

РОБАСТНИЙ АЛГОРИТМ ВБУДУВАННЯ ЦИФРОВИХ ВОДЯНИХ ЗНАКІВ В ЦИФРОВІ ЗОБРАЖЕННЯ

Ю. В. Науринський¹, А. Н. Барановський²

^{1, 2} Національний технічний університет України «Київський політехнічний інститут»

Анотація

В даній роботі описується алгоритм вбудовування цифрових водяних знаків в цифрові зображення. Отримані результати, надаються висновки та рекомендації, щодо використання алгоритму.

Ключові слова: цифровий водяний знак, цифрове зображення, алгоритм

Вступ

Проблема захисту авторських прав зображень завжди була актуальна у всі часи, починаючи з того часу, як людство винайшло живопис. В наш час, воно постає з новою силою, переходить у нову, раніше небачену цифрову площину мережі Інтернет, де кожний її користувач, кожен день генерує десятки, сотні цифрових зображень. Кожне з них зловмисник може використати в своїх неприйнятних цілях, поширювати без дозволу, модифікувати, спотворюючи оригінальний зміст зображення. Зазвичай, для захисту зображень використовують методи вбудовування цифрових водяних знаків у вигляді нанесених на зображення тексту, або додаткового зображення. У цій роботі пропонується метод створення цифрового водяного знаку, не помітного для людського ока, для виявлення несанкціонованих модифікування та використання зображення. В подальшому, під водяним знаком розуміється спеціальне графічне цифрове зображення, яке містить інформацію, яка використовується для доказу авторських прав власника оригінального зображення.

1. Опис запропонованого алгоритму

Цей алгоритм базується на запропонованому алгоритмі [1] який був значно модифікований автором для підвищення ефективності алгоритму, його швидкодії, та зручності у реалізації, ціною деякого його спрощення.

1.1. Вбудовування водяного знаку в оригінальне зображення

Нехай, є два зображення, оригінальне X з розміром $N \times N$ та зображення, яке використовується в якості цифрового водяного знаку w з розмірами $M \times M$. Накладається обмеження на розмір водяного знаку, він повинен бути не більше ніж розмір оригінального зображення. Зазвичай, розмір водяного знаку вдвічі менше ніж розмір оригінального зображення, але це не обов'язково.

Схема вбудовування:

Крок 1. Обраховується хеш-суму водяного знаку, для контролю несанкціонованої модифікації оригінального зображення.

Крок 2. Власник оригінального зображення зашифровує водяний знак w алгоритмом шифрування зображень, який базується на використанні логістичних мап хаосу, отримуючи зашифрований водяний знак w' .

Крок 3. Тричі застосовується вейвлет декомпозиція оригінального зображення X , з середньочастотної ділянки декомпозиції беруться вейвлет коефіцієнти $CA3$.

Крок 4. Застосовується вейвлет декомпозиція для водяного знаку, з цієї декомпозиції беруться вейвлет коефіцієнти $cw1$.

Крок 5. Відбувається вбудовування водяного знаку в оригінальне зображення використовуючи формулу:

$$CA3'(i, j) = CA3(i, j) + \alpha * CA3(i, j) * cw1(i, j) \quad (1)$$

де α відповідає за робастність водяного знаку, тобто чим більше значення α , тим більше стійкість водяного знаку, але й тим більше його видно в вихідному зображенні. Чим воно менше, тим менш стійкий водяний знак, також у вихідному зображенні він вже менш помітний.

Крок 6. Тричі застосовується вейвлет реконструкція до зображення, яке було розкладено за допомогою вейвлет декомпозиції, отримано фінальне зображення з вбудованим водяним знаком

1.2. Здобування водяного знаку з зображення, в якому він міститься

Схема здобування водяного знаку зворотня до схеми вбудовування. Так як це не сліпа схема, оригінальне зображення потрібно для здобування вейвлет коефіцієнтів, які потім комбінуються з вейвлет коефіцієнтами зображення з вбудованим водяним знаком. Вже з цієї комбінації отримується водяний знак, який використовують для доказу авторського права.

Схема здобування:

Крок 1 Власник оригінального зображення тричі застосовує вейвлет декомпозицію зображення з вбудованим водяним знаком, з середньочастотної ділянки декомпозиції беруться вейвлет коефіцієнти $CA3'$.

Крок 2 Тричі застосовується вейвлет декомпозиція оригінального зображення X , з середньочастотної ділянки декомпозиції беруться вейвлет коефіцієнти $CA3$.

Крок 3 Вейвлет коефіцієнти водяного знаку обчислюються, з рівняння:

$$cw1(i, j) = (CA3'(i, j)/CA3(i, j) - 1)/\alpha \quad (2)$$

Крок 4 Застосовується вейвлет реконструкція для отриманих коефіцієнтів зображення, яке ми отримали на попередньому кроці.

Крок 5 Зображення розшифровується за допомогою алгоритму шифрування, використаного при зашифруванні, це важливо.

Крок 6 Обчислюється хеш-сума отриманого водяного знаку, для перевірки несанкціонованої модифікації оригінального зображення, якщо вона відрізняється від тієї що у власника, це значить що оригінальне зображення спотворювали з метою несанкціонованої модифікації оригінального зображення.

2. Рекомендації щодо реалізації алгоритму

Для того щоб обчислити хеш-суму водяного знаку, можна використовувати будь-який такий алгоритм обчислення, проте важливо щоб вони співпадали, для отримання достовірних результатів. Найкраще використовувати криптографічно стійкі функції такі як наприклад сімейство SHA, проте вони потребують значних обчислювальних ресурсів, тому треба обирати баланс між стійкістю та швидкістю.

Для шифрування водяного знаку за допомогою логістичних мап хаосу, автор використовував алгоритм [2], який надає зручний баланс між надійністю та швидкодійністю. Але реалізація кінцевого алгоритму залежить від кінцевого користувача алгоритму, важливо, щоб ці алгоритми співпадали на процесах вбудовування та здобування водяного знаку.

Вибір алгоритму вейвлет декомпозиції базується на балансі між простотою його реалізації та якості вихідного зображення з вбудованим водяним знаком. Автор цієї роботи використовував алгоритм [3], так як це подальший розвиток вейвлетів Хаара, з покраще-

ною якістю вихідного зображення, ціною незначного підвищення складності алгоритму.

Так як алгоритм призначений для реалізації на комп'ютері, постає питання, яку мову програмування доцільно обрати для реалізації цього алгоритму. Вибір мови програмування, це ще один шарнір гнучкості реалізації алгоритму. Якщо обирати мову низького рівня, тобто наприклад C чи наприклад взагалі асемблер, алгоритм буде працювати дуже швидко, проте ці мови не надто зручні для такого високорівневого програмування. Мови скриптові типу Python, JavaScript надто повільно виконують алгоритм. Саме тому, доцільною мовою програмування є мови так званого середнього рівня, тобто Java чи C#. На обох цих мовах є весь необхідний функціонал для реалізації алгоритму максимально ефективним та зручним у користуванні.

Для порівняння двох зображень використовують так звану середньоквадратичну похибку чи пікове співвідношення сигнал/шум, яке вимірює рівень спотворень при перетворенні зображень, вимірюється у децибелах, чим воно більше, тим менше спотворень у зображенні. Саме так порівнюються оригінальне зображення, та зображення з вбудованим водяним знаком.

Висновки

В цій роботі був описаний ефективний та гнучкий алгоритм вбудовування водяного знаку в цифрове зображення з метою захисту його авторських прав, захисту зображення від несанкціонованого модифікування. Цей алгоритм пропонує гнучкий баланс ефективності, швидкодії, легкості реалізації, за допомогою гнучкої блочної системи його побудови, де на кожному кроці обирається відповідний метод відповідно до потреб користувача цього алгоритму. Детальний практичний огляд алгоритму буде представлений на конференції.

Перелік використаних джерел

1. Chittaranjan Pradhan, Bidyut Jyoti Saha, Kunal Kumar Kabi, Ajay Kumar Bisoi Robust Watermarking Technique using 2D Logistic Map and Elliptic Curve Cryptosystem in Wavelets. — 2014. —
2. Narendra K Pareek, K.K. Sud Image encryption using chaotic logistic map. — 2006. —
3. Jinal Patel, Ketki Pathak Implementation of the 5/3 Lifting 2D Discrete Wavelet Transform. — 2014. —