

Ця заявка претендує на пріоритет тимчасової заявки за реєстраційним №60/045915, поданої 7 травня 1997р.

Винахід стосується загалом способів стиснення даних, і, зокрема, удосконалення JPEG-способу стиснення зображень при його застосуванні до сейсмічних даних.

Алгоритм стандарту JPEG стиснення даних (Pennebaker and Mitchell, 1993) складається з трьох таких стадій, що здійснюються для кожного блоку 8x8 пікселів (елементів зображення) удвоєвимірному масиві:

1. Трансформування блоку 8x8 пікселів з використанням дискретної косинус-трансформації.

2. Квантування (масштабування та заокруглювання) коефіцієнтів перетворення на малі цілі числа.

3. Кодування результатів з використанням кількох бітів для зображення цілих чисел, що зустрічаються найчастіше.

Алгоритм декомпресії інвертує кожну з цих стадій у зворотному порядку. Обидва алгоритми можуть бути легко поширені на стиснення та декомпресію масивів будь-якої розмірності.

Фігура 1 зображує 2-D масив сейсмічних даних, який не було стиснено. 2-D масив 32-бітних чисел з плаваючою комою на Фігурі 1 є вирізкою у постійний момент часу, витягнутою з 3-D сейсмічної зйомки. Фігура 2 зображує збільшену підмножину цього самого масиву.

Фігура 3 зображує ту саму збільшену підмножину після стиснення та декомпресії усього 2-D масиву за JPEG-подібним алгоритмом. Коефіцієнт стиснення для масиву у цілому становив приблизно 103:1, тобто вихідний масив 32-бітних чисел з плаваючою комою містив приблизно у 103 рази більше біт, ніж стиснений масив.

Для таких великих коефіцієнтів стиснення цей JPEG-подібний алгоритм створює блочні артефакти (хибні сигнали), які видно на Фігурі 3. При менших ступенях стиснення ці розриви між блоками стають менш помітними, але усе ще можуть бути значними, особливо коли після декомпресії здійснюється подальша обробка чи інтерпретація.

Штучні об'єкти на Фігурі 3 є результатом того, що стиснення та декомпресія кожної блочної вибірки 8x8 здійснюється незалежно, без будь-якої спроби створення безперервності переходів між блоками.

Попри ці артефакти, здатність до незалежного стиснення та декомпресії таких малих підмножин даних є бажаною ознакою. Зокрема, вона забезпечує можливість доступу до якоїсь малої частини великого стисненого масиву без декомпресії масиву у цілому. Вона також дає змогу адаптувати алгоритм стиснення до просторових варіацій у амплітуді та спектрі даних. Цих ознак немає у способах стиснення, оснований на перетворенні елементарних хвиль (наприклад, Bradley et al., 1993; Wickerhauser, 1994). У даній заявці ставиться задача забезпечення цих ознак без виникнення блочних артефактів.

Одним з рішень цієї проблеми є стиснення даних з використанням блоків, які перекриваються так, що декомпресовані значення відліку можуть бути розраховані незалежно від значень, прилеглих до меж блоку. Це рішення було використано Yeو and Liu (1995) при адаптації ними алгоритму JPEG для об'ємної візуалізації 3-D медичних зображень. На жаль, використання блоків, що перекриваються, збільшує кількість блоків, які мають бути стиснені, і це збільшує час розрахунків та зменшує коефіцієнти стиснення.

Головною метою винаходу є створення удосконаленого JPEG-алгоритму стиснення даних.

Важливою метою винаходу є створення удосконаленого алгоритму стиснення даних JPEG, який може бути використаний для стиснення малих підмножин даних без блочних артефактів.

Даний винахід є способом стиснення даних, який використовує методи JPEG для дискретної косинус-трансформації та для кодування за Хаффманом квантованих коефіцієнтів перетворення. Цей спосіб є удосконаленням стандартних JPEG-способів, у першу чергу на додаткових стадіях для запобігання блочним артефактам, та при квантуванні коефіцієнтів перетворення.

Фігура 1 представляє вирізку у постійний момент часу, витягнену з 3-D сейсмічної зйомки, відображення якої є 2-D масивом 32-бітних чисел з плаваючою комою, яку не було стиснено;

Фігура 2 є збільшеною підмножиною 2-D масиву за Фіг.1;

Фігура 3 зображує дані Фігури 2 після стиснення та декомпресії із застосуванням простого варіанту JPEG алгоритму, де видно блочні артефакти, які є розривами між блочними вибірками розміром 8x8, що були стиснені незалежно;

Фігура 4 зображує дані Фігури 2 після стиснення та декомпресії із застосуванням JPEG-подібного алгоритму, який було розроблено для заглушення блочних артефактів;

Фігура 5 зображує матрицю  $C_{III}^{-1}$ , яка відповідає оберненій ДКТ-III довжиною 32, де чорні пікселі відповідають позитивним значенням; білі пікселі відповідають негативним значенням;

Фігура 6 зображує матрицю, яка відповідає оберненій блочній ДКТ-III з довжиною блоку 8;

Фігура 7 зображує колонки 11 та 19 оберненої блочної ДКТ-III матриці за Фігурою 6, де блочні артефакти при JPEG-стисненні спричинені розривами, такими як зображені між вибірками 15 та 16, і перша вибірка кожного косинуса лежить поза графіком функції та на цій фігурі внаслідок розриву функції  $b(j)$ , визначеної рівнянням (1с) нижче;

Фігура 8 зображує плавно згладжені косинуси, одержані розгорткою зрізаних косинусів, наведених на Фігурі 7, які є колонками 11 та 19 оберненої згорненої ДКТ-III матриці за Фігурою 9;

Фігура 9 зображує матрицю, яка відповідає оберненій згорненій ДКТ-III, у якій колонки цієї матриці перекриваються від блока до блока, а їх значення плавно збігаються до нуля таким чином, що зважена сума цих колонок не виявляє блочних артефактів;

Фігура 10 зображує матрицю, що відповідає операції, яка використовується для розгортки оберненої блочної ДКТ-III, де матриця за Фігурою 9 дорівнює добутку цієї матриці з матрицею, зображеною на Фігурі 6;

Фігура 11 зображує операцію розгортки, яка є попарним сполученням та заміною вибірових значень на межах ДКТ-блоків, де згортання є оберненням цієї операції;

Фігура 12 ілюструє стадії декомпресії виділеної вибірки у блоці А, для якої треба спочатку декодувати, десантувати та здійснити обернену ДКТ-III блоків А, В, С та D, а потім розгорнути чотири виділених вибірки, де операції згортання та розгортання є попарним змішуванням вибірок по межах ДКТ-блоків.

Даний винахід є удосконаленням способу стиснення даних, оснований на змішуванні даних у суміжних блоках під час компресії за JPEG-подібним алгоритмом. JPEG-подібне стиснення даних було описано у різній формі багатьма авторами (наприклад, Princen and Bradley, 1986; Malvar and Stailin, 1989; Malvar, 1990; Wickerhauser, 1994; Jawerth et al., 1995). Перевагою JPEG-подібного стиснення даних є те, що воно підвищує ступінь стиснення і лише незначно збільшує час обчислення.

Фігура 4 зображує ту ж саме підмножину з 2-D масиву Фігури 1 після стиснення та декомпресії з використанням другого JPEG-подібного алгоритму, оснований на способі за описаним нижче винаходом. Ступінь стиснення для масиву у цілому становить приблизно 112:1. Блочні артефакти у цих декомпресованих даних відсутні.

Різниця між JPEG-подібними алгоритмами за відомих рівнів техніки та новим JPEG-подібним алгоритмом за винаходом полягає у модифікації блочної дискретної косинус-трансформації, визначеної у стандарті JPEG. Ця модифікація в алгоритмі за даним винаходом 'ослаблює' блочні артефакти, зберігаючи бажані ознаки JPEG-подібного алгоритму за відомих рівнів техніки.

#### ДИСКРЕТНА КОСИНУС-ТРАНСФОРМАЦІЯ

Трансформація, що використовується у JPEG-стисненні за відомих рівнів техніки, є дискретною косинус-трансформацією, яка називається ДКТ-II (DCT-II). Згідно з винаходом використовується інша дискретна косинус-трансформація, яка називається ДКТ-III (DCT-III) (перелік дискретних косинус-трансформацій наведено у Wickerhauser, 1994, стор.84).

#### ДКТ-III

Пряма ДКТ-III визначається як

$$z(k) = \sum_{j=0}^{M-1} y(j) b(j) \sqrt{\frac{2}{M}} \cos \left[ \frac{\pi(2k+1)j}{2M} \right], \quad (1a)$$

$k = 0, 1, \dots, M-1$ ,

а відповідна обернена трансформація - як

$$y(j) = \sum_{k=0}^{M-1} z(k) b(j) \sqrt{\frac{2}{M}} \cos \left[ \frac{\pi(2k+1)j}{2M} \right], \quad (1b)$$

$j = 0, 1, \dots, M-1, 0$

де

$$b(j) = \begin{cases} \frac{1}{\sqrt{2}}, j = 0 \\ 1, \text{інакше} \end{cases}$$

Пряме та обернене перетворення можуть бути представлені як матричні множення, такі як  $z = C_{III}$  та  $y = C_{III}^{-1}z$ , де матриця  $C_{III}$  має елементи

$$C_{III}(k, j) = b(j) \sqrt{\frac{2}{M}} \cos \left[ \frac{\pi(2k+1)j}{2M} \right], \quad (2)$$

Матриця оберненої трансформації  $C_{III}^{-1}$  зображена на Фігурі 5,  $M = 32$ . Будь-який вектор з 32 дійсних чисел може бути представлений як сума колонок цієї матриці.

За одновимірним алгоритмом стиснення вектор у вибіркових значень може бути апроксимований лише кількома колонками матриці  $C_{III}^{-1}$ . Вага кожної колонки буде визначатись коефіцієнтами перетворення у векторі  $z$ . Високі ступені стиснення вимагають, щоб багато коефіцієнтів у  $z$  були нехтовно малими, майже дорівнювали нулю. Такі малі коефіцієнти будуть квантовані до нулів, що може бути ефективно закодоване кількома бітами.

Стандарт JPEG описує стиснення двовимірних зображень, а не одновимірних векторів. ДКТ-II, що використовувався у JPEG-стисненні за відомим рівнем техніки, є 2-D-перетворенням. Однак, оскільки дискретна косинус-трансформація багатовимірних даних може бути здійснена як каскад 1-D-перетворень по кожному виміру даних, для спрощення буде описано лише 1-D-перетворення.

#### Блочне ДКТ-III

Для довгих векторів даних, таких як сейсмічні спостереження, окрема дискретна косинус-трансформація навряд чи дасть багато коефіцієнтів перетворення, що будуть нехтовно малими. Тому використовується блочна ДКТ, така як JPEG, з довжиною перетворення  $M = 8$ , а вектори доповнювальних даних зводять до нулів, як потрібно, по довжині  $N$ , яка кратна 8. Матриця, що відповідає оберненій блочній ДКТ-III, зображена на Фігурі 6, для  $N = 32$  і  $M = 8$ . Крім того, що блочне перетворення є більш придатним для стиснення, воно також є більш ефективним, маючи обчислювальні витрати, які зростають лише лінійно з довжиною  $N$  векторів даних.

Як і ДКТ-II за відомим рівнем техніки, що використовувався при JPEG-стисненні, ДКТ-III має кілька корисних ознак. По-перше, ДКТ-III трансформує дійсні числа у дійсні числа, так що не потрібно складної арифметики. Крім того, як і ДКТ-II за відомим рівнем техніки, ДКТ-III є унітарним перетворенням:  $C_{III}^{-1} = C_{III}^T$  (Ця властивість витікає з рівнянь (1) вище). Пряма блочна ДКТ-III матриця, що відповідає Фігурі 6, є просто транспозицією зображеної там оберненої блочної ДКТ-III.

Іншою корисною властивістю є те, що обернена ДКТ-III еквівалентна прямій ДКТ-II:  $C_{III}^{-1} = C_{II}$ . Це співвідношення між ДКТ-III і ДКТ-II, та вибір довжини перетворення  $M = 8$ , дає змогу використовувати дуже оптимізовані ДКТ-II-алгоритми, що використовуються у JPEG-стисненні, шляхом простої підстановки прямого та оберненого алгоритмів.

Усі названі вище властивості є корисними, але чому б не використовувати просто ДКТ-II для JPEG за відомим рівнем техніки? Відповідь полягає у модифікації ДКТ-III згідно з винаходом для запобігання блочним артефактам, спричинених JPEG-стисненням.

Блочні артефакти з'являються тоді, коли для апроксимації вектора використовується лише підмножина колонок матриці, зображеної на Фігурі 6. Припустимо, наприклад, що такий вектор піддають високому ступеню стиснення шляхом використання лише двох колонок з індексами  $k = 11$  та  $k = 19$ . Як показано на Фігурі 7, будь-яка нетривіальна комбінація цих двох колонок спричинить при апроксимації розрив між вибірковими індексами  $j = 15$  та  $j = 16$ . Такі розриви створюють блочні артефакти, які видно на зображеннях, що були стиснені з використанням алгоритму JPEG.

#### Згорнена ДКТ-III

Для запобігання артефактам, спричиненим використанням для стиснення блочної ДКТ-III, блочні косинуси, зображені на Фігурі 7, заміщують згладженими косинусами, зображеними на Фігурі 8. Порівняйте матрицю повного оберненого перетворення, зображену на Фігурі 9, з матрицею оберненої блочної ДКТ-III на Фігурі 6. Число вибірок у кожному косинусі зросло до 16 (за винятком кінців), так що косинуси у суміжних блоках тепер перекриваються, і кожен косинус плавно сходиться

на нуль. Зважена сума цих косинусів не спричинить розриву, як зображено на Фігурі 7.

Перетворення, що відповідають цим згладженим косинусам, часто називають локальними косинус-трансформаціями або ортогональними трансформаціями, що перекриваються (Wickerhauser, 1994, стор.370; існує схожість між Фігурою 8 та Фігурою 11.5 у Wickerhauser). Тут ці перетворення називаються згорненими косинус-трансформаціями для відображення способу їх обчислення. Конкретно, перетворенням, що використовується у алгоритмі стиснення за винаходом, є згорнена ДКТ-III. Обернене перетворення, що відповідає матриці, зображеній на Фігурі 9, називається оберненою згорненою ДКТ-III.

Згортання є способом одержання згладжених 16-вибіркових косинусів за допомогою високооптимізованих блочних  $M = 8$  ДКТ-алгоритмів. Wickerhauser (1994, стор.103) описує цей метод як «важливе спостереження, зроблене одночасно кількома особами», і продовжує описувати його застосування у стисненні. Операція згортання, що використовується у стисненні за даним винаходом, є однією з багатьох, описаних Wickerhauser'ом, але була нав'язана роботою Jawerth and Sweldens (1995) та Jawerth et al. (1996). Останні автори описують аспекти згортання, які особливо стосуються стиснення.

Придатність застосування та ефективність згортання полягає у факті, що матриця оберненої згорнутої ДКТ-III, зображена на Фігурі 9, є добутком матриці розгортання, зображеної на Фігурі 10, та матриці оберненої блочної ДКТ-III, зображеної на Фігурі 6. Оскільки кожен ряд та колонка матриці розгортання містять не більше двох ненульових елементів, обчислювальні витрати на розгортання будуть майже несуттєвим додатком до витрат для оберненої блочної ДКТ-III.

На практиці матриці, зображені на Фігурах 6, 9 або 10, насправді ніколи не створюються. Замість цього здійснюються операції над вибірковими значеннями, що мають такий саме результат, як перемноження цих матриць. Для матриці оберненої блочної ДКТ-III на Фігурі 6 така операція є високоефективним алгоритмом прямої блочної ДКТ-II, що використовується у JPEG-стисненні. Для матриці розгортання за Фігурою 10 така операція є попарним змішуванням вибіркових значень на межах блоків при індексах  $j = 8, 16, \dots$ , як зображено на Фігурі 11.

Операція згортання є просто оберненою операцією розгортання, іншим попарним змішуванням тих самих вибіркових значень. Оскільки згортання та розгортання зосереджені на межах блоків, ці операції найічіткіше описуються за допомогою зсунутих вибіркових значень, які визначено як  $y(j) \equiv y(I M + j)$ . Символ  $I$  є індексом блока, а  $j$  - індексом вибірки усередині блока.

Тоді згортання здійснюється шляхом

$$\begin{aligned} y(j) &= f(j)x(j) + f(-j)x(-j), \\ y(-j) &= f(j)x(-j) - f(-j)x(j), \\ I &= 1, 2, \dots, N/M-1, \\ j &= 1, 2, \dots, M/2-1, \\ y(j) &= x(j), \text{ інакше,} \end{aligned} \quad (3a)$$

а розгортання - шляхом

$$\begin{aligned} x(j) &= f(j)y(j) - f(-j)y(-j), \\ x(-j) &= f(j)y(-j) + f(-j)y(j); \\ I &= 1, 2, \dots, N/M-1, \\ j &= 1, 2, \dots, M/2-1, \\ x(j) &= y(j), \text{ інакше,} \end{aligned} \quad (3b)$$

де функція згортання  $f(j)$  визначається як

$$f(j) \equiv \sin \left[ \frac{\pi}{4} \left( 1 + \frac{2j}{M} \right) \right], \quad (3в)$$

Згладжені косинуси на Фігурі 8 та матриця таких косинусів на Фігурі 9 були обчислені шляхом застосування рівнянь (3б) до колонок матриці, зображеної на Фігурі 6.

Пряма згорнена ДКТ-III вибіркових значень  $x(j)$  розраховується шляхом обчислення спочатку  $y(j)$  за допомогою рівнянь (3a), а потім, для кожного блока з 8 вибірок, обчислення  $z(k)$  за допомогою рівняння (1a). Аналогічно, обернена згорнена ДКТ-III розраховується шляхом обчислення спочатку  $y(j)$  за допомогою рівняння (1б) для кожного блока, а потім обчислення  $x(j)$  за рівняннями (3б).

Як було розглянуто Wickerhauser (1994, стор. 105), Jawerth and Sweldens (1995) та Jawerth et al. (1996), можливо багато альтернативних, але схожих операцій згортання. Операція згортання, визначена рівняннями (3), обрана за двома причинами.

По-перше, операція згортання є унітарною. Матриця розгортання, зображена на Фігурі 10, є транспозицією відповідної матриці згортання (не наведена). Для перевірки цієї властивості зазначають, що функція згортання рівняння (3в) задовольняє умові  $f^2(j) + f^2(-j) = 1$ , а потім записують операції згортання та розгортання за рівняннями (3а) та (3б) як перемноження матриць  $2 \times 2$ . Аналітичне інвертують матрицю згортання  $2 \times 2$  для перевірки того, що її обернена матриця дорівнює її транспонованій матриці, яка дорівнює матриці розгортання  $2 \times 2$ . Таким чином, уся операція згортання є унітарною, оскільки вона складається з цих  $2 \times 2$  змішувань вибірових значень на межах блоків.

По-друге, операція згортання за винаходом забезпечує те, що постійна функція, така як  $x(j) = 1$ , дає у кожному блоці коефіцієнти перетворення  $z(k)$ , які мають ненульові значення лише для  $k = 0$ , таким чином полегшуючи стиснення постійних (або таких, що повільно змінюються) даних. За термінологією Jawerth et al. (1996), пряма згорнена ДКТ-III здійснює «розв'язання констант».

Для перевірки цієї другої властивості згорненої ДКТ-III, аналітичне застосовують операцію згортання рівнянь (3а) на постійні вибірові значення  $x(j)=1$ , і перевіряють, щоб результат складав  $y(j)=\sqrt{M}C_{III}(0,j)$ , де  $C_{III}(k,j)$  визначено у рівнянні (2). Іншими словами, функцію згортання обирають таким чином, щоб результати згортання постійних вибірових значень точно співпадали (до масштабного множника  $\sqrt{M}$ ) з першим ( $k = 0$ ) косинусом ДКТ-III.

Оскільки цей косинус є ортогональним щодо усіх інших косинусів цієї трансформанти, у кожному блоці лише коефіцієнт трансформації  $k = 0$  буде ненульовим після прямої згорненої ДКТ-III. Значення кожного ненульового коефіцієнта становитиме  $z_1(0) = \sqrt{M}$ .

Точніше, ця властивість зберігається для усіх блоків, окрім першого та останнього. Хоч розв'язання констант для першого та останнього блоків може бути здійснено шляхом модифікації матриць згортання та розгортання таким чином, щоб вони не були унітарними по кутах, для цих блоків є прийнятною менша ступінь стиснення і підтримується суворі унітарність операції згортання та розгортання.

Порівняння з алгоритмом ДКТ-II, що використовується у JPEG-стисненні.

Пряма ДКТ-II, що використовується у JPEG-стисненні, забезпечує розв'язання констант без згортання, оскільки перший косинус ДКТ-II (наприклад, перший ряд матриці на Фігурі 5) є константою. Дійсно, ця властивість є причиною того, чому ДКТ-II, а не ДКТ-III, було визначено стандартом для JPEG-стиснення. Використання прийому згортання з ДКТ-II може бути розглянуто, оскільки згортання перед JPEG-стисненням та розгортання після JPEG-декомпресії можуть бути здійснені без зміни стандартного алгоритму JPEG.

На жаль, операції згортання та розгортання за рівняннями (3) не є доречними для ДКТ-II. Конкретно, рівняння (3б) не дадуть згладжених косинусів, як ті, що зображені на Фігурі 8.

Jawerth et al. (1995) описує альтернативні операції згортання та розгортання, які є придатними для ДКТ-II і які забезпечують розв'язання констант. Однак, ці операції не є унітарними. У дійсності вони є погано обумовленими, маючи тенденцію до збільшення розривів вибірових значень, які спостерігаються біля меж блоків, і тим самим знижують ефективність стиснення.

На відміну від цього, операції згортання та розгортання за рівняннями (3) є унітарними, так само як і пряма та обернена ДКТ-III за рівняннями (1). Якщо  $F$  позначає матрицю згортання, то пряма згорнена ДКТ-III може бути записана як  $z = C_{III}Fx$ , і тоді

$$z^T z = x^T F^T C_{III}^T C_{III} F x = x^T F^{-1} C_{III}^{-1} C_{III} F x = x^T x$$

$$\sum_{k=0}^{N-1} z^2(k) = \sum_{j=0}^{N-1} x^2(j), \quad (4)$$

Іншими словами, сума квадратів вибірових значень після згорненої ДКТ-III дорівнює цій сумі до згорненої ДКТ-III. Ця властивість може бути використана для обчислення спотворення декомпресованих даних, спричиненого квантуванням коефіцієнтів перетворення.

#### КВАНТУВАННЯ ТА КОДУВАННЯ

Як було описано вище, трансформація, використана у алгоритмі стиснення за винаходом, є просто операцією, оберненою до тієї, що використовується у JPEG-стисненні, але із згортанням, яке включають для зменшення блочних артефактів. Способи квантування та кодування також є адаптованими варіантами способів, що використовуються у

JPEG-стисненні. Різниця між способами JPEG та винаходом описана нижче.

#### Квантування

Оскільки JPEG-стиснення призначене для зображень, дані перед стисненням є 8-бітними або 12-бітними значеннями. (Кольорові зображення є червоно-зелено-синіми триплетами таких вибірок). Після 2-0-перетворення методом ДКТ-II 8-бітні дані можуть вимагати до 11 біт на вибірку, оскільки найбільше значення вибірки після 2-D ДКТ-II збільшується до 8 разів порівняно із значенням до трансформації. Загалом, найбільше вибіркове значення після ДКТ-II перевищує найбільше значення до трансформації в  $M^{D/2}$  разів, де D означає число трансформованих вимірів. Іншими словами,

$$|z|_{\max} \leq M^{D/2} |x|_{\max} \quad (5)$$

Ця верхня межа досягається, якщо дані точно співпадають з одним із косинусів, що використовуються у ДКТ-II, таким як постійний перший косинус  $C_{11}(k=0,j)$ .

Той самий показник  $M^{D/2}$  є застосовним також до згорненої ДКТ-III, що використовується у алгоритмі стиснення за даним винаходом. Таким чином, після трансформації цілочислових 8-бітних даних за допомогою згорненої ДКТ-III, спосіб за винаходом квантує та кодує 11-бітні цілочислові значення при 2-D-стисненні та 13-бітні цілочислові значення при 3-D-стисненні.

Стадія квантування як за алгоритмом JPEG-стиснення, так і за способом даного винаходу є операцією скейлінгу, призначеною зменшити число бітів, що кодуються, і саме це зменшення числа бітів спричинює спотворення даних, що стискаються, а потім декомпресуються. У таких способах стиснення із втратами це спотворення є прийнятним для високих коефіцієнтів стиснення.

У JPEG-стисненні коефіцієнти перетворення усередині кожного блоку квантуються по-різному, причому високі хвильові числа (просторові частоти) відображаються меншим числом бітів, ніж низькі хвильові числа. Цей залежний від хвильового числа скейлінг у JPEG-стисненні звичайно оптимізується для візуального сприйняття людиною.

У способі стиснення за винаходом усі хвильові числа квантуються однаково. Не з'являються похибки, що є залежними від хвильового числа. Однією з причин цього є те, що у сейсмічних даних великий інтерес часто представляють високі хвильові числа. Наприклад, візуалізовані підповерхневі дефекти у сейсмічних даних відповідають високим хвильовим числам. Іншою причиною є те, що сейсмічні дані часто аналізуються за комп'ютерними алгоритмами, незалежними від системи візуального сприйняття людини.

#### Локальне квантування

При стисненні 8-бітних даних зображень вибірккові значення лежать в інтервалі від -128 до +127; і можна прийняти, що низькоамплітудні блоки даних є незначущими і можуть бути безпечно квантовані до нуля при стисненні. Стандарт JPEG, зокрема, робить таке припущення, оскільки це дозволяє використовувати лише одну множину масштабних скейлінгових множників для усього зображення у цілому. Як обговорювалось вище, ці скейлінгові множники можуть змінюватись для різних коефіцієнтів (різних хвильових чисел) усередині блока, але одна й та сама множина використовується для кожного блока. У цьому сенсі JPEG-квантування є глобальним.

При стисненні 32-бітних даних з плаваючою комою краще здійснювати локальне квантування з використанням скейлінгових множників, що змінюються від блока до блока. Перед стисненням таких даних максимальна вибіркова амплітуда  $|x|_{\max}$  може бути невідомою, а зчитування кожної вибірки перед стисненням для визначення цієї величини може вимагати великих витрат. Крім того, припущення про незначущість низьких амплітуд може бути невірним; зокрема, сейсмічні дані часто потребують значної обробки для того, щоб це припущення було вірним. Таким чином, хоч усередині блока для квантування усіх коефіцієнтів трансформації використовується один скейлінговий множник, величина скейлінгового множника може змінюватись від блока до блока. Конкретно, у наведеному вище рівнянні (5)  $|z|_{\max}$  позначає максимальний коефіцієнт усередині кожного трансформованого блока, і для кожного блока розраховується окремий скейлінговий множник s.

Локальне квантування дає менші коефіцієнти стиснення (продукує більше біт-на-вибірку), ніж глобальне квантування. Очевидною причиною цього є те, що потрібні додаткові біти для зберігання скейлінгових множників квантування для кожного стисненого блока. Менш очевидною

причиною є те, що при локальному квантуванні менша кількість вибірок може дати в результаті нуль, ніж при глобальному квантуванні. Таким чином, алгоритм стиснення за винаходом дозволяє обирати локальне або глобальне квантування.

Локальне квантування не маніпулює великими динамічними діапазонами у окремому блоці. Наприклад, низькоамплітудні відбиття у необроблених сейсмічних даних можуть бути прихованими під високоамплітудними поверхневими хвилями. Усередині блоків, що містять високоамплітудні шуми, алгоритм стиснення за винаходом навіть при локальному квантуванні може дати нуль в результаті квантування низькоамплітудного сигналу, так що сигнал буде неможливо відновити шляхом обробки після декомпресії. Таким чином, слід приглушувати високоамплітудні шуми перед стисненням.

Стиснена віртуальна пам'ять

Віртуальна пам'ять створює ілюзію пам'яті, яка перевищує ту, що фізично існує. Ця ілюзія є найбільш ефективною у прикладних програмах, що використовують доступ до даних, розташованих поблизу інших даних, доступ до яких було щойно здійснено. Такі прикладні програми мають добру локальність посилань.

Прикладні програми, які працюють з 2-D або 3-D масивами, часто виявляють добру локальність посилань. Наприклад, прикладні програми для інтерпретації сейсмічних даних можуть відображати послідовні 2-D зрізи даних з 3-D масиву. За допомогою достатньо швидкого алгоритму локальної декомпресії може здійснюватись вибіркова декомпресія блоків, що містять вибірки для таких зрізів без декомпресії усього 3-D масиву у цілому. Алгоритм стиснення за винаходом є особливо корисним для таких застосувань.

Ключовою для таких застосувань є здатність стискати чи декомпресувати підмножину великого масиву без стиснення чи декомпресії масиву у цілому. Для стиснення, оснований на ДКТ-II, такого як те, що використовується у JPEG-стисненні, ця ознака легко реалізується. Конкретно, для декомпресії окремої вибірки треба лише декодувати, деквантувати та здійснити обернену ДКТ-II блока, що містить цю вибірку. При декомпресії однієї вибірки здійснюється також декомпресія усіх вибірок цього блока. За припущенням про локальність посилань обчислювальні витрати на декомпресію інших вибірок у цьому блоці не будуть марними.

Для алгоритму стиснення за винаходом, оснований на згорненій ДКТ-III, потрібна певна додаткова робота. Розглянемо 2-D-стиснення та чотири блоки вибірок розміром 8x8, зображені на Фігурі (12). Для декомпресії вибірки, що відповідає зафарбованому кружку у блоці А, треба (1) декодувати, деквантувати та здійснити обернену ДКТ-III усіх чотирьох блоків (А, Б, В та Г), і (2) розгорнути чотири вибірки, що відповідають чотирьом зафарбованим кружкам. Розгортання через межі блоків здійснюється, як проілюстровано на Фігурі 11 і описано рівнянням (3б), і виконується спочатку для одного виміру, а потім для другого. Хоч потрібні чотири блоки, при здійсненні декомпресії однієї вибірки у блоці А виконується більша частина роботи, потрібної для декомпресії суміжних вибірок. Знов-таки, припускаючи локальність посилань, ці додаткові обчислення не будуть марними.

Інша різниця між стадією квантування за винаходом і JPEG-ом витікає з нашої потреби квантувати 32-бітні дані з плаваючою комою, які мають набагато більший динамічний діапазон, ніж дані 8-бітних чи 12-бітних зображень. Для квантування величини  $z$  з плаваючою комою до цілого числа і з  $B+1$  бітами (включаючи знаковий біт), використовується такий алгоритм

$$i = \begin{cases} \lfloor z \cdot s + 1/2 \rfloor, & z \geq 0 \\ \lfloor z \cdot s + 1/2 \rfloor, & z < 0 \end{cases} \quad (6)$$

де  $s$  означає скейлінговий множник квантування. Для запобігання переповнення потрібно  $|i| < 2^B$ . Це обмеження та рівняння (6) приводять до такого рівняння для скейлінгового множника:

$$s = \frac{(2^B - 1/2)(1 - \epsilon)}{|z|_{\max}}, \quad (7)$$

де  $\epsilon$  позначає епсілон-функцію перетворення цілого числа на число з плаваючою комою, яка є найменшим позитивним числом, яке може бути віднято від 1 з використанням арифметики з плаваючою комою, щоб одержати число, яке не дорівнює 1.

#### Кодування за Хаффманом

Стандарт стиснення JPEG дозволяє два способи кодування цілих чисел, одержаних квантуванням - кодування за Хаффманом та арифметичне кодування - причому можливі також багато інших способів кодування. Хаффман-кодування JPEG використовується у алгоритмі стиснення за винаходом, оскільки воно є швидким у обчислюванні, простим у застосуванні і вільно доступним.

За одним винятком, алгоритми кодування та декодування за Хаффманом співпадають зі стандартом JPEG. Стандарт JPEG визначає спеціальне кодування коефіцієнтів трансформації DC ( $k=0$ ). Стиснення JPEG використовує той факт, що ці коефіцієнти DC для суміжних блоків часто сильно корелюють. Цей спеціальний метод обробки не використовується у способі за винаходом з двох причин: (1) він вводить міжблочні залежності, які ускладнюють стиснення та декомпресію окремих блоків, і (2) сейсмічні дані звичайно мають відносно малі коефіцієнти DC (хоч які малі, коефіцієнти DC рідко бувають нехтовними, внаслідок використання короткої трансформації  $M=8$ ). Таким чином, коефіцієнт DC кодується так само, як і інші коефіцієнти (AC).

#### ЛОКАЛЬНЕ СТИСНЕННЯ

Головною перевагою JPEG-подібного алгоритму є те, що частина багатовимірного масиву може бути стиснена чи використана без обробки усього масиву в цілому. На відміну від нього, алгоритми стиснення, основані на трансформаціях окремої хвилі (наприклад, Bradley et al., 1992; Wickerhauser, 1994), не мають цієї ознаки. Хоч стандарт JPEG явно не підтримує цієї ознаки, блочна ДКТ-II, що використовується у JPEG-стисненні, робить це можливим. Ця здатність використовується у алгоритмі стиснення за винаходом, основаному на згорненій ДКТ-III.

Кожен блок даних у масиві може розглядатись як аналог сторінки віртуальної пам'яті. Для 2-D-стиснення кожна сторінка буде містити  $64 = 8 \times 8$  вибірок для 3-D-стиснення, кожна сторінка буде містити  $512 = 8 \times 8 \times 8$  вибірок. Вибірki декомпресуються по мірі завантаження сторінок, і стискаються за хвильовими числами при їх вивантаженні. Робоча множина декомпресованих сторінок зберігається у пам'яті, у той час як більшість сторінок залишаються стисненими і зберігаються у пам'яті або на диску. (При зберіганні на диску сторінки можуть бути об'єднані для підвищення ефективності операцій вводу/виводу). Якщо робоча множина є досить великою і якщо застосування має добру локальність посилань, то обчислювальні витрати стиснення та/або декомпресії кожної сторінки будуть компенсуватись доступом до більшості вибірок на цих сторінках.

#### ПІДСУМОК

Адаптація алгоритму стиснення JPEG за винаходом уможливорює повторне використання значної частини цього алгоритму. Способи JPEG повторно використовуються для ДКТ довжиною 8 простою перестановкою прямої та оберненої трансформацій. Спосіб квантування JPEG модифікований для запобігання надання переваги обробці низьких хвильових чисел та для обробки міжблочних варіацій в амплітудах даних. Вибіркові значення також згортаються на межах блоків перед прямою ДКТ, і ці значення розгортаються після оберненої ДКТ. Це згортання та розгортання приглушує блочні артефакти, що спостерігаються на зображеннях, які було стиснено за допомогою алгоритму JPEG.

При порівнянні ефективності способу за винаходом з іншими алгоритмами двома корисними показниками ефективності є час обчислень та спотворення при визначеному ступені стиснення. Попередні результати тестування перших варіантів втілення алгоритму за винаходом обнадіюють. Час обчислень скорочується приблизно вдвічі, а рівень спотворення є майже ідентичним з алгоритмом окремої хвилі для широкого інтервалу ступенів стиснення (Diller, 1997, персональне повідомлення). Вважається, що часи обчислення для способу за винаходом є нижчими, ніж для способів, основаних на окремій хвилі (завдяки меншій кількості операцій множення та додавання та більш локальному використанню пам'яті).

#### Джерела інформації

1. Bradley, J.N., C.M., and Hopper, T., 1993, The FBI wavelet/scalar quantization standard for gray-scale fingerprint image compression: Visual Information Processing II, SPIE Proceedings, 293-304. (<ftp://ftp.c3.lanl.gov/pub/WSQ>.)
2. Jawerth, B., and Sweldens, W., 1995, Biorthogonal smooth local trigonometric bases: J. Fourier Anal. Appl., 2. (<http://cm.bell-labs.com/who/wim/papers/-papers.html>).
3. Jawerth, B., Liu, Y., and Sweldens, W., 1996, Signal compression with



smooth local trigonometric bases: <http://cm.bell-labs.com/who/wim/papers/papers.html>.

4. Malvar, H.S., and Staelin, D.H., 1989, The LOT - transform coding without blocking effects: IEEE Transactions on Acoustic, Speech, and Signal Processing, 37, №4, 553-559.

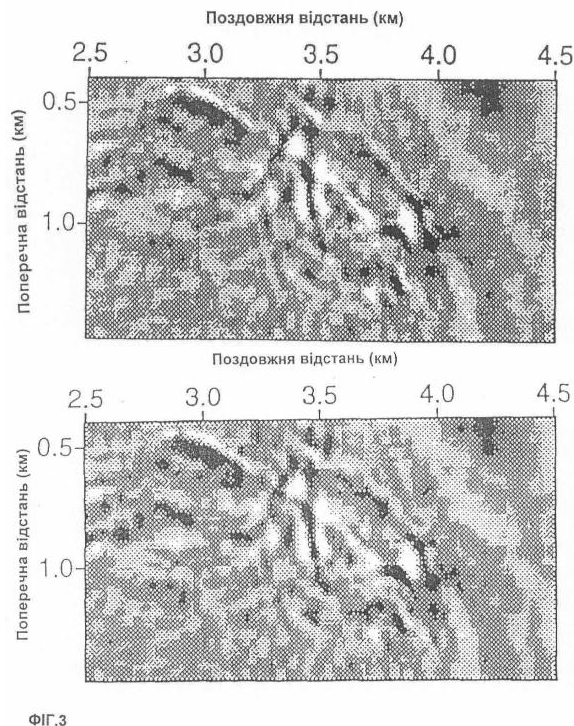
5. Malvar, M.S., 1990, Lapped transforms for efficient transform/subband coding: IEEE Transactions on Acoustic, Speech, and Signal Processing, 38, №6, 969-978.

6. Pennebaker, W.B., and Mitchell, J.L., 1993, JPEG still image data compression standard: Van Nostrand Reinhold.

7. Princen. J.P., and Bradley, A.B., 1956, Analysis/synthesis filter bank design based on time domain aliasing cancellation: IEEE Transaction on Acoustics, Speech, and Signal Processing, 34, №5, 1153-1161.

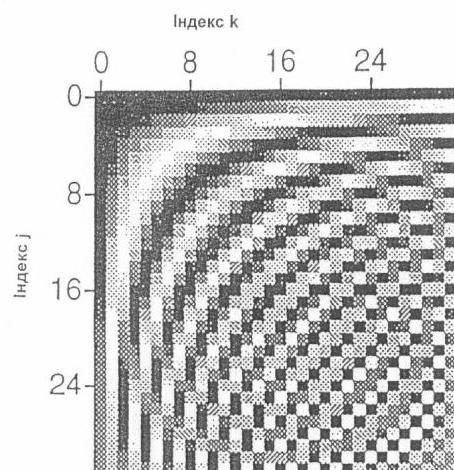
8. Wickerhauser, M.V., 1994, Adapted wavelet analysis from theory to software: A. K. Peters.

9. Yeo. B., and Liu, B., 1995, Volume rendering of DCT-based compressed 3D scalar data: IEEE Transactions on Visualization and Computer Graphics, 1, №1, 29 -43.

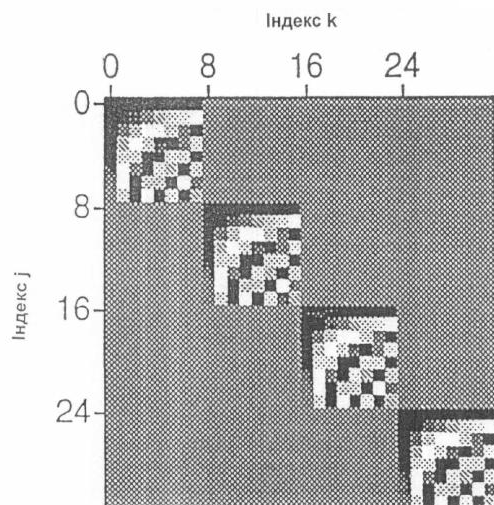




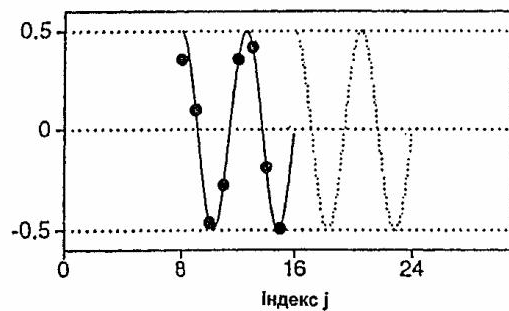
ФІГ.4



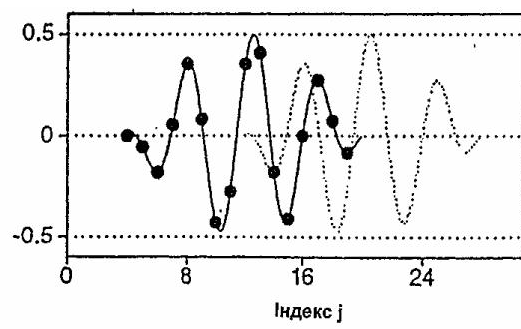
ФІГ.5



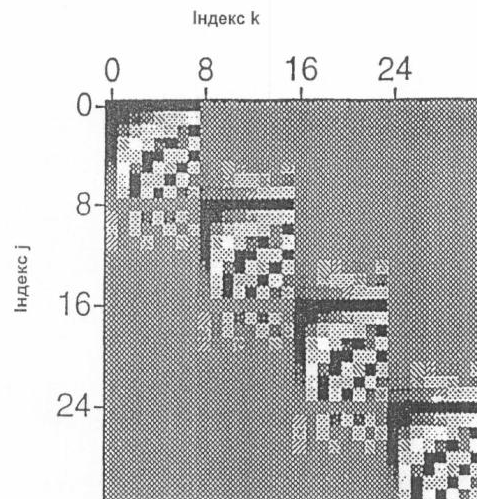
ФІГ.6



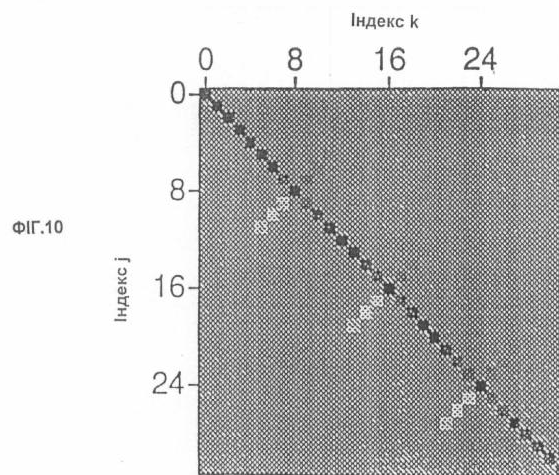
ФІГ.7



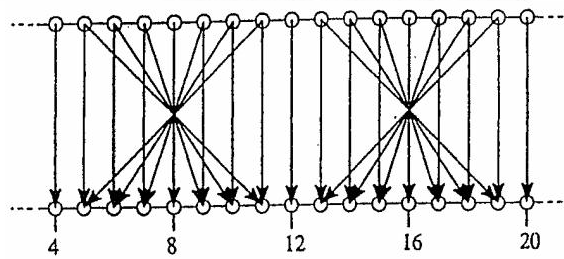
ФІГ.8



ФІГ.9



ФІГ.10



ФІГ.11

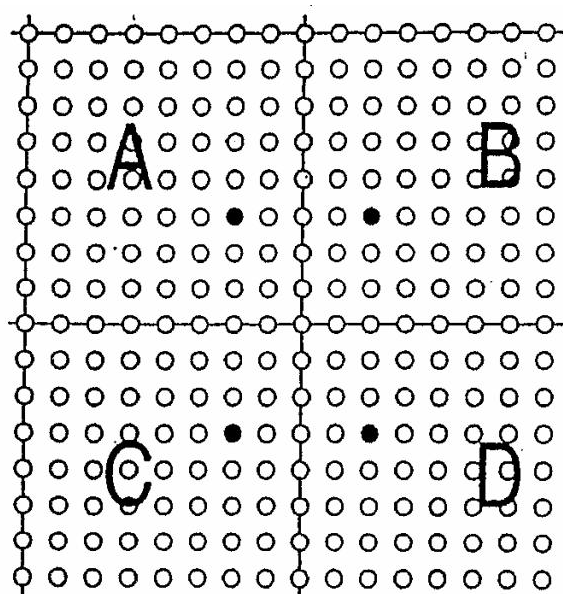


FIG.12