

УДК 621.327

Д.В. Гриньов, З.З. Закіров

Харківський університет Повітряних Сил ім. І. Кожедуба, Харків

МЕТОДИ СТИСНЕННЯ ЗОБРАЖЕНЬ В СИСТЕМАХ ЦИФРОВОЇ ОБРОБКИ ДАНИХ

В статті проведений аналіз та оцінка основних методів стиснення зображень, які використовуються в системах цифрової обробки даних. Запропонований підхід до стиснення зображень на основі методу DEFLATE з використанням етапу попередньої обробки даних та переходом до кольорорізницевої моделі уявлення кольорів.

Ключові слова: методи стиснення зображень, алгоритм стиснення без втрат, алгоритм стиснення з втратами.

Вступ

Постановка проблеми і аналіз літератури.

Основною причиною виникнення задач стиснення даних в системах цифрової обробки даних була потреба розрахункових можливостей процесорів комп'ютерів над системами збереження і передачі даних. Стиснення дозволяє зменшити навантаження на канали передачі даних або зменшити обсяг, який займають дані за рахунок їх обробки та перетворення, що як відомо створює навантаження на процесор комп'ютера, тому що стиснення даних так чи інакше пов'язано з великими обсягами математичних розрахунків [1, 2].

Здавалося, що зі збільшенням обсягів носіїв даних, зі зростом пропускної здатності мережного обладнання ця проблема відійде у минуле, проте користувачі і прогрес постійно бажають еволюції, покращення якості, ефективності у їх спілкуванні з цифровим світом, що у свою чергу призводить до збільшення обсягів потрібної для цього інформації.

Враховуючи вищесказане ми розуміємо, що стиснення даних – це минуле, сьогодення і майбутнє, широко використане та з наявністю попиту.

Тому для зменшення часу обробки і передачі інформації каналами зв'язку і зниження обсягу даних, що зберігаються на засобах ЕОМ, необхідно розробляти нові методи або модернізувати існуючі методи компактного представлення зображень. Таким чином, **метою статті** є аналіз і оцінка методів стиску зображень в системах цифрової обробки даних для вибору перспективного підходу до проблеми стиснення зображень.

Основний матеріал

У класифікації інформації зокрема можна виділити таку, що розподіляється на критичну до точності передачі даних, відносно критичну та некритичну. Критичність залежить від обсягу корисної інформації на одиницю (байт) фактичної. Залежно від

рівня критичності даних алгоритми поділяються на втратне та безвтратне стиснення.

Стиснення без втрат (англ. Lossless) – метод стиснення даних, при використанні якого закодована інформація може бути відновлена з точністю до біта. Для кожного з типів цифрової інформації, як правило, існують свої алгоритми стиснення без втрат.

Стиснення з втратами (англ. Lossy compression) – метод стиснення даних, при якому розпакований файл відрізняється від оригіналу, проте є корисним для використання. Стиснення із втратами найчастіше використовується для мультимедійних даних, особливо для потокової передачі даних та телефонії.

Алгоритми стиснення без втрат.

Алгоритм RLE. Даний алгоритм надзвичайно простий у реалізації. Групове кодування Run Length Encoding (RLE) – один із найстаріших і найпростіших алгоритмів архівації графіки. Зображення в ньому (як і в деяких алгоритмах, описаних нижче) витягається в ланцюжок байт по рядках растру. Стиснення в RLE відбувається за рахунок того, що у вихідному зображенні зустрічаються ланцюжки однакових байт. Заміна їх на пари <лічильник повторень, значення> зменшує надмірність даних.

Алгоритм розрахований на ділову графіку – зображення з більш великими областями повторюваного кольору. Для цього простого алгоритму можлива і загалом вірогідна ситуація, коли розмір даних після стиснення збільшується. Така ситуація можлива при застосуванні групового кодування до оброблених кольорових фотографій. Для того, щоб збільшити зображення у два рази, його треба застосувати до зображення, у якому значення всіх пікселів один за одним попарно не повторюються.

Характеристики алгоритму RLE. Коефіцієнти компресії: від 32 до 0,5. Клас зображень: орієнтований алгоритм на зображення з невеликою кількістю кольорів, тобто на ділову й наукову графіку. Симетричність: приблизно одиниця. Характерні риси: не

вимагає додаткової пам'яті при пакуванні і розпакуванні, а також швидко працює. Цікава особливість групового кодування полягає в тому, що ступінь архівації для деяких зображень може бути істотно підвищений всього лише за рахунок зміни порядку кольорів у палітрі зображення [1].

Алгоритм LZ. Існує досить велике сімейство LZ-подібних алгоритмів, що розрізняються, наприклад, методом пошуку повторюваних ланцюжків. Один з досить простих варіантів цього алгоритму, припускає, що у вхідному потоці ідуть або пари <лічильник, зсув щодо поточної позиції>, або просто <лічильник> "пропуску" байт і самі значення байтів. При розпакуванні для пари <лічильник, зсув> копіюються <лічильник> байт із вихідного масиву, отриманого в результаті розпакування, на <зсув> байт раніше, а <лічильник> (тобто число, що дорівнює лічильнику) значень "пропуску" байт просто копіюються у вихідний масив із вхідного потоку. Даний алгоритм є несиметричним за часом, оскільки вимагає повного перебору буфера при пошуку однакових ланок. У результаті нам складно задати великий буфер через різке зростання часу компресії. Однак потенційна побудова алгоритму, у якому на <лічильник> і на <зсув> буде виділено по 2 байти (старший біт старшого байта лічильника - ознака повтору рядка / копіювання потоку), дасть нам можливість стискати всі повторювані ланки рядка розміром до 32 Кб у буфері розміром 64 Кб.

При цьому ми одержимо збільшення розміру файлу в найгіршому разі на 32770/32768 (у двох байтах записано, що потрібно переписати у вихідний потік наступний 215 байт). Максимальний коефіцієнт стиску досягає границі 8192, оскільки максимальний стиск ми одержуємо, перетворюючи 32 Кб буфера в 4 байти, а буфер такого розміру ми отримуємо не відразу. Однак, мінімальна ланка рядка, для якої нам вигідно проводити стиск, повинна складатися в загальному випадку мінімум з 5 байт, що й визначає малу цінність даного алгоритму. До достоїнств LZ можна віднести надзвичайну простоту алгоритму декомпресії [1, 2].

Алгоритм LZW. Розглянутий нами нижче варіант алгоритму буде використовувати дерево для подання й зберігання ланцюжків. Очевидно, що це досить сильне обмеження на вид ланцюжків, і далеко не всі однакові ланки в нашому зображенні будуть використані при стисненні. Однак у пропонованому алгоритмі вигідно стискати навіть ланцюжки, що складаються з 2 байт.

Процес стиску виглядає досить просто. Зчитуються послідовно символи вхідного потоку й перевіряється, чи є в створеній таблиці рядків такий рядок. Якщо рядок є, то зчитується наступний символ, а якщо рядок відсутній, то заноситься в потік код для попереднього знайденого рядка, заноситься ря-

док у таблицю й починається пошук знову. Алгоритм LZW реалізовано в форматах GIF та TIFF.

Характеристики алгоритму LZW. Коефіцієнти компресії: від 1000 до 5/7. Стиснення в 1000 разів досягається тільки на одноколірних зображеннях розміром кратним приблизно 7 Мб. Клас зображень: 8-бітні зображення, побудовані на комп'ютері. Стискає за рахунок однакових ланок у потоці. Симетричність: майже симетрична, за умови оптимальної реалізації операції пошуку рядка в таблиці. Характерні риси: ситуація, коли алгоритм збільшує зображення, зустрічається вкрай рідко. Алгоритм LZW універсальний - саме його варіанти використовуються у звичайних архіваторах.

Класичний алгоритм Хаффмана. Один із класичних алгоритмів, відомих з 60-х років. Використовує тільки частоту появи однакових байт у зображенні. Зіставляє символам вхідного потоку, які зустрічаються більше число раз, ланцюжок біт меншої довжини. І, навпаки, що зустрічається рідко - ланцюжок більшої довжини. Для збору статистики вимагає двох проходів по зображенню.

Класичний алгоритм Хаффмана вимагає запису у файл таблиці відповідності кодуємих символів і ланцюжків, що кодуєть.

На практиці використовуються його різновиди. Так, у деяких випадках резонно або використовувати постійну таблицю, або будувати її "адаптивною", тобто в процесі пакування/розпакування. Ці прийоми рятують нас від двох проходів по зображенню й необхідності зберігання таблиці разом з файлом. Кодування з фіксованою таблицею застосовується як останній етап архівації в JPEG і в розглянутому нижче алгоритмі CCITT Group 3.

Характеристики класичного алгоритму Хаффмана. Коефіцієнти компресії: від 8 до 1. Клас зображень: практично не застосовується до зображень у чистому вигляді. Зазвичай використовується як один з етапів компресії в більш складних схемах. Симетричність: 2 (за рахунок того, що вимагає двох проходів по масиву стиснених даних). Характерні риси: єдиний алгоритм, що не збільшує розміру вихідних даних у найгіршому разі (якщо не вважати необхідності зберігати таблицю перекодування разом з файлом).

Алгоритм Хаффмана з фіксованою таблицею CCITT Group 3. Близька модифікація алгоритму використовується при стисненні чорно-білих зображень (один біт на піксел). Послідовності один за одним ідуть чорних і білих крапок у ньому замінюються числом, що дорівнює їхній кількості, які в свою чергу, стискаються по Хаффману з фіксованою таблицею. Алгоритм реалізований у форматі TIFF.

Характеристики алгоритму CCITT Group 3. Коефіцієнти компресії: від 213 до збільшення файлу в

5 разів. Клас зображень: двокольорові чорно-білі зображення, у яких переважають більші простори, заповнені білим кольором. Симетричність: близька до 1. Характерні риси: даний алгоритм надзвичайно простий у реалізації, швидкий і може бути легко реалізований апаратно [1].

JBIG. Алгоритм розроблений групою експертів ISO спеціально для стиснення однобітних чорно-білих зображень. Може застосовуватися до 2-х, 4-х бітових картинок. При цьому алгоритм розбиває їх на окремі бітові площини. JBIG дозволяє управляти такими параметрами, як порядок розбивки зображення на бітові площини, ширина смуг у зображенні, рівні масштабування. Остання можливість дозволяє легко орієнтуватися в базі більших за розмірами зображень, переглядаючи спочатку їхні зменшені копії. Розпаковуватися зображення на екрані буде поступово, як би повільно "проявляючись".

Lossless JPEG. Алгоритм розроблений групою експертів в області фотографії. На відміну від JBIG, Lossless JPEG орієнтований на повнокольорові 24-бітні або 8-бітні в градаціях сірого зображення без палітри. Він являє собою спеціальну реалізацію JPEG без втрат. Коефіцієнти стиску: від 20 до 1. Lossless JPEG рекомендується застосовувати в тих додатках, де необхідна побітова відповідність вихідного й декомпресованого зображень.

DEFLATE – це алгоритм, що використовує комбінацію алгоритму LZ і алгоритму Хаффмана.

Цей метод кодування використовує принцип ковзного вікна і враховує те що раніше зустрічалася інформація, тобто інформація, що вже відома для кодувальника й декодувальника (друге й наступне входження деякого рядка символів у повідомленні замінюються посиланнями на її перше входження).

Ковзне вікно можна представити у вигляді буфера або більш складної динамічної структури даних. Таким чином, сам процес стискаючого кодування згідно LZ нагадує написання програми, команди якої дозволяють звертатися до елементів "ковзного вікна", і замість значень стисненої послідовності вставляти посилання на ці значення в "ковзному вікні". Розмір ковзного вікна може динамічно змінюватися й становити 2 КБ, 4 КБ або 32 КБ. Слід також зазначити, що розмір вікна кодувальника може бути меншим або дорівнювати розміру вікна декодувальника, але не навпаки.

Характеристики алгоритму DEFLATE. Коефіцієнти компресії: від 1000 до 1/3. Клас зображень: будь-які зображення, бажано з великими одноколірними ланцюжками. Симетричність: близька до 1. Характерні риси: достатньо простий у реалізації, швидкий, широко використовується у програмному забезпеченні. Недоліком алгоритму є мала ефективність при кодуванні незначного обсягу даних.

Наведені вище алгоритми досить універсальні й

покривають всі типи зображень, з іншого боку – у них, на сьогодні, занадто малий коефіцієнт архівації. Використовуючи один з алгоритмів стиску без втрат, можна забезпечити архівацію зображення приблизно у два рази, хоча це і багато в чому залежить від самого зображення його насиченості, кількості кольорів складності зображення і т.п.

У той же час алгоритми стиску із втратами оперують із коефіцієнтами в середньому 10 – 200. Крім можливості модифікації зображення, одна з основних причин подібної різниці полягає в тому, що традиційні алгоритми орієнтовані на роботу з ланцюжком. Вони не враховують "когерентність областей" у зображеннях, що полягає в малій зміні кольору й структури на невеликій ділянці зображення.

Алгоритми стиснення з втратами.

Алгоритм JPEG – один з найновіших і достатньо потужних алгоритмів. Практично він є стандартом для повнокольорових зображень. Оперує алгоритм областями 8x8, на яких яскравість і колір міняються порівняно плавно. Внаслідок цього, при розкладанні матриці такої області в подвійний ряд по косинусах значущими виявляються тільки перші коефіцієнти. Стиснення в JPEG здійснюється за рахунок плавності зміни кольорів в зображенні.

Алгоритм заснований на дискретно-косинусному перетворенні (ДКП), вживаному до матриці зображення для отримання нової матриці коефіцієнтів. Для отримання попереднього зображення використовується зворотне перетворення.

ДКП розкладає зображення по амплітудах деяких частот. Таким чином, при перетворенні ми отримуємо матрицю, в якій багато коефіцієнтів або близьких, або рівних нулю. Крім того, завдяки недосконалому людському зору, можна апроксимувати коефіцієнти грубіше без помітної втрати якості зображення.

Для цього використовується квантування коефіцієнтів. У найпростішому випадку – це арифметичне побітове зсув вправо. При цьому перетворенні втрачається частина інформації, але можуть досягатися великі коефіцієнти стиснення. Процес відновлення зображення в цьому алгоритмі повністю симетричний. Метод дозволяє стискати деякі зображення в 10 – 15 разів без серйозних втрат.

Переваги: задається ступінь стиснення; вихідне кольорове зображення може мати 24 біта на точку.

Недоліки: при підвищенні ступеня стиску зображення розпадається на окремі квадрати (8x8). Це пов'язане з тим, що відбуваються більші втрати в низьких частотах при квантуванні, і відновити вихідні дані стає неможливо. Проявляється ефект Гиббса – ореоли по межах різких переходів кольорів.

Характеристики алгоритму JPEG. Коефіцієнти компресії: 2 – 200 (задається користувачем). Клас зображень: повнокольорові 24 бітні зображення або

зображення в градаціях сірого без різких переходів кольорів (фотографії). Симетричність: 1.

Фрактальний алгоритм - це алгоритм стиску зображень з втратами, заснований на застосуванні систем ітерованих функцій до зображень. Даний алгоритм відомий тим, що в деяких випадках дозволяє одержати дуже високі коефіцієнти стиснення (кращі приклади - до 1000 разів при прийнятній візуальній якості) для реальних фотографій природних об'єктів, що недоступно для інших алгоритмів стиску зображень у принципі. Через складну ситуацію з патентуванням широкого поширення алгоритм не одержав.

Основа методу фрактального кодування - це виявлення самоподібних ділянок у зображенні. Метод використовує системи доменних і рангових блоків зображення, блоків квадратної форми, що покривають все зображення. Цей підхід став основою для більшості методів фрактального кодування, застосовуваних сьогодні. Відповідно до даного методу зображення розбивається на безліч неперекритих рангових підзображень і визначається безліч неперекритих доменних підзображень. Для кожного рангового блоку алгоритм кодування знаходить найбільш підходящий доменний блок і афінне перетворення, що переводить цей доменний блок у даний ранговий блок. Структура зображення відображається в систему рангових блоків, доменних блоків і перетворень.

Основна складність фрактального стиснення полягає в тому, що для знаходження відповідних доменних блоків потрібен повний перебір. Оскільки при цьому переборі щораз повинні порівнятися два масиви, дана операція виходить досить тривалою. Порівняно простим перетворенням її можна звести до операції скалярного добутку двох масивів, однак ця операція також тривала.

Характеристики фрактального алгоритму. Коефіцієнти компресії: 2 – 2000. Клас зображень: повнокольоровий 24-бітні зображення або зображення в градаціях сірого без різких переходів кольорів. Бажано, щоб області більшої значимості були більш контрастними й різкими, а області меншої значимості – неконтрастними й розмитими. Симетричність: 100 – 100000. Характерні риси: може вільно масштабувати зображення при розархівуванні, збільшуючи його в 2 – 4 рази без появи "сходового ефекту". При збільшенні ступеня компресії з'являється "блоковий" ефект на границях блоків у зображенні [1].

Рекурсивний (хвильовий) алгоритм (wavelet). Орієнтовано алгоритм на кольорові й чорно-білі зображення із плавними переходами. Ідеальний для картинок типу рентгенівських знімків. Коефіцієнт стиску задається й варіюється в межах 5 - 100. При спробі задати більший коефіцієнт на різких границях, що особливо проходять по діагоналі, проявляється "сходовий ефект".

Ідея алгоритму полягає в тому, що зберігається у файл різниця - число між середніми значеннями сусідніх блоків у зображенні, що звичайно приймає значення, близькі до 0.









Вейвлетна компресія в сучасних алгоритмах компресії зображень дозволяє значно (до двох разів) підвищити ступінь стиснення чорно-білих і кольорових зображень при порівнянній візуальній якості стосовно алгоритмів попереднього покоління, заснованих на дискретному косинус-перетворенні, таких, наприклад, як JPEG. Для роботи з дискретними зображеннями використовується варіант вейвлет-перетворення, відомий як алгоритм Малла. Вихідне зображення розкладається на дві складові - високочастотні деталі (що складаються в основному з різких перепадів яскравості), і згладжену зменшену версію оригіналу. Це досягається застосуванням пари фільтрів, причому кожна з отриманих складових удвічі менша за вихідне зображення. Як правило, використовуються фільтри з кінцевим імпульсним відгуком, у яких пікселі, що потрапили в невелике "вікно", помножуються на заданий набір коефіцієнтів, отримані значення підсумуються, і вікно зсувається для розрахунку наступного значення на виході. Між вейвлетами й фільтрами є тісний зв'язок. Вейвети безпосередньо не фігурують в алгоритмах, але якщо ітерувати відповідні фільтри на зображеннях, що складаються з єдиної яскравої точки, то на виході будуть все чіткіше проступати вейвети.

Оскільки зображення двовимірні, фільтрація здійснюється й по вертикалі, і по горизонталі. Цей процес повторюється багаторазово, причому щораз як вхід використовується згладжена версія з попереднього кроку. Тому що зображення "деталей" складаються зазвичай з набору різких границь, і містять великі ділянки де інтенсивність близька до нуля. Якщо припустимо зневажити деякою кількістю дрібних деталей, то всі ці значення можна просто занулити. У результаті виходить версія вихідного зображення, що добре піддається стисненню. Для відновлення оригіналу знову застосовується алгоритм Малла, але з парою фільтрів, зворотної до вихідного.

Характеристики хвильового алгоритму. Коефіцієнти компресії: 2 – 200. Клас зображень: як у фрактальному і JPEG. Симетричність: 1,5. Характерні риси: при високому ступені стиску зображення розпадається на окремі блоки.

Аналіз методів стиснення зображень зведено у табл. 1. Проведемо оцінку методів стиску без втрат (без ССІТТ, як частково-комерційне рішення) на якість і швидкість стиснення. Для цього використовувався тестовий набір повнокольорових і монохромних зображень у форматі BMP 24bit. У табл. 2 приведено загальні значення.

Таблиця 1
Методи стиснення зображень

Алгоритм	За рахунок чого відбувається стиснення	Ілюстрація	Тип
RLE	Послідовне повторювання однакових значень кольорів.		Безвтратний
LZW	Повторювані однотипні послідовності кольорів.		Безвтратний
Хаффмана	Різна частота з'явлення кольорів у зображенні.		Безвтратний
DEFLATE	За рахунок різної частоти з'явлення пікселів у зображенні та їх однотипних рядків.		Безвтратний
СІТТ-3	Монохромні зображення з повторюваними елементами		Безвтратний
Wavelet	Поступові переходи кольорів		3 втра- тами
JPEG	Відсутність різких границь		3 втра- тами
Фракталь- ний	Самонагадування елементів зображень		3 втра- тами

Таблиця 2
Оцінка методів безвтратного стиснення

Алгоритм	Середній коефіцієнт	Швидкість
LZW	2,94	89%
Хаффман	1,49	73%
RLE	2,12	100%
DEFLATE	3,45	81%

З табл. 2 чітко видно, що оптимальним з розглянутих алгоритмів за співвідношенням якості-втрати є алгоритм DEFLATE. Тому доцільно розробити новий підхід до стиснення зображень на основі методу DEFLATE з використанням етапу попередньої обробки. На цьому етапі повинні підготуватися дані, які по-перше, мають повторювані фрагменти, по-друге, мають велику кількість однакових пікселів на зображенні. Етап попередньої обробки мо-

же включати наступні основні алгоритми: визначення кольору фону; визначення наявності зображення; аналіз можливості відділення фрагментів зображення; доведення фрагментів до прямокутної форми; безпосередня обробка фрагментів зображення. Додатково збільшити коефіцієнт стиснення можливо за рахунок вибору кольорової моделі. Так як алгоритм DEFLATE стискає за рахунок різної частоти з'явлення пікселів у зображенні, то доцільно використовувати кольорорізнцеву модель, в якій у двох кольорових компонентів знижений динамічний діапазон даних для фрагментів зображення з однотипними рядками [3, 4].

Висновки

Проведений аналіз основних методів стиснення зображень показав відсутніх універсальних алгоритмів, незалежних від класу зображень. Найбільш перспективним методом стиснення без втрат є DEFLATE, але в чистому вигляді його застосування не дасть вагомих результатів. Запропоновано використовувати нові підходи до стиснення зображень на основі методу DEFLATE з використанням етапу попередньої обробки та переходом до кольорорізнцевої моделі представлення кольорів.

Список літератури

1. Методы сжатия данных / В. Ватолін, А. Ратушняк, М. Смирнов, В. Юкин. – М.: ДИАЛОГ – МИФИ, 2002. – 384 с.
2. Зив Дж. Алгоритм универсального сжатия данных / Дж. Зив // Проблемы передачи информации. – 1996. – № 2. – С. 47-55.
3. Стрюк А.Ю. Цветовые модели в системах сжатия видеоинформации / А.Ю. Стрюк, К.О. Бохан // Радиоэлектроника и информатика. – 2002. – № 1. – С. 23-25.
4. Гриньов Д.В. Цветоразностная модель представления цвета в задачах сжатия изображений / Д.В. Гриньов // Системи обробки інформації: зб. наук. пр. – Х.: ХУ ПС, 2009. – Вип. 6 (80). – С. 30-34.

Надійшла до редколегії 6.03.2010

Рецензент: д-р техн. наук, проф. В.А. Краснобаєв, Харківський національний технічний університет сільського господарства ім. П. Василенка, Харків.

МЕТОДЫ СЖАТИЯ ИЗОБРАЖЕНИЙ В СИСТЕМАХ ЦИФРОВОЙ ОБРАБОТКИ ДАННЫХ

Д.В. Гринев, З.З. Закиров

В статье проведен анализ и оценка основных методов сжатия изображений, которые используются в системах цифровой обработки данных. Предложен подход к сжатию изображений на основе метода DEFLATE с использованием этапа предыдущей обработки данных и переходом к цветоразностной модели представления цветов.

Ключевые слова: методы сжатия изображений, алгоритм сжатия без потерь, алгоритм сжатия с потерями.

METHODS OF COMPRESSION OF IMAGES IN SYSTEMS OF DIGITAL DATA PROCESSING

D.V. Grinyov, Z.Z. Zakirov,

In article the analysis and an estimation of the basic methods of compression of images which are used in systems of digital data processing are lead. The approach to compression of images on the basis of method DEFLATE with use of a stage of the previous data processing and transition to colors incremental is offered to model of representation of colors.

Keywords: methods of compression of images, algorithm of compression lost-free, algorithm of compression with losses.