

УДК 004.932

© Дмитро Афанасьєв, аспірант, КПІ ім. Ігоря Сікорського, Київ, Україна

Систематизація методів стиснення цифрових зображень

Systematic methods of compressing digital images. Analyzes compression algorithms implemented in popular software imaging tools. Systematic compression tasks for the systems.

Сучасні системи репродукування відтворюють текстово-ілюстраційну інформацію з параметрами, що забезпечують її візуальне сприйняття в межах 2-9 полів тонової шкали при стандартному освітленні. Хоча цього і достатньо для людини — спостерігача або споживача зображувальної інформації, але для процесів оброблення в системах репродукування має важливе значення обсяг оброблюваної інформації. Як відомо і колірність, і контрастність, і насиченість, і роздільна здатність оригіналу визначають обсяги кодованого зображення, що у свою чергу спрощує або ускладнює оброблення і запис остаточного файлу для відтворення.

Тож актуальним завданням сучасного репродукування за наявності потужних апаратно-програмних комплексів є систематизація методів стиснення цифрових зображень з метою обґрунтування параметрів для зменшення спотворень при кодуванні і декодування зображень і репродукції.

Наведено методи стиснення зображень (рис.1) за джерелами [1-4]. Спосіб стиснення даних призначений для зниження обсягу вихідного потоку інформації за допомогою перетворення без втрат або перетворення з втратами. Тому способи стиснення поділяють на два класи: стиснення без втрат та з втратами. Методи стиснення цифрових зображень можна систематизувати за їх основним характеристиками: точність відновлення, симетричність основного перетворення і тип використовуваного перетворення [4].

Результатом стиснення без втрат завжди є зниження обсягу вихідного потоку інформації без зміни його інформативності, тобто без втрати інформаційної структури. З вихідного потоку за допомогою відновлювального алгоритму можна отримати вхідний потік [1]. Методи стиснення без втрат використовуються в основному в наукових і медичних прикладних програмах, коли втрата інформації неприпустима або самі шуми зображення є головною інформацією, наприклад, в системах оцінки якості оптико-електронних систем. Коефіцієнт стиснення досягається цими методами не більше 1,5.

Для симетричних методів стиснення процедури стиснення і відновлення однотипні. Час стиснення і відновлення для таких методів порівнянні. Для несиметричних методів процедура стиснення відрізняється від процедури відновлення і зазвичай займає більше машинного часу [1-4].

Стиснення з втратами — перетворення вхідного потоку даних у вихідний потік, котрий має певний формат інформації, повторює зовнішні характеристики вхідного потоку з різницею у обсязі. Ступінь подібності вхідного і вихідного потоків визначається ступенем відповідності певних властивостей зображення, що представляється даним потоком інформації. Методи стиснення з втратами дозволяють отримати істотно більші коефіцієнти стиснення. Однак при цьому відбувається спотворення вихідного зображення, погіршення його якості. У зв'язку з цим при порівнянні різних методів стиснення крім коефіцієнта стиснення потрібно враховувати якість відновлення зображення [1-4].

Всі методи стиснення інформації засновані на припущенні про надлишкові елементи в наборі даних. Стиснення досягається за рахунок пошуку та кодування надлишкових елементів. Потік даних про зображення має істотну кількість зайвої інформації, яка може бути усунена практично без помітних для ока спотворень. При цьому розрізняють два типи надмірності.

Статистична надлишковість пов'язана з кореляцією і передбачуваністю даних. Ця надмірність може бути усунена без втрати інформації, вихідні дані при цьому можуть

бути повністю відновлені. Найбільш відомі методи ефективного кодування символів засновані на знанні частоти кожного символу присутнього в повідомленні. Знаючи ці частоти, будують таблицю кодів, що має такі характеристики: різні коди можуть мати різну кількість біт; коди символів, які зустрічаються з більшою частотою, мають менше біт, ніж коди символів з меншою частотою; хоча коди мають різну бітову довжину, вони можуть бути відновлені єдиним чином, тобто коди будуються як префіксні. Ці характеристики забезпечує відомий алгоритм Хаффмана [1].

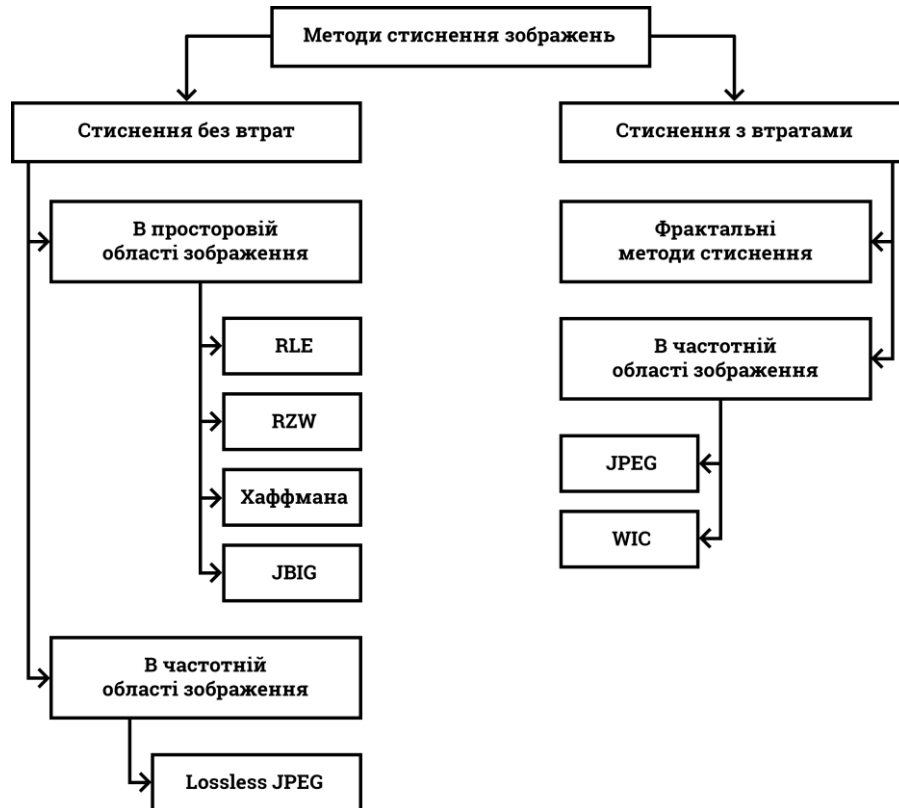


Рис. 1. Систематизація методів стиснення цифрових зображень

Візуальна (суб'єктивна) надлишковість, яку можна усунути з частковою втратою даних, мало впливає на якість відтворених зображень, бо не порушує візуальне сприйняття зображень. Усунення візуальної надмірності зображень є основним резервом скорочення інформації, що передається [1, 3]. Для оптимізації процесу кодування з метою забезпечення передачі найменшого обсягу інформації необхідно, з одного боку, не передавати надлишкову інформацію, а з іншого, — не допустити надмірної втрати якості зображення.

У процесі роботи із зображеннями програмне забезпечення, в якому здійснюється обробка, ставить різні вимоги до алгоритмів стиснення зображень. Через специфіку програмних продуктів такі вимоги іноді можуть суперечити один одному. У виразно такі головні вимоги до алгоритмів стиснення зображень: високий ступінь компресії; висока якість стисненого зображення (дана вимога суперечить виконанню попереднього вимоги, тому завжди присутній компроміс між ступенем стиснення і якістю відновленого зображення); висока швидкість компресії (дана вимога актуально для додатків, що займаються кодуванням зображень в реальному масштабі часу: цифрових фотоапаратів, відеокамер); висока швидкість декомпресії (дана вимога актуально майже для всіх додатків); можливість показати приблизне зображення, не чекаючи повної його завантаження (дана вимога актуально для мережних додатків і для додатків, що займаються передачею великих зображень); врахування специфіки зображення (дану вимогу реалізують

алгоритми стиснення, засновані на визначенні «області особливого призначення» (ROI — regions of interest)) [1-4].

Етапи процедури стиснення даних [1-4]: кодування або первинне стиснення — реалізується за алгоритмами JPEG, перетворення Хартлі, Уолша-Адамара, Вейвлет, Фрактальне; вторинне стиснення — Хаффмана, RLE, ZIP, HA; декодування або відновлення зображення.

Головні параметри і критерії стиснення [1-4]: коефіцієнт стиснення ($K_{ст}$); оцінка якості стисненого кодованого зображення; час перетворення.

Тож для оцінювання методів стиснення аналізують критерії відповідно до класу зображень, на який орієнтований алгоритм, приміром за [3]: власне клас зображень; найгірший, середній і кращий коефіцієнт стиснення — якийсь середньостатистичний коефіцієнт; симетричність — характеризує ресурсомісткість процесів кодування і декодування (при цьому найбільш важливим є відношення часу кодування до часу декодування); втрати якості; характерні особливості алгоритму і зображень, до яких його застосовують.

Систематизація алгоритмів стиснення полягає у визначенні актуального для завдань репродукування за порівнянням переваг і недоліків.

Зокрема, стиснення зображень без втрат способом кодування серій Run Length Encoding (RLE) — це кодування серій послідовностей [4]. Суть полягає в заміні ланцюжків або серій повторюваних байтів або їх послідовностей на один кодуєчий байт і лічильник числа їх повторень. Кращий, середній і найгірший коефіцієнти стиснення 1/32, 1/2, 2/1. Подібні методи ефективні для стиснення растрових тонових зображень (BMP, PCX, TIFF), оскільки містять довгі серії повторюваних послідовностей байтів. RLE не вимагає додаткової пам'яті при роботі і швидко виконується, проте має низький ступінь стиснення.

Кодування Хаффмана актуальний для текстових файлів, де символи тексту замінюються ланцюжками біт різної довжини, і забезпечує однозначну побудову коду з найменшим для даного розподілу ймовірностей середнім числом символів на літеру [1].

Щодо стиснення зображень, то в основі лежить облік частоти появи однакових байтів в зображенні. При цьому пікселям вихідного зображення, які зустрічаються більше число разів, відповідає код меншої довжини, а пікселям, що зустрічаються рідко — код більшої довжини. Для збору статистики потрібно два проходи по файлу — один для перегляду та збору статистичної інформації, другий — для кодування. Коефіцієнти стиснення: 1/8, 2/3, 1 [3].

Методи Хаффмана дають досить високу швидкість і задовільну якість стиснення. Застосовується запис у файл і таблиці відповідності кодованих пікселів та кодуєчих ланцюжків, приміром, в якості останнього етапу архівації в JPEG. Оскільки кожен символ кодується цілим числом біт, то при кодуванні даних з двохсимвольним алфавітом стиснення завжди відсутнє, тому незважаючи на різні ймовірності появи символів у вхідному потоці алгоритм фактично зводить їх до 1/2. Такий алгоритм реалізований у форматі TIFF.

Алгоритм Лемпеля-Зива (LZ-compression) полягає в наступному: пакувальник постійно зберігає деяку кількість останніх оброблених символів у буфері — так званого ковзаючого словника. По мірі обробки вхідного потоку новоприйняті символи потрапляють в кінець буфера, зсуваючи попередні символи і витісняючи найстаріші. Розміри буфера варіюються в різних реалізаціях кодуєчих систем. Потім, після побудови хеш таблиць, виділяють (шляхом пошуку в словнику) найдовший початкову підрядок вхідного потоку, який збігається з одним з підрядків в словнику, і видають на вихід пару (length, distance), де length — довжина знайденого в словнику підрядка, а distance — відстань від нього до вхідного підрядка (тобто фактично індекс підрядка в буфері, вирахований з його розміру). Якщо такий підрядок не знайдено, у вихідний потік просто копіюється черговий символ вхідного потоку [4]. LZ-подібні алгоритми розрізняються методом пошуку

повторюваних ланцюжків. Наприклад, в одному з варіантів цього алгоритму припускається, що у вхідному потоці йде або пара <лічильник, зміщення щодо поточної позиції>, або просто <лічильник> байт «що пропускаються» і самі значення байтів. При розархівації для пари <лічильник, зміщення> копіюються <лічильник> байт з вихідного масиву, отриманого в результаті розархівації, на <зміщення> байт раніше, а <лічильник> (тобто число рівне лічильнику) значень байт «що пропускаються» просто копіюються у вихідний масив з вхідного потоку. Даний алгоритм є несиметричним по часу, оскільки вимагає повного перебору буфера при пошуку однакових підрядків. До переваг LZ можна віднести простий алгоритм декомпресії [4].

Алгоритм Лемпеля-Зива-Велч (Lempel-Ziv-Welch -LZW) має коефіцієнти стиснення: 1/1000, 1/4, 7/5. Коефіцієнт 1/1000 досягається тільки на одноколірних зображеннях розміром більше 4 Мб. Ситуація, коли алгоритм збільшує зображення, зустрічається вкрай рідко. Стиснення забезпечується за рахунок однакових підланцюжків у потоці. Алгоритм є майже симетричним, за умови оптимальної реалізації операції пошуку рядка в таблиці. LZW універсальний — саме його варіанти використовуються в звичайних архіваторах. Він реалізований у форматах GIF, TIFF і TGA. Недолік — низький ступінь стиснення у порівнянні зі схемою двоступеневого кодування. Існує досить велике сімейство LZW-подібних алгоритмів, що розрізняються, наприклад, методом пошуку повторюваних ланцюжків [4].

Алгоритм JBIG розроблений групою експертів ISO (Joint Bi-level Experts Group) спеціально для стиснення однобітних чорно-білих зображень (наприклад, для факсів або відсканованих документів), і може застосовуватися і до 2-х, і до 4-х бітових зображень, розбиваючи їх на окремі бітові площини. JBIG дозволяє керувати порядком розбиття зображення на бітові площини, шириною смуг в зображенні, рівнями масштабування. Останнє дозволяє легко орієнтуватися в базі великих за розмірами зображень, переглядаючи спочатку їх зменшені копії. Налаштування означених параметрів дозволяє при отриманні зображення по мережі або по будь-якому іншому каналу, пропускна здатність якого мала в порівнянні з можливостями процесора, розпаковувати зображення на екрані поступово, що забезпечує оператору аналіз картинки задовго до кінця процесу розархівації. Алгоритм побудований на базі Q-кодувальника, патентом на який володіє ІВМ. Q-кодер, як і алгоритм Хаффмана, використовує для символів, що частіше з'являються, короткі ланцюжки, а для тих, що з'являються рідше, довгі. Однак на відміну від нього, використовуються і послідовності символів. Характерною особливістю JBIG є різке зниження ступеню стиснення при підвищенні рівня шумів вхідного зображення [3].

Алгоритм Lossless JPEG розроблений групою експертів з фотографії (Joint Photographic Expert Group). На відміну від JBIG, Lossless JPEG орієнтований на повнокольорові 24-бітові зображення [1]. Коефіцієнти стиснення: 1/20, 1/2, 1. JPEG включає два способи стиснення — без втрат і з втратою якості. Перший — стиснення без втрат, заснований на методі різницевого (диференціального) кодування — стисненні послідовності різниць чисел $y_i = x_i - x_{i-1}$, $i = 1, 2, \dots$, $x_0 = 0$, що характеризують помилку передбачення числа x_i . Lossless JPEG рекомендується застосовувати де необхідна побітова відповідність вихідного і розархівованого зображень.

Наведені алгоритми універсальні і охоплюють всі типи зображень, однак забезпечують занадто маленький коефіцієнт архівації — приблизно в два рази. У той же час стиснення з втратами оперують з коефіцієнтами 10-200 разів. Крім можливості модифікації зображення, одна з основних причин подібної різниці полягає в тому, що традиційні алгоритми орієнтовані на роботу з ланцюжком. Вони не враховують так звану «когерентність областей» в зображеннях — зміні кольору і структури зображення на невеликій ділянці.

Алгоритми стиснення з втратами найбільш ефективні, проте вимагають також оцінювання за означеними вище критеріями, оскільки важливим в методах репродукування

— забезпечення раціональне відтворення оригіналу (аналогового чи цифрового) з повним набором кольорових і тонових характеристик.

Тож метод усередненого блочного кодування (УБК) підлаштовує параметри кодування не під деяку усереднену характеристику всього зображення, а під локальні особливості в межах кожного блоку. Це дозволяє зберегти дрібні деталі зображень. Метод не призводить до розмивання меж, що є характерним для деяких інших алгоритмів. Метод УБК порівнянний з більшістю інших методів по ефективності стиснення даних і за обсягом обчислень, необхідних для кодування, але не має конкурентів за простотою декодування [1, 4].

JPEG — один з найпоширеніших і досить потужних алгоритмів стиснення чорно-білих і повноколірних зображень, що реалізується різними способами [1, 4]. Схема процедури стиснення зображень за стандартом JPEG наведена на рис. 2.



Рис. 2. Основні етапи процедури стиснення за стандартом JPEG

У типових зображеннях у форматі RGB є істотна кореляція між кольоровими компонентами і з точки зору стиснення зображення цей формат є завідомо надлишковим. Як відомо, в стандартах телевізійного мовлення використовується інше представлення зображень, при якому також використовуються 3 компоненти сигналу, але при цьому ці компоненти майже некорельовані один з одним. Компоненти R , G і B перетворюються в компоненту Y і дві різнокольорові компоненти U і V , формату YUV.

Коефіцієнт архівації в JPEG може варіюватися в межах від 2 до 200 разів, хоча на практиці коефіцієнт стиснення не перевищує 20-25 [3]. У JPEG є свої особливості. Найбільш відомі «ефект Гіббса» і дроблення зображення на квадрати 8×8 . Перший виявляється на границях різких меж предметів, утворюючи своєрідний «ореол». Розбиття на квадрати відбувається, коли задається занадто великий відсоток архівації для даного конкретного зображення. Недоліком методу JPEG є також те, що нерідко горизонтальні і вертикальні смуги на дисплеї абсолютно не видно, і можуть виявитися тільки при друці у вигляді муару. З цієї причини JPEG не рекомендується використовувати в поліграфії при високих коефіцієнтах стиснення [3].

Однак при застосуванні раціональних меж стиснення і алгоритмів розмиття точок растрування він може розглядатися як варіант стиснення для зменшення обсягу файлу повноколірного зображення в оригінал-макетах видань, які будуть друкуватися.

Вейвлет-стиснення — це стиснення з використанням сплесків, які визначені лише на частині області задання аргументу і можуть розглядатися як репліки єдиної базової функції, що відрізняються за масштабом та розміщенням. Така базова функція називається материнською. Ідея алгоритму полягає в тому, щоб зберігати у файл різницю між середніми значеннями сусідніх блоків у зображенні, які зазвичай приймають значення близькі до 0. Сьогодні фахівці визначають низку переваг вейвлет-стиснення в порівнянні з алгоритмами, побудованими на основі дискретного косинусного перетворення Фур'є, яке використовується в JPEG, MPEG-1, MPEG-2 [3].

Основоположним для фрактального підходу стиснення є те, що зображення реального світу мають афінну надмірність [1]. Переваги фрактального методу полягають у високих коефіцієнтах стиснення та швидкості зворотного перетворення, а також можливості подальшого структурного аналізу зображення [2]. Однак результати стиснення залежать від принципів відбору базових елементів і доменів, а коефіцієнт стиснення залежить від повторюваності базових елементів. Алгоритм орієнтований на повноколірні зображення і градації сірого. Фрактальне стиснення реалізовано у форматі FIF [2].

В даний час існує досить багато алгоритмів стиснення зображень. Основою будь-яких методів стиснення даних є використання природної надмірності вихідної інформації. Стиснення зображень здійснюється або в просторовій або в частотній областях зображення. Найбільш яскравими прикладами просторового стиснення зображень є алгоритми PCX, GIF, а частотного стиснення – JPEG. В просторовій області не можна домогтися досить великого коефіцієнта стиснення, тому в даний час більшість алгоритмів стиснення працюють в частотній області зображення, оскільки найбільш явно виражена надмірність вихідних даних проявляється при їх спектральному аналізі, як результату розкладання в деякому ортогональному базисі [1-4].

На підстав викладеного передбачається обрання таких алгоритмів стиснення за визначеними критеріями, які б найбільше відповідали параметрам відтворюваності текстово-ілюстраційної інформації в системах репродукування.

Перелік посилань

1. Гонсалес Р., Вудс Р. Цифровая обработка изображений // Пер. с англ.— М.: Техносфера, 2006. — 1072 с.
2. Крылов Е. В. Исследование вейвлетного метода сжатия изображений для повышения быстродействия веб приложений / Е.В. Крылов, В.К. Аникин, Е.В. Аникина // Зб.: Адаптивні системи автоматичного управління. — 2013. — № 2(23). — С.35-40.
3. Тропченко А.Ю., Тропченко А.А. Методы сжатия изображений, аудиосигналов и видео: Учеб. пособ. — СПб: СПбГУ ИТМО, 2009. — 108 с.
4. Сэломон Д. Сжатие данных, изображений и звука. — М.: Техносфера, 2004. — 368 с.