

УДК 621.397

¹І.В. Шелевицький, д.т.н.

²Ю.М. Барабанов, к.т.н.

²О.Г. Гуйда

²А.М. Негода

²О.М. Кушнір

АНАЛІЗ ВТРАТ ПРИ КОМПРЕСІЇ ВІДЕО ЗОБРАЖЕНЬ

¹Криворізький державний педагогічний інститут, e-mail: sheleviv@gmail.com

²Національний авіаційний університет, e-mail: brbnv@i.ua

Розглядаються проблеми втрати якості зображень в процесі стиснення та обробці відео послідовностей (зображень). Визначено основні критерії оцінки якості зображень і проведено аналіз причин втрати якості при стисненні відео послідовностей.

Ключові слова: відео, відео компресія, зображення, обробка інформації, обробка зображень.

Вступ

Значна частина задач обробки інформації й аналізу даних пов'язана із цифровими зображеннями і цифровими відеоданими, зокрема, які несуть в собі великий об'єм інформації, яка сприймається зоровою системою людини. Об'єм пам'яті, необхідний для зберігання інформації, буде пропорційний кількості пікселів у зображенні, і відповідно, кількості біт необхідних для представлення кожного пікселя. Наприклад, зображення розміром 1024×768 точок (пікселів) з глибиною кольору 24 біт займає більше 2 Мбайт пам'яті, а одна секунда відеофільму телевізійної якості вимагає близько 20 Мбайт. Такі об'єми інформації необхідно не лише зберігати, але і передавати по каналам зв'язку. Найбільш ефективним рішенням даної проблеми, є використання алгоритмів стиснення або компресії даних.

Аналіз останніх досліджень і публікацій

Всі методи стиснення інформації ґрунтуються на припущенні, що дані завжди містять елементи з надлишковою інформацією. Стиснення досягається за рахунок пошуку і кодування надлишкової інформації [6,8].

Дані в зображенні мають істотну кількість надлишкової інформації, яка може бути вилучена практично без помітних для людського ока спотворень. При цьому розрізняють два типи надлишковості:

Статична надлишковість пов'язана з кореляцією і передбачуваністю даних. Ця надлишковість може бути усунена без втрати інформації, вихідні дані при цьому можуть бути повністю відновлені. Саме на принципі статичної надлишковості ґрунтуються алгоритми стиснення зображень без втрати інформації. До таких алгоритмів відносяться:

– найбільш відомий і простий алгоритм стиснення інформації – це кодування серій повторів (Run Length Encoding – RLE) [6,8];

– алгоритм Лемпеля-Зива (LZ-compression) [8];

– алгоритм Лемпеля-Зива-Велча (Lempel-Ziv-Welch – LZW). Існує досить велике сімейство LZW – подібних алгоритмів, що відрізняються, наприклад, методом пошуку однакових послідовностей даних [8];

– алгоритм JBIG. Алгоритм розроблений групою експертів ISO (Joint Bi-level Experts Group) спеціально для стиснення одно-бітових чорно-білих зображень. В принципі може застосовуватися і до 4-х бітовим зображень [8].

– алгоритм Lossless JPEG. Алгоритм розроблений групою експертів в області фотографії (Joint Photographic Expert Group). На відміну від JBIG, Lossless JPEG орієнтований на повно-кольорові 24-бітові зображення [6,4].

Алгоритми стиснення з втратою інформації ґрунтуються на візуальній надлишковості, яку можна усунути з частковою втратою інформації, що має незначний вплив на якість відтвореної інформації. Алгоритми стиснення з втратами

– стиснення по стандарту JPEG. Стандарт JPEG (Joint Photographic Experts Group – Об'єднана експертна група по фотографії) – один з найпоширеніших і досить потужних алгоритмів стиснення зображень [7,8]. Працює він як на чорно-білих, так і на повно-кольорових

зображеннях. ;

- стиснення по стандарту JPEG2000[5];
- фрактальне стиснення зображень[7].

На практиці застосовуються два стандартних алгоритм стиснення зображень – JPEG та удосконалений стандарт JPEG2000, які використовують різні кодуєчі/декодуєчі перетворення. В основу першого покладено дискретно-косинусне перетворення, а другого – дискретно-вельветне перетворення. Фрактальний алгоритм стиснення практичного застосування не отримав.

Зробимо деякі узагальнення. Алгоритми стиснення без втрат інформації досить універсальні і покривають усі типи зображень, проте вони, забезпечують занадто малий коефіцієнт стиснення. Алгоритми без втрат, можна забезпечити коефіцієнт стиснення зображення приблизно в два рази. В той же час, стискування з втратами оперують з коефіцієнтами 10-200 разів. Окрім можливості модифікації зображення, одна з головних причин подібної різниці полягає в тому, що традиційні алгоритми орієнтовані на роботу з послідовністю даних. Вони не враховують так звану "когерентність областей" в зображеннях. Ідея когерентності областей полягає в малій зміні кольору і структури зображення на невеликій ділянці. Усі алгоритми стиснення з втратою інформації були створені пізніше спеціально для стиснення графіки і використовують цю ідею.

Успіхи цифрової відеоіндустрії (перш за все широкоформатного цифрового відео і DVD - відео) базувалися на міжнародному стандарті ISO/IEC 13818, широко відомому під аббревіатурою MPEG-2 (названий так по імені робочої групи експертів по рухомих зображеннях, що розробила цей стандарт, - Moving Picture Experts Group). Необхідність кращого стиснення відеоданих спонукало розробку подальших стандартів відеостиснення, відомих під назвами ISO/IEC 14496 Part 2 (MPEG - 4 Visual) і Рекомендація організації ITU, - E H.264/ISO/IEC 14496 Part 10 (скорочено H.264). Стандарти MPEG - 4 Visual і H.264 мають загальне походження і багато загальних рис. Вони були розроблені на основі більше ранніх стандартів стиснення. Проте вони розвивають старі стандарти в істотно різних напрямках. В основу даних стандартів покладено дискретне-косинусне перетворення.

Постановка задачі

Розглянуті алгоритми стиснення зображень, зокрема JPEG дає гарний результат при невеликому ступені стиснення. При великому ступені стиснення в відновленому зображенні помітний ефект блочності. JPEG2000 позбавлений даного недоліку, але при його використанні спостерігається розмиття ділянок з подібною структурою та поява ореолу на ділянках різкого перепаду інтенсивності – ефект Гібса.

Відповідно, алгоритми стиснення рухомих зображень (відео) використовують в основі ту ж технологію, що і алгоритм JPEG, а саме дискретне-косинусне перетворення, головним недоліком даних стандартів є розпадання зображень на блоки при високому ступені стиснення.

Таким чином, подальший розвиток алгоритмів стиснення і поліпшення методів компресії і декомпресії зображень і відео є перспективним і актуальним напрямком досліджень.

Аналіз втрат при компресії відео зображень

Процес стиснення відео інформації включає в себе наступні етапи:

- підготовка зображень до подальшої обробки (перехід до іншої системи кольоропередачі, подавлення шумів);
- апроксимація зображень;
- збереження результатів.

Найвідомішим і широко вживаним методом апроксимації відеопослідовностей є метод оцінки/розрахунку руху (Motion Estimation - ME), заснований на припущенні про схожість сусідніх кадрів відеопослідовності [2]. Оскільки стандартна відеопослідовність PAL - 25 кадрів в секунду, це припущення для звичайних фільмів виявляється достатньо правдоподібним. Відповідно до методу зображення розбивається на макроблоки стандартного розміру 16-16 пікселів, хоча деякі методи працюють з різними розмірами блоків. З послідовності береться перший кадр і наступний, що знаходиться на відстані N кадрів від нього, і кодується одним з методів кодування статичних зображень [2]. Отримані кадри називають опорними. Всі проміжні

кадри складаються з двох опорних кадрів шляхом знаходження для кожного макро-блоку кадру найбільш відповідного йому макроблоку, що передбачається, з опорних кадрів. Далі залишається зберегти лише вектор переміщення для даного макроблоку в результуючому потоці. Координати вектора переміщення визначаються таким чином: $(x-v_x; y-v_y)$, де x, y - координати верхнього лівого кута оброблюваного макроблоку, а v_x, v_y - координати верхнього лівого кута апроксимуючого макроблоку.

Дана схема кодування дозволяє досягти досить великих коефіцієнтів стискування відеопослідовностей. Кількість векторів для зображення

$$V = 2(m/b_x)(n/b_y)$$

де b_x - розмір макроблоку по горизонталі;

b_y - розмір макроблоку по вертикалі.

У свою чергу, розмір зображення в байтах

$$S = mncdb$$

де C - кількість байт для представлення одного пікселя.

Якість стисненої послідовності сильно залежить від значення N . Чим воно більше, тим гірше буде якість, оскільки відеопослідовностям в переважній більшості випадків властива динамічність. На практиці для досягнення високих ступенів стискування значення N має бути достатнє великим.

Апроксимація є не єдиним етапом в стискуванні відеопослідовності, на якому втрачається якість. Розглянуті і проаналізовані решта етапів, а також деяких особливостей методів, вживаних на даних етапах, які приводять до втрати якості.

Більшість алгоритмів використовують перетворення з кольорового простору RGB в кольоровий простір YUV (YCbCr), де Y - компоненту, що відповідає за яскравість, а $U(Cb)$ і $V(Cr)$ - компоненти, що відповідають за колір, для відділення яскравості від кольору. Людський зір чутливіший до яскравості, чим до кольору. Користуючись цим, можна ще до апроксимації зменшити кількість інформації про колір в 2 або 4 рази[1]. Спрощено переклад з колірною простору RGB в колірний простір YUV можна представити за допомогою матриці переходу:

$$\begin{pmatrix} Y \\ Cb \\ Cr \end{pmatrix} = \begin{pmatrix} 0,299 & 0,587 & 0,114 \\ 0,5 & -0,4187 & -0,0813 \\ 0,1687 & -0,3313 & 0,5 \end{pmatrix} * \begin{pmatrix} R \\ G \\ B \end{pmatrix} + \begin{pmatrix} 0 \\ 128 \\ 128 \end{pmatrix}$$

Перетворення $[Y Cb Cr]$ у $[R G B]$ (зворотно до попереднього перетворення). RGB-колір може бути обчислений безпосередньо з Y, Cb, Cr у такий спосіб:

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1,402 \\ 1 & -0,34414 & -0,71414 \\ 1 & 1,772 & 0 \end{pmatrix} * \begin{pmatrix} Y \\ Cb - 128 \\ Cr - 128 \end{pmatrix}$$

Очевидно, що при зменшенні інформації про колір знижується якість зображення.

Опорні кадри стискуються алгоритмами з втратами і спочатку вносять помилку до результуючої відеопослідовності.

Всі проміжні кадри, також як і опорні, перед кодуванням переводяться в модель представлення кольору YUV.

Не завжди вдається розташувати опорні кадри достатньо близько один до одного, це приводить до того, що вони можуть знаходитися в різних сценах відеопослідовності. Тоді апроксимація відбуватиметься лише поодиноці опорному кадру, що, природно, приведе до позитивних результатів лише на слабодинамічних сценах.

Навіть при близькому розташуванні опорних кадрів може виникнути ситуація, коли деякі деталі проміжного кадру не присутні ні в одному з опорних.

Розбиття зображення на блоки ведеться без урахування особливостей зображення. Стандартні розміри блоку 16-16 досить великі для точної роботи МЕ. Вплив останнього чинника зменшують шляхом розбиття макроблоку на блоки менших розмірів, але все таки це

не вирішує проблему в глобальному плані.

Оскільки як опорний кадр часто використовують попередній апроксимований, то всі артефакти з нього автоматично переносяться в поточний передбачений.

Найпоширенішими стандартами для стискування відеопослідовностей являються Mpeg2 і Mpeg4, в яких для усунення помилки застосовується метод компенсації руху (Motion Compensation - MC) [4]. Суть методу полягає в наступному: якщо знайдений кращий апроксимуючий блок не дає хорошої якості, то обчислюється різниця між оригінальним і знайденим блоком, далі до масиву різниці застосовується дискретне косинусне перетворення (ДКП), потім цей масив квантується і стискується методом кодування із змінною довжиною (variable length coder) [4].

Метод добре пропрацював і успішно використовується впродовж останніх років, але все таки має ряд недоліків:

- спотворюється інформація про помилку при зворотному ДКП із-за квантування коефіцієнтів, отриманих після ДКП; якщо придивитися до краю об'єктів у фільмах, закодованих за допомогою Mpeg2 або Mpeg4, легко можна побачити чітку наявність муару ("ефект Гіббса") [3];

- виявляється ефект "блоковості" (на зображенні виразно видно межі блоків) через те, що якість рядом блоків, що стоять, може сильно відрізнятися, оскільки метод працює на блоках 16-16, і для кожного блоку, незалежно від сусідніх, ухвалюється рішення про збереження помилки;

- процедура ДКП досить ресурсоемна (близько 80% часу роботи всього методу), що створює проблеми при реалізації додатків для кодування або декодування відео в реальному часі.

Після відновлення відео даних (декомпресії) втрачені елементи зображення впливають на сприйняття зображення. В кожному окремому випадку необхідно оцінити рівень втраченої інформації, оскільки візуальне сприйняття зображення відноситься до суб'єктивної категорії оцінки якості. Знайти відповідну кількісну оцінку, адекватну візуальній якості, не вдається.

Одним із простих критеріїв оцінки втрат якості зображень являється середньоквадратичне відхилення (СКВ) значення пікселів стисненого зображення до оригіналу. Нехай є два зображення: $f(x, y)$ – сигнал, і $\hat{f}(x, y)$ відновлене зображення розміром $M \times N$, тоді СКВ матиме вигляд:

$$d(x, y) = \sqrt{\frac{\sum_{x,y}^{M,N} (f(x, y) - \hat{f}(x, y))^2}{M \cdot N}}$$

Відповідно до даног критерію зображення буде сильно зіпсованим при зміні яскравості лише на 5%. В той же час зображення із снігом, різкими змінами кольору окремих точок буде визнане майже незмінним.

Іншим критерієм являється максимальне відхилення від оригіналу:

$$d(x, y) = \max_{x,y} |f(x, y) - \hat{f}(x, y)|$$

Дана міра надзвичайно чутлива до биття окремих пікселів, в зображення може міститися лише один битий піксель, і даний критерій визнає зображення сильно зіпсованим.

На практиці, зазвичай, використовується міра відношення сигналу до шуму (PSNR).

$$d(x, y) = 10 \log_{10} \frac{255^2 \cdot M \cdot N}{\sum_{x,y}^{M,N} (f(x, y) - \hat{f}(x, y))^2}$$

Дана міра аналогічна СКВ, але використання використання її більш зручне через логарифмічний масштаб шкали.

Але найкраще за все, втрату якості зображень оцінює людське око. Відмінним вважається стиснення, за якого неможливо розпізнати оригінал і декодоване зображення. Зі збільшенням ступеню стиснення стають помітними спотворення (артефакти), що характерні для всіх

алгоритмів стиснення з втратами.

Серед великої кількості критеріїв оцінки якості зображень в роботі для оцінки якості відновленого зображення вибрано СКВ та відношення сигналу до шуму (PSNR).

Висновки

Таким чином, задача ефективного стиснення відео зображень являється важливою і в той же час складною, тому що поняття ефективності стиснення вимагає максимального ступеня стиснення при максимальній якості відновленого після стиснення зображення (кадра)

Для усунення вказаними вище негативними явищами, які проявляються при стисненні відео інформації і знижують якість стисненого зображення, перспективним є вдосконалення алгоритмів компресії–декомпресії, у тому числі й несиметричних методів, у напрямку використання сплайна в якості базисної функції. Так як сплайни – кусково-поліноміальні функції, то вони легко можуть бути використані при обчисленнях. Дійсно, алгоритми для графічного зображення кривих з допомогою сплайнів та для обчислення їх поліноміальних складових надзвичайно ефективні. Усі такі алгоритми можуть використовуватись у режимі реального часу.

Список літературних джерел

1. Bhaskaran V., Konstantinos K. Image and Video Compression Standards: Algorithms and Architectures. – Boston: Kluwer Academic Publishers, 1995. – 432 p.
2. MPEG Video Compression Standard / Mitchell J.L., Pennebaker W.B., Fogg C.E., Le Gall D.J. Chapman and Hall, 1997. – 516 p.
3. ISO/IEC 14496-2 standart. Information tehnology. Coding of audio-visual objects. P. 2: Visual – 2001. – 517 p. – <http://www.iso.cn>. – 29.12.2003.
4. Kuhn P. Algorithms, Complexity Analysis and VLSI Architectures for MPEG-4 Motion Estimation. – Boston: Kluwer Academic Publishers, 1999.
5. Блаттер К. Вейвлет-анализ. Основы теории.// – М. – Техносфера. - 2006, - 279 с.
6. Ватолин Д., Ратушняк А., Смирнов М. Юкин В. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео// -М. – Диалог-МИФИ. -2003. - 384 с.
7. Гонсалес Р., Вудс Р. Цифровая обработка изображений.// Пер. с англ.- Москва.- Техносфера. – 2006. -1072 с.
8. Сэломон Д. Сжатие данных, изображений и звука// – М. – Техносфера. - 2004. – 368 с.