**УДК 681.3, 004.62**

**Берегуляк Ю.В., Глухов В.С.**

Національний університет "Львівська політехніка"

кафедра електронних обчислювальних машин

**Порівняння алгоритмів стиску зображень без втрат JPEG-LS і FLIF для подальшої реалізації їх на ПЛІС**

*© Берегуляк Ю.В., Глухов В.С., 2019*

**Розглянуто проблему використання алгоритмів стиску зображень без втрат. Проаналізовані алгоритми JPEG-lS і FLIF. Знайдено аналоги програмної реалізації і зібрані початкові дані.**

**Ключові слова: алгоритми стиску зображень, ПЛІС.**

**The problem of using lossless image compression algorithms is considered. JPEG-lS and FLIF algorithms are analyzed. Analogs of software implementation were found and initial data were collected.**

**Keywords: image compression algorithms, FPGA.**

**Вступ.** Постійні поліпшення якості (підвищення роздільної здатності і числа біт, що використовується для подання пікселя) і збільшення кількості одержуваних візуальних даних (зображень) призводять до зростання витрат на їх зберігання - побудови нових центрів зберігання даних. У зв'язку з цим завдання компактного представлення візуальних даних особливо актуальна в останні роки.

Для її вирішення було запропоновано два основні підходи:

- **стиснення без втрат** - вся інформація, що міститься на оригінальному документі, представлялася в більш компактному вигляді за рахунок застосування різних методів кодування джерела;

- **стиснення з втратами** - для підвищення ефективності стиснення допускалося часткове знищення або спотворення інформації, що міститься в оригінальному документі.

У даній роботі досліджуються способи стиснення різних класів зображень, що відносяться до загального класу високоточних, де важливі точність даних та швидкість їх передачі між вузлами системи. Для подібних зображень втрата і спотворення збережених в них даних є критичними, тому методи стиснення з втратами не розглядаються. У високонавантажених системах практичне значення має саме його апаратна реалізація.

У роботі розглянуто особливості виконання стиснення зображень без втрат методамм JPEG-LS і FLIF а також їх реалізацію на ПЛІС.

**Стан проблеми.** Список можливих застосувань апаратного кодера JPEG-LS/FLIF дуже широкий, ось лише деякі варіанти:

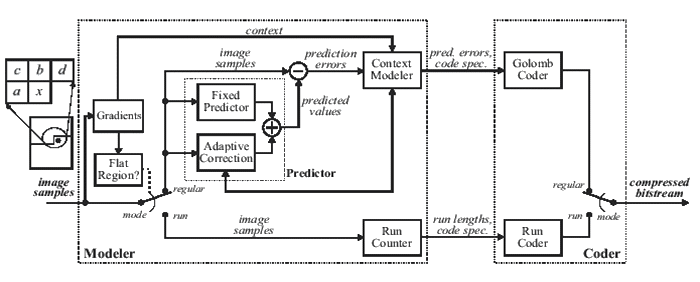
* Охоронні системи. Використання кодера в відеоохоронні системах може дозволити істотно (в рази) скоротити трафік цифрового відеопотоку або збільшити час запису для автономних систем.
* Системи збору зображень (в науковому, промишелнном обладнанні).
* Медична техніка. Кодер дозволить в реальному часі і, що дуже важливо, без втрат стискати медичні зображення (рентгенівські знімки, томограми, УЗД-й ендоскопічні зображення та ін.) Для подальшого архівування або передачі в проектах телемедицини.
* Відеостудійное обладнання, відеорекордери. Стиснення без втрат видеопоследовательностей дозволить зберігати вихідне якість відеоматеріалів, істотно заощаджуючи місце в системах зберігання.
* ВЕБ-камери. Кодер дозволить розробляти більш ефективні продукти з високою якістю зображення.
* CCD- і CMOS- камери. Використання кодера дозволить створювати більш ефективні рішення.
* Супутникові, підводні та інші спеціальні системи фото- і відеоспостереження.
* Астрономічні інструменти, телескопи.

У цих галузях найчастіше використовуються монохромні зображення, оскільки потрібна велика роздільна здатність і невеликі розміри. Також важливим є швидкодія компресії зображення, оскільки велика кількість даних надходить в реальному часі і повинна бути збереженна без затримок, бо вони можуть призвести до часткової втрати даних про зображення. Швидкість декомпресії не є такою критичною, оскільки зображення будуть аналізуватися вже після отримання всіх необхідних даних.

**Постановка задачі.** Знайти аналоги програмних реалізацій алгоритмів JPEG-LS і FLIF на мові C. Зробити свою реалізацію алгоритма FLIF на мові С.

Знайти множину зображень для порівняння. Порівняти результати роботи програмних реалізацій алгоритмів які використовувалися на даній множині. Ці результати буду використані після портування алгоритмів для ПЛІС.

**Розв’язання задачі.** Формат JPEG-LS базується на форматі LOCO-I (Low Complexity Lossless Compression for Images)[1]. Алгоритм стис без атрат LOCO-I, прийнятий за базу при розробці стандарту JPEG-LS, вперше передбачав не тільки lossless, але і near lossless режим (стиснення з обмеженими втратами, що задаються користувачем)[2]. Цей алгоритм є симетричний, бо кодування і декодування даних майже не відрізняється[3].

Рис. 1. Діаграма роботи алгоритма JPEG-LS

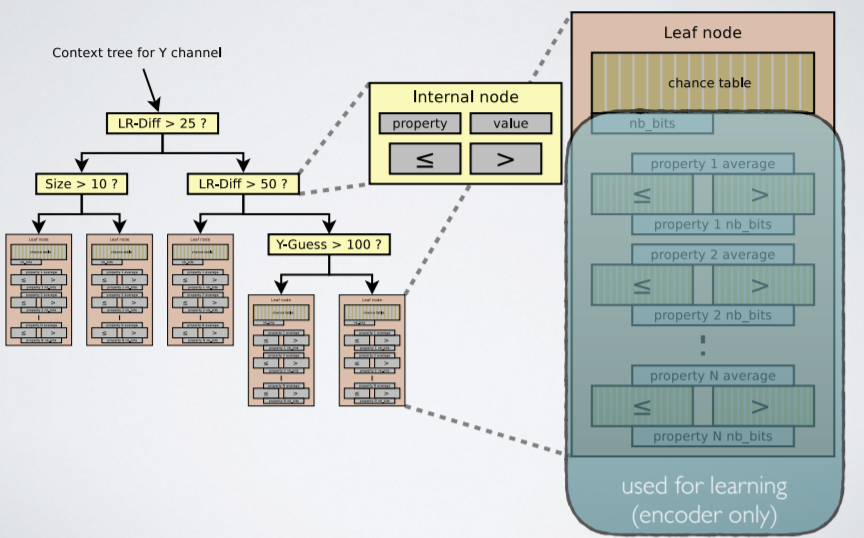
Free Lossless Image Format (FLIF) - безкоштовний формат зображення, який, за завіренням розробників, за степенами зберігає PNG, без втрат WebP, BPG без втрат і без втрат JPEG 2000[4]. FLIF використовує один з різних варіантів арифметичного кодування (MANIAC, метаадаптивне майже-нульове ціле арифметичне кодування) в якості алгоритму ентропійного стиску[5].

Рис. 2. Дерево MANIAC кодування

Для тестування алгоритмів було вирішено, використати множину зображеннь яка використовується при машинному навчанні[6]. Також була створена власна реалізація алгоритма FLIF для порівння з доступною для всіх реалізацією JPEG-LS. Оскільки для машинного навчання використовується велика кількість зображень з великою роздільною здатністю, тому було вирішено використати саме їх. Всі зображення з множини збережені у фіорматі .ppm. Вони є різноманітні: кольорові, монохромні, з великою роздільною здатність і навпаки.

В Табл.1 наведено результати використання алгоритмів JPEG-LS і FLIF на даній множині зображень. Як можна побачити з результатів, алгоритм FLIF стискає зображення майже у всіх випадках на ~10% краще ніж JPEG-LS.

Це зумовлено тим, що FLIF спеціально розроблявся для максимального стиску всіх видів зображень.

Табл. 1. Порівняльна таблиц результатів використання алгоритмів JPEG-LS та FLIF

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset (.ppm) | Size (bytes) | jpeg-ls size (bytes) | flif size (bytes) | jpeg-ls result (%) | flif - result (%) |
| stop01.ppm | 2,252,941 | 1,128,451 | 946,499 | 49.91209268 | 57.98829175 |
| stop02.ppm | 2,332,861 | 1,234,666 | 736,342 | 47.07502933 | 68.43609628 |
| stop03.ppm | 2,339,341 | 1,086,418 | 826,489 | 53.55880139 | 64.67000749 |
| stop04.ppm | 2,566,141 | 1,388,624 | 1,101,383 | 45.88668355 | 57.08018382 |
| stop05.ppm | 2,003,062 | 1,322,795 | 1,124,575 | 33.96135517 | 43.85720462 |
| stop10.ppm | 2,235,661 | 1,408,321 | 1,140,432 | 37.00650501 | 48.98904619 |
| stop11.ppm | 2,332,861 | 1,375,906 | 1,200,112 | 41.0206609 | 48.55621488 |
| stop12.ppm | 2,001,061 | 1,306,834 | 1,117,714 | 34.69294539 | 44.14393164 |
| stop13.ppm | 3,044,221 | 1,360,475 | 1,017,071 | 55.30958495 | 66.5901063 |
| stop14.ppm | 2,332,861 | 1,199,856 | 1,029,833 | 48.56718853 | 55.85536386 |
| stop16.ppm | 2,397,661 | 1,368,346 | 1,107,706 | 42.92996383 | 53.80055813 |
| znonstop01.ppm | 2,332,861 | 910,694 | 787,572 | 60.96235481 | 66.24008031 |
| znonstop02.ppm | 2,332,861 | 1,176,658 | 1,033,843 | 49.56158982 | 55.68347193 |
| znonstop03.ppm | 2,240,065 | 1,167,976 | 970,588 | 47.85972728 | 56.67143587 |
| znonstop04.ppm | 2,155,981 | 1,043,368 | 907,709 | 51.6058815 | 57.89809836 |
| znonstop11.ppm | 2,332,861 | 1,247,274 | 1,084,646 | 46.53457707 | 53.50575967 |

Також на Рис.3, можна побачити, що дійсно, що різниця між результатами майже стала. Наступними кроками, буде реалізувати дані алгоритми самостійно з можливістю покращення. Після цього, алгоритми будуть синтезовані для роботи с ПЛІС. Буде проведено ще одне порівняння апаратних реалізації. В кінці дослідження, Всі дані будуть проаналізованні і вибрана найкраща програмна і апаратна реалізація.

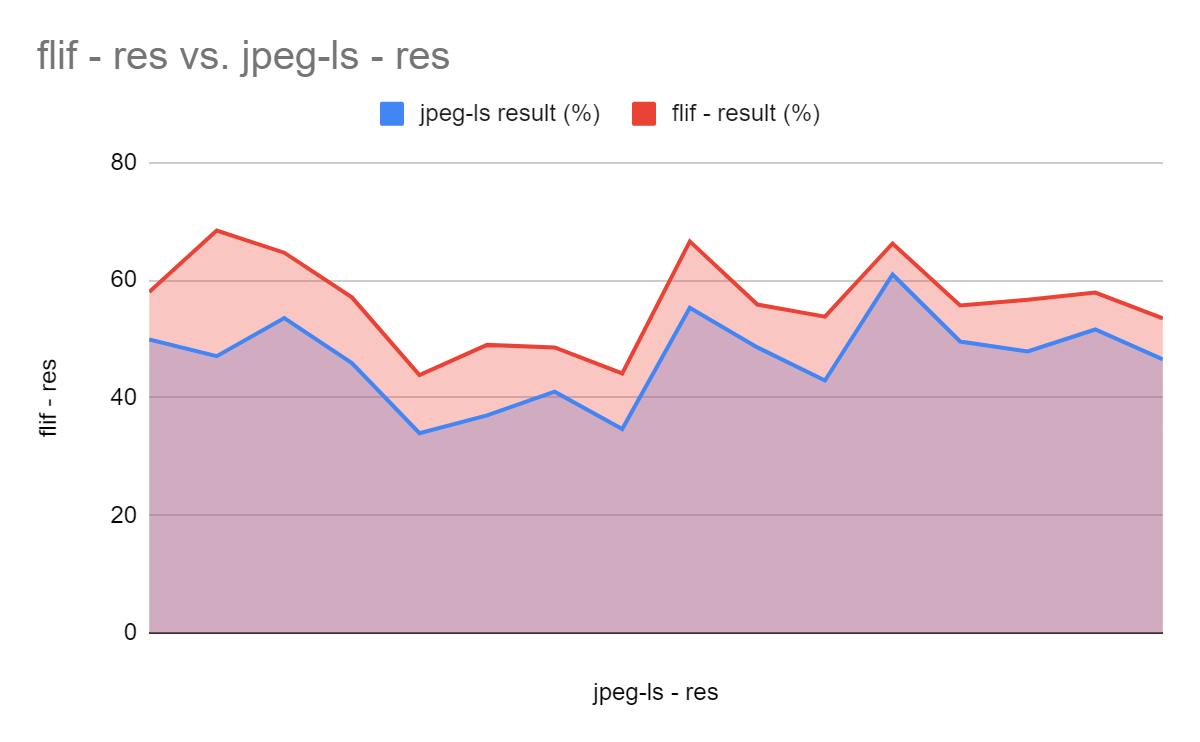


Рис. 3. Діаграма порівняння алгоритмів JPEG-LS та FLIF.

**Висновки.** У данй роботі були проаналізовання програмні реалізації алгоритмів стиску зображення без втрат JPEG-LS і FLIF. Отрмані дані, показали, що FLIF на ~10% ефективніше стискає всі види зображень.

**Література**

1. Глухов, В. С. Хоміць, В. М. Підхід до стиснення зображень без втрат методом JPEG-LS: Львів. НУ «Львівська Політехніка» 2017
2. Глухов, В. С. Хоміць, В. М. Підхід до реалізації на ПЛІС засобами пакета VIVADO C-описів алгоритму стиснення зображень: Львів. НУ «Львівська Політехніка» 2017
3. Vajnberger M., Seroussi G., Schapiro G. The Loco-I stysnennja zobrazhen bez vtrat. Algorytm: pryntsypy i standartyzatsiji v JPEG-LS // Hewlett-Packard Laboratories Technical Report No. HPL-98- 193R1, lystopad 1998, pereroblene zhovtnja +1999. IEEE Trans. Obrobka zobrazhen, Vol. 9 serpnja 2000 roku, pp. 1309–1324
4. Jon Sneyers Pieter Wuille FLIF: FREE LOSSLESS IMAGE FORMAT BASED ON MANIAC COMPRESSION 2016
5. [https://flif.info](https://flif.info/)
6. Hlukhov Valerii, Lukenyuk Adolf, Shenderuk Sergii. Satellite scientific data collection and accumulation system as a basis for cyber-physical systems construction. Advances in CyberPhysical Systems. Volume 1. Number 2. Lviv Polytechnic National University. 2016. – P. 77–86.