визначення внутрішньоблокових кореляцій в моделі генеруються чотири різницеві матриці: горизонтальна $F_h = F(u,v) - F(u+1,v)$, вертикальна $F_\varpi = F(u,v) - F(u+1,v)$, головна діагональна $F_d = F(u,v) - F(u+1,v+1)$ та побічна діагональна $F_m = F(u+1,v) - F(u,v+1)$, де $F(u,v)$ – абсолютні значення ДКП коефіцієнтів зображення. Потім для кожної різницевої обчислюється матриця ймовірностей переходу. Наприклад, для горизонтальної різницевої матриці вона буде мати вигляд

$$M_h(i,j) = \frac{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v}\lambda(F_h(u,v)=i, F_h(u+1,v)=j}{\sum_{u=1}^{S_u-1}\sum_{v=1}^{S_v}\lambda(F_h(u,v)=i},$$

де $\lambda = 1$, якщо його аргументи виконуються і $\lambda = 0$ в протилежному випадку, S_u та S_v – розміри зображення. Міжблокові кореляції відображають залежності між ДКП коефіцієнтами, які розташовані в одній і тій же позиції в сусідніх блоках. Для аналізу міжблокових порушень для відповідних горизонтальної і вертикальної різницевих матриць обчислюються також матриці ймовірностей переходу. Діагональні матриці автори ігнорують, бо вони суттєво не впливають на результат. Щоб зменшити розмірність результуючого вектору для всіх матриць ймовірностей переходу використовується усікання до діапазону [-4, +4], що з кожної різницевої матриці дає по 81 елементу для характеристичного вектора. Таким чином, 81×4 елементи фіксують внутрішньоблокові порушення після стеганоперетворення, 81×2 – міжблокові.

2. CC-CHEN – модель CHEN, покращена декартовим калібруванням. Взагалі ідея калібрування JPEG-зображення полягає в десинхронізації блоків 8×8, всередину яких відбувається приховання даних. Як правило, для калібрування застосовують обрізування на 4 пікселі в горизонтальному та вертикальному напрямках від початку, але з цією ж метою може бути використане і масштабування або обертання на невеликий кут. Результатом калібрування деякого зображення F є опорне зображення F_r. Так як калібрування руйнує приховані дані, опорне зображення можна розглядати як наближення пустого контейнера. Тому спершу в стеганоаналізі виникла ідея розглядати калібровані характеристичні вектори, як такі, що отримані за значеннями різниці між вихідним та каліброваним зображенням $F_{cal} = F_r - F$. Але в роботі [4] показано, що статистика опорного зображення не завжди близька до статистики оригінального і аналіз різниць відповідно не завжди приводить до покращення точності детектування. Крім того в деяких випадках може бути погіршення у порівнянні з некаліброваним варіантом, бо при калібруванні відніманням може втрачатися корисна інформація. Як більш ефективну альтернативу в [4] запропоновано використовувати декартове калібрування, коли $F_{cal} = [F_r, F]$. Таким чином, CC-CHEN – це

модель, в якій аналізується шість матриць ймовірностей переходу для досліджуваного зображення і таких же шість матриць – для його каліброваної (шляхом обрізування) версії.

3. LIU – модель, яка запропонована у роботі [5] та базується на тому, що стеганографічні приховання змінюють спільну щільність сусідніх елементів. Як і попередні, дана модель аналізує внутрішньоблокову та міжблокову статистику значень квантованих ДКП коефіцієнтів зображення. Матриці щільності сусідніх внутрішньоблокових з'єднань в горизонтальному та вертикальному напрямках визначаються як:

$$absNJ_{1h}(x,y)=\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{n=1}^{7}\lambda\left(\left|c_{ijmn}\right|=x,\left|c_{ijm(n+1)}\right|=y\right)}{56MN},$$

$$absNJ_{1v}(x,y)=\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{n=1}^{8}\lambda\left(\left|c_{ijmn}\right|=x,\left|c_{ij(m+1)n}\right|=y\right)}{56MN},$$

де c_{ijmn} – коефіцієнт, розташований у m рядку та n стовпчику блоку $M\times N$ ДКП коефіцієнтів. Для оптимізації обчислень в подальшому аналізується усереднена матриця з двох даних. Аналогічні міжблокові матриці мають вигляд

$$absNJ_{2h}(x,y)=\frac{\sum_{m=1}^{8}\sum_{n=1}^{8}\sum_{i=1}^{M}\sum_{j=1}^{N-1}\lambda\left(\left|c_{ijmn}\right|=x,\left|c_{i(j+1)mn}\right|=y\right)}{64M(N-1)}$$ та

$$absNJ_{2v}(x,y)=\frac{\sum_{m=1}^{8}\sum_{n=1}^{8}\sum_{i=1}^{M-1}\sum_{j=1}^{N}\lambda\left(\left|c_{ijmn}\right|=x,\left|c_{(i+1)jmn}\right|=y\right)}{64(M-1)N}$$ і

також усереднюються. Так як спільна щільність варіюється для різних зображень, щоб відобразити її зміну, спричинену стеганоперетворенням, застосовується калібрування зображення. Потім обчислюються середні значення матриць щільності каліброваної версії та розраховуються диференціальні характеристики між отриманими статистиками. Автори обмежили розгляд цілочисельних параметрів x та y діапазоном [0, 5], що на виході дало 144 елементи для значень різниць матриць щільності вихідного та каліброваного зображення *ref-diff-absNJ* та 72 елементи для значень часток *diff-absNJ-ratio*.

4. CC-PEV – модель PEV, покращена декартовим калібруванням, запропонованим в [4]. Вихідна модель PEV представлена у роботі [6]. Характеристичний вектор в цій моделі має 274 елементи, 193 з них сформовані на базі ДКП, а 81 – елемент, отримано на основі процесів Маркова. Набір статистичних даних на базі ДКП в свою чергу включає 11 елементів глобальної гістограми H розподілу коефіцієнтів ДКП; по 11 елементів для п'яти локальних гістограм h_l^{ij} розподілу значень перших п'яти АС-коефіцієнтів; по 9 елементів для 11-ти дуальних гістограм g_{ij}^{d}; 1 елемент усередненої варіації V значень коефіцієнтів ДКП у суміжних блоках розбиття зображення; 2 елементи блочності (скаляри, обчислені з декомпресованого зображення, що представляють інтегральну міру міжблокової залежності); 25 елементів матриці спільної появи значень коефіцієнтів ДКП, квантованих у діапазоні [-2, +2] в блоках розбиття зображення. 81 елемент на основі процесів Маркова – це усереднені значення чотирьох внутрішньоблокових матриць ймовірності переходу, розрахованих за алгоритмом моделі CHEN. Для підвищення точності детектування в моделі CC-PEV розраховуються всі вищезгадані елементи як для вихідного зображення, так і для його каліброваної версії.

5. CC-C300 – модель, що грунтується на використанні високорозмірних векторів, з метою охопити найбільше залежностей між коефіцієнтами ДКП. Запропонована у роботі [7]. Характеристичні вектори складають елементи матриць спільної появи пар коефіцієнтів ДКП, де пара визначається згідно формулі

$$P\left(\Delta i,\Delta j,k_1,l_1,k_2,l_2\right)=$$
$$\left\{\left[D_{k_1l_1}^{(i,j)},D_{k_2l_2}^{(i+\Delta i,j+\Delta j)}\right]\middle|\, i=1,\ldots N^{(i)},j=1,\ldots N^{(j)}\right\}\cdot$$

Тут $D_{kl}^{(i,j)}$ – (k, l)-тий коефіцієнт в (i, j)-тому блоці ДКП; $k_1,l_1,k_2,l_2\in\{0,\ldots 7\}$; $\Delta i,\Delta j$ – натуральні числа; $N^{(i)}=8\lceil M/8\rceil-\Delta i$, $N^{(j)}=8\lceil N/8\rceil-\Delta j$; $M\times N$ – розміри зображення. При чому вихідні коефіцієнти ДКП усікаються до діапазону $[-T, T]$ та повинна справджуватися нерівність $|\Delta i|+|\Delta j|+|k_1-k_2|+|l_1-l_2|>0$. Матриці спільної появи мають вигляд:

$$C_{st}=\frac{1}{N^{(i)}N^{(j)}}\sum_{i=1}^{N^{(i)}}\sum_{j=1}^{N^{(j)}}\left\{\begin{array}{c}[a,b]\in\\ P(\Delta i,\Delta j,k_1,l_1,k_2,l_2)\\ |\,a=s,b=t\end{array}\right\}.$$

Автори фіксують $T=4$, що дає 81-елементну матрицю C_{st} для кожної шестірки значень $(\Delta i,\Delta j,k_1,l_1,k_2,l_2)$. Для того щоб зробити побудову окремих матриць більш системною всі можливі пари коефіцієнтів ДКП сортуються за ступенем важливості і матриці потім обчислюються у порядку від найважливіших до найменш важливих пар. Як міра важливості використовується взаємна інформація, обчислена на досить великому наборі випадково вибраних пар коефіцієнтів. Таким чи-

ном, результуючий набір елементів характеристичного вектора об'єднує в собі Ω найважливіших матриць спільної появи, маючи розмірність $\Omega \times 81$. Після декартового калібрування розмірність, яку автори позначають через CC-CΩ, подвоюється до $2\times\Omega\times 81$. В даній роботі використовувався набір CC-C300, тобто такий що об'єднує 300 матриць спільної появи.

6. GFR – модель характеристичних векторів, побудованих як гістограми квантованих залишків, отриманих з використанням двовимірних фільтрів Габора (Gabor Filter Residual). 2D-фільтри Габора описують особливості текстури зображення з позицій різних масштабів та орієнтацій. Модель представлена в [8]. На відміну від попередніх будується в просторовій області. Спочатку зображення декомпресується без заокруглення значень. Генерується двовимірний банк фільтрів Габора, що включає фільтри з двофазним зміщенням $(\phi = 0, \pi)$, чотирма масштабами ($\sigma = 0.5,\ 0.75,\ 1,\ 1.25$) та 32-ма орієнтаціями ($\theta = 0, \pi/32, \ldots 31\pi/32$). Далі розпаковане зображення згортається з 8×8 2D-фільтром Габора $G^{\phi,\sigma,\theta}$, щоб отримати відповідне залишкове зображення $U^{\phi,\sigma,\theta}$. Відповідно до фази $(a,b),\ 0 \le a,b \le 7$ залишки поділяються на 64 підмножини $U_{a,b}^{\phi,\sigma,\theta}$, для кожної з яких обчислюються гістограми

$$h_{a,b}^{\phi,\sigma,\theta}(r) = \frac{1}{\left|U_{a,b}^{\phi,\sigma,\theta}\right|}\sum_{u\in U_{a,b}^{\phi,\sigma,\theta}}\left[Q_T\left(|u|/q\right)=r\right],$$

де Q_T – квантувач з цілочисельними центроїдами $\{0,1,\ldots T\}$, q – крок квантування, а [P] - дужка Іверсона, що дорівнює 0, коли твердження P хибне, і 1, коли P істинне. Завдяки симетрії отримані 64 гістограми $h_{a,b}^{\phi,\sigma,\theta}$ зливаються до 25-ти: разом складаються гістограми, індекси яких $(a,b),(a,8-b),(8-a,b),(8-a,8-b)$, за умови що ці показники залишаються в межах $\{0,1,\ldots 7\}\times\{0,1,\ldots 7\}$. Потім ці 25 гістограм з'єднуються в гістограму $h^{\phi,\sigma,\theta}$ залишку $U^{\phi,\sigma,\theta}$. Гістограми $h^{\phi,\sigma,\pi-\theta}$ та $h^{\phi,\sigma,\theta}$ об'єднуються згідно симетричним орієнтаціям. Наостанок результуючі гістограми стають елементами характеристичного вектора.

7. DCTR – модель характеристичних векторів із ДКП залишків (Discrete Cosine Transform Residual), що враховує фази. Була запропоновані у роботі [9], як і GFR будується в просторовій області. Елементи характеристичних векторів в даній моделі будуються як гістограми залишків, отримані з використанням базових шаблонів ДКП, що мають вигляд $B_{mn}^{(k,l)} = \frac{w_k w_l}{4}\cos\frac{\pi k(2m+1)}{16}\cos\frac{\pi l(2n+1)}{16}$, $w_0 = \frac{1}{\sqrt{2}}$, $w_k = 1 (k>0),\ 0 \le m,n \le 7$. Для створення характеристичних векторів потрібно обчислити 64 згортки розпакованого в просторову область без заокруглення до цілих чисел JPEG зображення з 64 ядрами 8×8 та сформувати нормалізовані гістограми, аналогічні тим, що описані для попередньої моделі. Для подальшої компактизації векторів використовується симетрія шаблонів – гістограми об'єднуються за тим же принципом, що й в моделі GFR. Загальна розмірність симетризованого характеристичного вектора становить $64\times(36/4+24/2+4)\times(T+1) = 1600\times(T+1)$. Як компроміс між швидкодією та точністю автори пропонують використовувати $T=4$.

Опис класифікаторів. Для досліджень було обрано два найбільш популярні з огляду їх згадування у наукових публікаціях бінарні класифікатори – метод опорних векторів (support vector machine, SVM) та ансамблевий класифікатор (ensemble classifier). Коротко опишемо кожен із них.

Нехай маємо навчальну вибірку $(E,X) = \{\vec{\delta}_n, \chi_n\}_{n=1}^N$, де $\vec{\delta}_n$ – деякий об'єкт в просторі R^n, $\chi_n \in \{-1,+1\}$ – його мітка класу. Задача полягає в тому, щоб на основі навчальної вибірки спрогнозувати мітку класу $\hat{\chi}$ для нового об'єкту $\vec{\delta}$. В застосуванні до стеганоаналізу $\vec{\delta}_n$ – характеристичний вектор. Якщо $\vec{\delta}_n$ вилучено з пустого контейнера, то $\chi_n = -1$, якщо з заповненого, то $\chi_n = +1$.

Функція прийняття рішень для лінійної SVM має вигляд $p(\vec{\delta}) = sign(\vec{w}\cdot\vec{\delta} - b)$. З погляду геометрії лінійний класифікатор відповідає деякій поділяючій гіперплощині, де об'єкт відноситься до першого класу, якщо він лежить з додатної сторони від гіперплощини, та до другого в протилежному випадку. Вектор $\vec{w}$ є перпендикуляром до поділяючої гіперплощини, а параметр b визначає її зсув відносно початку координат. Провести поділяючу гіперплощину можна по-різному, разом з тим оптимальною є така гіперплощина, яка максимізує відстань між нею та найближчим об'єктом класу. Метод опорних векторів зводить навчання класифікатора до задачі квадратичної оптимізації, яка розв'язується евристичними алгоритмами. Розв'язок – $\vec{w} = \sum_{i=1}^{N}\alpha_i\chi_i\vec{\delta}_i$. Для більшості векторів

$\alpha_i = 0$. Всі вектори, для яких $\alpha_i > 0$ називають опорними. Для будь-якого опорного вектора $b = \vec{w} \cdot \vec{\delta}_i - \chi_i$, тобто він належить опорній гіперплощині (всі об'єкти певного класу лежать по одну сторону від даної гіперплощини).

Об'єкти, що класифікуються, не завжди можуть бути розділені гіперплощиною. У реальних системах наявні похибки даних, внаслідок яких гіперплощина не виконає розподіл абсолютно точно. Тому для роботи методу SVM вводять допустиму похибку класифікації, що називається м'якою межею. Крім того, існує ще один шлях до вирішення проблеми лінійної нероздільності: вихідний простір можна відобразити в простір більш високого розміру, де навчальна вибірка стане лінійно роздільною: $\Phi : \vec{\delta} \to \phi(\vec{\delta})$ (спрямляючий простір).

Об'єкти навчальної вибірки входять в лінійну функцію прийняття рішень тільки у вигляді попарних скалярних добутків $\vec{\delta}_i \cdot \vec{\delta}_j$. Отже для того, щоб побудувати оптимальну поділяючу гіперплощину в новому просторі, необхідно знати лише $\phi(\vec{\delta}_i) \cdot \phi(\vec{\delta}_j)$. Припустимо, що існує деяка функція $K : R^n \to R$, така що $K(\vec{\delta}_i, \vec{\delta}_j) = \phi(\vec{\delta}_i) \cdot \phi(\vec{\delta}_j)$. Тоді для побудови оптимальної поділяючої гіперплощини не обов'язково задавати перетворення Φ в явному вигляді, достатньо лише знати K. При цьому функцію прийняття рішень можна переписати як $p(\vec{\delta}) = sign(\sum_{i=1}^{N} \alpha_i \chi_i K(\vec{\delta}_i, \vec{\delta}) - b)$. Такий підхід називають переходом до ядра (kernel trick).

У наших дослідженнях використовувалися два найбільш вживані для стеганоаналізу ядра SVM: лінійне $K(\vec{\delta}_i, \vec{\delta}_j) = \vec{\delta}_i \cdot \vec{\delta}_j + \theta, \theta \geq 0$ та гаусівське $K(\vec{\delta}_i, \vec{\delta}_j) = \exp\left(- \frac{\left\| \vec{\delta}_i - \vec{\delta}_j \right\|}{2\sigma^2} \right), \sigma > 0$.

Як альтернатива SVM аналізувався також ансамблевий класифікатор. Він працює за наступною схемою:

1) взяти *d* елементів (ознак) статистичної моделі характеристичних векторів;

2) отримати *L* випадково обраних підмножин із множини всіх елементів, кожна з яких складається з $d_{\text{sub}} < d$ ознак;

3) навчити *L* елементів ансамблевого класифікатора на навчальній вибірці розрізняти оригінальні зображення та стеганоконтейнери.

Нехай далі $N_v(\vec{z})$ – кількість елементів ансамблю, що голосують за належність зображення $\vec{z}$ до класу пустих. Рішення про мітку класу цього зображення приймається згідно наступному правилу:

$$Rule(L, N_v) = \begin{cases} -1, & \text{при } N_v > L/2, \\ +1, & \text{при } N_v < L/2, \\ random\{-1, +1\} & \text{інакше.} \end{cases}$$

Зауважимо, що існують різні варіанти вибору елементів ансамблю, проте слідуючи рекомендаціям у статті [10], автори якої запропонували використання ансамблевого класифікатора в стеганоаналітичних системах, ми зупинилися на лінійному дискримінанті Фішера (ЛДФ) з огляду на його швидке навчання та гарні результати отримані при вирішенні задач стеганоаналізу.

При використанні ЛДФ віднесення зображення до класу пустих чи заповнених відбувається згідно наступній функції прийняття рішень

$$p(\vec{\delta}) = \underset{\chi \in \{\chi_1, \chi_2\}}{arg\,max} \begin{bmatrix} ln(\rho_\chi P_\chi) - \frac{1}{2}(\vec{\delta} - \mu_\chi)^T \Sigma_\chi^{-1} (\vec{\delta} - \mu_\chi) - \\ \frac{1}{2} ln\left(\left| \Sigma_\chi^{-1} \right| - \frac{d_{sub}}{2} ln(2\pi) \right) \end{bmatrix},$$

де $\vec{\delta}$ – характеристичний вектор досліджуваного зображення, χ_1, χ_2 – мітки класів пустих та заповнених контейнерів, ρ_χ – ваговий коефіцієнт (величина штрафу за хибку класифікацію), P_χ – апріорна ймовірність появи контейнерів класу χ (частка пустих або заповнених контейнерів у тестовому наборі), μ_χ – середні значення характеристичних векторів класу χ, Σ – оцінка матриці коваріації характеристичних векторів пустих чи заповнених контейнерів.

Для проведення чисельних експериментів використовувався пакет MatlabR2019. Навчання на класифікація за методом опорних векторів реалізовувалися за допомогою Matlab-функцій fitcsvm та predict відповідно. Також використовувалася Matlab-реалізація ансамблевого класифікатора на базі ЛДФ, описана [10] та доступна для громадського використання на ресурсі http://dde. binghamton.edu/. У кожному з варіантів класифікаторів були задіяні налаштування параметрів за замовчуванням.

Оцінка швидкості складових процесів стеганоаналізу. Обчислення характеристичних векторів контейнерів відбувається як на етапі навчання стеганоаналітичної системи, так і на етапі детектування стеганоприховувань. Для можливості порівняння швидкості цього процесу для різних моделей в таблиці 1 наведена швидкість створення одного вектора, усереднена за 100 експериментами та обчислена на двох різних персональних комп'ютерах (ПК) середнього рівня. Всі моделі в таблиці впорядковані за зростанням розмірності характеристичного вектора.

Таблиця 1

Параметри характеристичних векторів для різних моделей

№ п/п	Параметри / Модель	Кількість елементів вектора	Швидкість обчислення вектора на 1-му ПК, сек	Швидкість обчислення вектора на 2-му ПК, сек
1	LIU	216	44,2	19,1
2	CHEN	486	0,2	0,1
3	CC-PEV	548	1,5	0,6
4	CC-CHEN	972	0,9	0,5
5	DCTR	8000	2,0	1,3
6	GFR	17000	6,2	3,8
7	CC-C300	48600	1,1	0,6

Як бачимо з даних таблиці 1 різниця в швидкості обчислення характеристичних векторів для різних моделей досить помітна. Найшвидше обчислюються вектори для моделі CHEN, найповільніше – для LIU: коли в моделі LIU буде опрацьований тільки один файл, в моделі CHEN таких файлів буде біля 200 або й більше.

Обчисливши характеристичні вектори для всіх наборів тестових даних та всіх моделей, далі за допомогою генератора випадкових чисел, ми ділили кожен тестовий набір на дві половини – по 557 контейнерів. Перша половина використовувалася для навчання класифікатора, друга – контрольна. Щоб отримати достатньо стабільні результати в сенсі незалежності оцінок якості від поділу на навчальну та контрольну вибірку, в подальшому ми повторювали експерименти по 10 разів, кожного разу змінюючи стартове число генератора. Результуючі оцінки обчислювалися як середнє значення в кожній такій серії тестів, також для оцінювання рівня стабільності прораховувалося середньоквадратичне відхилення. Декілька експериментів були повторені по 100 або 1000 разів, результуючі оцінки точності співпали з отриманими за 10 повторами як мінімум до цілих значень.

На швидкість навчання SVM класифікатора безпосередньо впливає кількість елементів характеристичного вектора: чим більше елементів, тим повільніше будується поділяюча гіперплощина. Таким же чином на швидкість навчання впливає і кількість контейнерів навчальної вибірки. Разом з тим недостатня кількість контейнерів навчальної вибірки приводить до швидкого навчання, але зниженої точності. А завелика кількість – до сповільненого навчання, яке не покращує точність подальшого стеганоаналізу.

Аналіз публікацій показав, що у комбінації з моделями високої розмірності стеганоаналітики зазвичай використовують ансамблевий класифікатор, так як він краще за SVM масштабується за розмірністю характеристичного вектора і за кількістю контейнерів у навчальній вибірці.

Орієнтовна швидкість навчання SVM з лінійним та гаусівським ядром, а також ансамблевого класифікатора на базі ЛДФ наведена у таблиці 2 (середнє значення M та стандартне відхилення σ). Орієнтовна швидкість класифікації – у таблиці 3.

Таблиця 2

Орієнтовна швидкість навчання класифікатора для різних моделей

№ п/п	Параметри / Модель	Кількість елементів вектора	Швидкість навчання класифікатора, сек					
			Лінійна SVM		Гаусівська SVM		Ансамбль	
			M	σ	M	σ	M	σ
1	LIU	216	0,06	0,01	0,09	0,01	0,7	0,18
2	CHEN	486	0,2	0,02	0,4	0,07	2,7	0,70
3	CC-PEV	548	0,3	0,01	0,7	0,02	2,3	0,10
4	CC-CHEN	972	0,5	0,01	1,1	0,15	7,1	0,67
5	DCTR	8000	9,4	0,17	12,0	0,17	56,3	43,64
6	GFR	17000	21,5	0,15	25,7	0,33	56,5	23,53
7	CC-C300	48600	46,3	0,80	66,9	0,89	33,2	13,60

Таблиця 3

Орієнтовна швидкість передбачення для різних моделей

№ п/п	Параметри / Модель	Кількість елементів вектора	Швидкість навчання класифікатора, сек					
			Лінійна SVM		Гаусівська SVM		Ансамбль	
			M	*σ*	*M*	*σ*	*M*	*σ*
1	LIU	216	0,02	0,01	0,03	0,01	0,01	0,004
2	CHEN	486	0,04	0,004	0,08	0,03	0,03	0,03
3	CC-PEV	548	0,04	0,001	0,16	0,02	0,02	0,002
4	CC-CHEN	972	0,07	0,003	0,4	0,02	0,1	0,08
5	DCTR	8000	5,1	0,29	8,6	0,61	0,3	0,16
6	GFR	17000	14,8	0,73	23,5	0,40	0,4	0,06
7	CC-C300	48600	27,8	3,53	51,2	2,85	0,5	0,15

Якщо порівнювати отриману швидкодію різних класифікаторів, то бачимо, що найшвидше навчання для перших шести моделей забезпечила лінійна SVM, а на другому місці SVM із гаусівським ядром. Ансамблевий класифікатор програє в швидкості навчання для всіх моделей векторів, окрім моделі найбільшої розмірності CC-C300. А от швидкість передбачення у нього найкраща, на другому місці лінійна SVM, а найповільніше передбачення для SVM з гаусівським ядром. Також відмітимо, що в даних експериментах швидкість навчання ансамблевого класифікатора для «багатих» моделей, тобто DCTR, GFR та CC-C300 сильно залежить від вхідного набору векторів, про що свідчить отримане відносно високе середньоквадратичне відхилення – 43,64, 23,53 та 13,6 секунд відповідно (такий розкид в першу чергу спричинений різною кількістю ітерацій при автоматичному пошукові оптимального підпростору).

Зауважимо, що зі збільшенням кількості контейнерів у навчальній вибірці ансамблевий класифікатор найшвидше навчається не тільки для CC-C300, але й для двох інших «багатих» моделей. Зокрема, збільшивши начальну вибірку в два рази, ми отримали швидкодію, відображену в таблиці 4. Проте збільшення навчальної вибірки не гарантує покращення точності детектування, тож потрібно розглядати його доцільність з огляду на конкретну стеганоаналітичну систему.

Разом з тим в першу чергу стеганоаналітична система повинна мати високу точність, тому давати оцінку ефективності моделі на основі лише швидкодії не є доречним.

Оцінка точності класифікації та порівняння її для різних моделей і класифікаторів. Точність виявлення стеганоконтейнерів на базі досліджуваних статистичних моделей характеристичних векторів за умови їх комбінації з лінійною SVM наведена у таблиці 5, з гаусівською – у таблиці 6, з ансамблевим класифікатором – у таблиці 7.

Таблиця 4

Орієнтовна швидкість навчання класифікатора при збільшенні навчальної вибірки в 2 рази

№ п/п	Параметри / Модель	Кількість елементів вектора	Швидкість навчання класифікатора, сек					
			Лінійна SVM		Гаусівська SVM		Ансамбль	
			M	*σ*	*M*	*σ*	*M*	*σ*
1	LIU	216	0,2	0,01	0,3	0,3	1,3	0,4
2	CHEN	486	0,6	0,1	1,2	0,04	3,8	0,5
3	CC-PEV	548	0,8	0,02	2,3	0,2	3,6	0,2
4	CC-CHEN	972	1,2	0,1	2,8	0,06	14,6	0,3
5	DCTR	8000	27,2	0,5	42,9	0,2	24,7	8,8
6	GFR	17000	67,4	0,9	91,0	0,5	38,1	16,3
7	CC-C300	48600	136,6	2,7	218,3	2,4	31,0	15,8

Таблиця 5

Точність виявлення стеганоконтейнерів за допомогою лінійної SVM, %

№ п/п	Атака на / Модель	Jsteg		Jphide		Steganos		J-UNIWARD	
		M	*σ*	*M*	*σ*	*M*	*σ*	*M*	*σ*
1	CHEN	99,9013	0,0662	84,2729	0,8049	91,6068	0,5047	50,3680	0,4852
2	CC-CHEN	99,8923	0,0708	87,9443	0,7194	95,7899	0,3912	50,5476	0,3552
3	LIU	99,9282	0,0825	92,6661	0,5583	99,5871	0,1805	51,6068	0,5824
4	CC-PEV	99,9731	0,0434	94,0575	0,6373	96,9210	0,3224	50,0808	0,4113
5	CC-C300	95,0359	0,5895	82,9264	1,1580	78,1239	0,9961	50,0987	0,1914
6	GFR	92,9443	0,9917	91,9749	0,2998	86,1490	0,9843	52,1813	0,8271
7	DCTR	98,9048	0,1835	92,1634	0,8816	91,4811	0,8921	50,5476	0,1716

Таблиця 6

Точність виявлення стеганоконтейнерів за допомогою гаусівської SVM, %

№ п/п	Атака на / Модель	Jsteg		Jphide		Steganos		J-UNIWARD	
		M	*σ*	*M*	*σ*	*M*	*σ*	*M*	*σ*
1	CHEN	98,7792	0,2649	78,5368	1,1382	84,6499	1,0103	50,6463	0,4434
2	CC-CHEN	97,7469	0,3627	73,6355	1,2954	84,4883	0,8803	50,3321	0,5518
3	LIU	99,8025	0,1019	91,1221	0,5459	99,4345	0,1529	51,5619	0,7146
4	CC-PEV	96,7056	0,6515	84,6858	1,1652	81,7953	1,2973	50,1167	0,4713
5	CC-C300	99,2280	0,2938	93,1777	0,6190	95,4039	0,4494	50,1975	0,1930
6	GFR	64,2729	0,7814	74,8384	0,7379	57,8366	0,6220	51,2118	0,4403
7	DCTR	67,8636	0,7043	74,6140	0,4651	57,0736	0,8258	50,4399	0,1866

Таблиця 7

Точність виявлення стеганоконтейнерів за допомогою ансамблевого класифікатора, %

№ п/п	Атака на / Модель	Jsteg		Jphide		Steganos		J-UNIWARD	
		M	*σ*	*M*	*σ*	*M*	*σ*	*M*	*σ*
1	CHEN	99,5871	0,2546	85,2065	0,7685	96,9569	0,6466	50,4309	0,9143
2	CC-CHEN	99,6948	0,2325	90,3591	0,4939	98,1508	0,3620	50,9695	0,4249
3	LIU	99,8923	0,1019	94,0844	0,8687	99,8025	0,1182	51,6697	0,9558
4	CC-PEV	100	0	94,4794	0,7585	98,8600	0,3993	51,4722	0,5406
5	CC-C300	99,4883	0,1586	94,0664	0,5809	96,5440	0,5913	51,0233	0,7221
6	GFR	98,3752	0,4386	95,0987	0,6747	92,8546	0,8705	62,1813	0,5955
7	DCTR	99,7217	0,1960	95,3232	0,6998	97,3160	0,4625	58,6086	0,7910

Порівнявши оцінки точності, отримані з використанням різних класифікаторів, ми бачимо, по-перше, що з найбільшою вірогідністю були виявлені Jsteg стегановкладки, які, як продемонстровано на початку роботи, мають найбільше розбіжностей з оригінальними контейнерами. Наступна за ймовірністю виявлення – Steganos Privacy Suite. Дещо гірше за неї виявляються вкладки програми Jphide. А от, на відміну від однокілобайтових вкладок на базі НЗБ стеганографії, стегановкладки аналогічних розмірів J-UNIWARD практично не виявляються (виявлення низької ймовірності присутнє лише для комбінації ансамблевого класифікатора і моделей GFR та DCTR).

Найкращу точність майже для всіх варіантів моделей векторів та стеганоперетворень у наших дослідженнях забезпечив ансамблевий класифікатор. В тих небагатьох випадках, коли кращу оцінку дала лінійна SVM, розрив між нею та ансамблевим класифікатором не значний, на рівні похибки через різні вхідні дані. Разом з тим через автоматичний пошук оптимального підпростору, присутній

в налаштуваннях за замовчуванням, ансамблевий класифікатор продемонстрував найповільніше навчання. Зауважимо, що метод SVM також передбачає опцію автоматичного пошуку оптимальних гіперпараметрів, але при цьому він уповільнюється суттєво більше, ніж ансамблевий класифікатор навіть для низькорозмірних векторів, а отримані оцінки точності покращують попередні результати для SVM (особливо варіант з гаусівським ядром), проте не покращують, а максимум співставні з оцінками на базі ансамблю. Наприклад, швидкість навчання лінійної SVM для моделі LIU зростає з 0,06 до 248,9 сек в середньому, в той час як середня точність співставна з попередньою (99,9282 та 99,9192 відповідно). Для гаусівської SVM і LIU при сповільненні навчання з 0,09 до 43,3 сек точність незначно покращується – з 99,8025 до 99,9102. Для моделі CC-CHEN і лінійної SVM пошук оптимальних гіперпараметрів сповільнює швидкість навчання з 0,5 до 376,3 сек в середньому, що покращує точність досить незначно: з 99,8923 до 99,9372. Для CC-CHEN і SVM з гаусівським ядром швидкість навчання стає не 1,1, а 114 сек, але точність покращується більш суттєво: з 97,7469 до 99,9013. Для моделі GFR швидкість вже 2226 сек при покращенні точності з 92,9443 до 97,9623% для лінійної SVM, і 2463 сек при покращенні точності з 64,2729 до 97,5853% для гаусівської. Для інших моделей були отримані подібні результати.

Наостанок впорядкуємо досліджені моделі за досягнутою точністю (на базі ансамблю як найточнішого варіанту).

Таблиця 8

Впорядкування моделей за досягнутою точністю (ансамблевий класифікатор)

№ п/п	Jsteg	Jphide	Steganos	J-UNIWARD
1	CC-PEV	DCTR	LIU	GFR
2	LIU	GFR	CC-PEV	DCTR
3	DCTR	CC-PEV	CC-CHEN	LIU
4	CC-CHEN	LIU	DCTR	CC-PEV
5	CHEN	CC-C300	CHEN	CC-C300
6	CC-C300	CC-CHEN	CC-C300	CC-CHEN
7	GFR	CHEN	GFR	CHEN

Ми бачимо, що топ-3 лідерів серед цих моделей за точністю виявлення НЗБ стеганографії – це CC-PEV, LIU та DCTR. Також бачимо, що найчутливішими до перетворення J-UNIWARD є GFR та DCTR, а от щоб зробити висновки по іншим моделям потрібно збільшувати стеганонавантаження на контейнери, тобто кількість прихованих даних.

Напрямки подальших досліджень. Зауважимо, що у випадках, коли допустиме зниження швидкодії, існують резерви покращення точності. Так, можна не обирати якусь одну модель формування характеристичних векторів, а використовувати декілька ефективних для даного типу контейнерів та стеганоперетворення. При цьому можна як комбінувати чи усереднювати результати на базі різних моделей (наприклад, байесівське усереднення), так і навчити окрему модель тому, яку саме з наявних моделей використати для передбачення (наприклад, дерево прийняття рішень). Зокрема для ансамблевого класифікатора як результат голосування за певний клас можна розглянути загальну суму голосів за цей клас, отриману кожним елементом ансамблю, при навчанні його на характеристичних векторах різних статистичних моделей.

Окрім того, як показали проведені дослідження при однаковому розмірі вкраплюваного повідомлення J-UNIWARD приховує таємні дані значно безпечніше за методи на базі НЗБ стеганографії. Тому в подальшому має сенс більш детальне дослідження властивостей цього методу. А саме, яку кількість даних можна безпечно приховати, яка статистична модель чи комбінація моделей є найбільш ефективною для протидії, як змінюються базові характеристики стеганоаналітичної системи зі зміною параметрів класифікатора, тощо.

ЛІТЕРАТУРА

[1]. Н. Кошкина, "Обзор и классификация методов стеганоанализа", *Управляющие системы и машины*, № 3, С. 3-12, 2015.

[2]. T. Filler, J. Judas, J. Fridrich, "Minimizing Embedding Impact in Steganography using Trellis-Coded Quantization", *Proc. SPIE, Electronic Imaging, Media Forensics and Security XII*, San Jose, CA, vol. 7541, 2010.

[3]. C. Chen, Y.Q. Shi, "JPEG image steganalysis utilizing both intrablock and interblock correlations", *IEEE ISCAS, International Symposium on Circuits and Systems*, pp. 3029-3032, 2008.

[4]. J. Kodovsky, J. Fridrich, "Calibration revisited", *In J. Dittmann, S. Craver, and J. Fridrich, editors, Proceedings of the 11th ACM Multimedia and Security Workshop*, pp. 63-74, 2009.

[5]. Q. Liu, "Steganalysis of DCT–embedding based adaptive steganography and YASS", *In J. Dittmann, S. Craver, and C. Heitzenrater, editors, Proceedings of the 13th ACM Multimedia & Security Workshop*, pp. 77-86, 2011.

[6]. T. Pevny, J. Fridrich, "Merging Markov and DCT features for multiclass JPEG steganalysis", *In E. J. Delp, P. W. Wong, editors, Proceedings SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, pp. 301-314, 2007.

[7]. J. Kodovsky, J. Fridrich, "Steganalysis in high dimensions: fusing classifiers built on random subspaces", *8th SPIE Electronic Imaging, Media, Watermarking, Security and Forensics*, vol. 7880, pp. 1-13, 2011.

[8]. X. Song, F. Liu, C. Yang, X. Luo, Y. Zhang, "Steganalysis of Adaptive JPEG Steganography Using 2D Gabor Filters", *Proceedings of the 3rd ACM Workshop on Information Hiding and Multimedia Security. ACM*, pp.15-23, 2015.

[9]. V. Holub, J. Fridrich, "Low Complexity Features for JPEG Steganalysis Using Undecimated DCT", *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 2, pp. 219-228.

[10]. J. Kodovský, J. Fridrich, V. Holub, "Ensemble Classifiers for Steganalysis of Digital Media", *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 432-444, 2012.

RESEARCH OF MAIN COMPONENTS OF MACHINE LEARNING BASED JPEG-STEGANANALYSIS SYSTEMS

To create effective steganalysis systems in the given practical conditions it is necessary to perform analysis and quality estimation of existing methods and components. To select optimal system components it is required to compare the estimates of basic characteristics of the candidates available. However, making such comparison based on a data from scientific publications is quite difficult due to differences of the conditions of numerical experiments. The basis of this study is the principle of creating equal conditions for all the investigated statistical features based models for JPEG steganalysis using machine-learning methods. We analyzed the detection performance and accuracy for four different variants of data hiding in the frequency domain that were obtained using statistical models such as CHEN, CC-CHEN, LIU, CC-PEV, CC-C300, GFR and DCTR, as well as SVM with linear or Gaussian kernel functions or ensemble classifier. The main results of the study are tables containing numerical estimates of performance of the main stages of steganoanalysis and the classification accuracy of empty and stego images. Model LIU was the slowest for extraction of features vector, but it is the fastest in the classifier training process. Models CC-PEV, LIU and DCTR combined with the ensemble classifier provided the best detection accuracy for stego images created by programs like Jsteg, Jphide and Steganos Privacy Suite. The detection accuracy of stego images by Jsteg, Jphide and Steganos in our experiments reached 100, 95.3 and 99.8% accordingly. J-UNIWARD method has proved itself to be more secure than LSB-steganography. The GFR and DCTR models are proved to be the most sensitive to J-UNIWARD transformation. In distinction to other investigated models they analyze statistics in the spatial domain of the image but not in the frequency domain - where the message was embedded.

Keywords: information security, steganalysis, passive counteraction (steganalysis), machine learning methods, comparative analysis, features based models, SVM, ensemble classifier.

ИССЛЕДОВАНИЕ ОСНОВНЫХ КОМПОНЕНТОВ СИСТЕМ JPEG-СТЕГАНОАНАЛИЗА НА БАЗЕ МАШИННОГО ОБУЧЕНИЯ

Для построения эффективных стеганоаналитических систем в заданных практических условиях необходимо провести анализ и оценку качества существующих методов и компонентов. Для выбора оптимальных составляющих системы необходимо сравнить оценки базовых характеристик имеющихся кандидатов. Однако осуществить такое сравнение, основываясь на данных из научных публикаций, достаточно сложно из-за различий в условиях численных экспериментов. В основе данного исследования лежит принцип создания равных условий для всех исследуемых статистических моделей формирования характеристических векторов для стеганоанализа JPEG-изображений методами на базе машинного обучения. Проанализированы быстродействие и точность детектирования четырех различных вариантов сокрытия данных в частотной области, которые были получены с использованием таких статистических моделей как CHEN, CC-CHEN, LIU, CC-PEV, CC-C300, GFR и DCTR, а также SVM с линейным или гаусовским ядром или ансамблевого классификатора. Основными результатами проведенного исследования являются таблицы, отражающие численные оценки быстродействия основных этапов стеганоанализа и точности классификации пустых и заполненых контейнеров.

Ключевые слова: информационная безопасность, стеганоанализ, пассивное противодействие, методы с обучением и классификацией, сравнительный анализ, модели характеристических векторов, SVM, ансамблевый классификатор.

Кошкіна Наталія Василівна, доктор технічних наук, старший науковий співробітник, старший науковий співробітник відділу оптимізації чисельних методів, Інститут кібернетики імені В.М. Глушкова НАН України.
E-mail: nata.koshkina@gmail.com.
Orcid ID: 0000-0001-5180-2255.

Кошкина Наталья Васильевна, доктор технических наук, старший научный сотрудник, старший научный сотрудник отдела оптимизации численных методов, Институт кибернетики имени В.М. Глушкова НАН Украины.

Koshkina Natalia, Doctor of Engineering Sciences (Information security), Senior Researcher, Department of Numerical Methods Optimization, V.M. Glushkov NAS of Ukraine.