

*Подчашинський Ю. О., д-р. техн. наук., проф., завідувач кафедри,
Чепюк Л. О., канд. техн. наук., доцент
Державний університет «Житомирська політехніка»*

ПОРІВНЯННЯ ФРАКТАЛЬНОГО І ВЕЙВЛЕТНОГО ПІДХОДІВ ДО СТИСНЕННЯ ЗОБРАЖЕНЬ

Виконаємо порівняння двох підходів до стиснення і виявимо відносні переваги кожного. Системи, що тут наведені не є остаточними системами стиснення. Зокрема, тут відсутнє ентропійне кодування на виході фрактального і вейвлетного алгоритмів. При ентропійному кодуванні результати можуть відрізнятися від отриманих результатів. Наприклад, ті або інші алгоритми при ентропійному кодуванні можуть давати більше стиснення.

Ступінь викривлення порівнює відношення між стисненням і викривленням у алгоритмах із втратами. Оцінка визначається як середнє значення числа біт, що необхідно для представлення кожного пікселя. Вимірюється у бітах на піксель. Викривлення зазвичай вимірюються за допомогою PSNR, хоча це і не завжди зручна оцінка для прийнятної якості зображення. Криві оцінки викривлення зазвичай представляють собою залежність PSNR від біт на піксель.

Однак у літературі по фрактальному кодуванні ступінь викривлення частіше описують у термінах залежності PSNR від коефіцієнта стиснення. Можливо, причина в тому, що фрактальне кодування не пов'язане з розміром зображення у пікселях, як інші методи кодування. У подальшому також буде описуватися ступінь стиснення у термінах залежності PSNR від коефіцієнта стиснення. Розмір кодуемого зображення у фрактальних методах визначається шляхом співставлення 4 байтів кожному ранговому блоку.

Коефіцієнт стиснення визначається шляхом поділу розміру вихідного растрового зображення у байтах на кількість байтів закодованого зображення. Викривлення, що вимірюється через PSNR, визначається за допомогою декодування зображення того ж самого розміру, що і вихідне растрове зображення, і порівняння вихідного зображення з декодованим. Для вейвлет-методів розмір кодуемого зображення – це розмір двійкового нуль-дерева.

Існує принципіальна відмінність у тому, як для фрактальних і вейвлетних алгоритмів можна керувати відношенням якості зображення і коефіцієнта стиснення.

Фрактальні методи використовують схеми адаптивних квадродрев, що керуються раніше визначеною допустимою похибкою. Чим менша допустима похибка, тим краща якість декодованого зображення. Однак це призводить до збільшення числа рангових блоків, і внаслідок, до погіршення якості. З другої сторони, вейвлет-методи надають користувачу можливість керувати процесом стиснення, дозволяючи йому обирати бітові площини і/або відсоток децимації. У цьому випадку користувач заздалегідь визначає ступінь стиснення і вимушений миритися з такою якістю декодованого зображення, яка при цьому досягається. Слід пам'ятати про цю відмінність між фрактальними і вейвлетними алгоритмами стиснення при порівнянні графіків ступені викривлення для цих алгоритмів.

Крім коефіцієнта стиснення і якості зображення, існує третій параметр, а саме швидкість кодування, прийматися до уваги при порівнянні алгоритмів стиснення. Саме цей базовий фрактальний алгоритм, який досить успішно витримував конкуренцію зі сторони вейвлетних алгоритмів, повинен бути виключений із подальшого розгляду.

Фактично при одній і тій самій глибині квадродрев і допустимої похибки методи фрактального кодування дають приблизно однакову кількість рангових блоків для зображення любого розміру. Той факт, що кількість рангових блоків не збільшується зі збільшенням розміру зображення – це теж властивість, завдяки якій виникає властивість незалежності і розрізняльної здатності у фрактальному кодуванні. Фрактальне кодування шукає у зображенні інформацію. Розмір зображення не має значення.

У вейвлет-кодуванні ситуація схожа. Тут, як і для фрактальних методів, коефіцієнт стиснення приблизно у 4 рази більше для зображення 512×512 у порівнянні із зображенням 256×256 .

Збільшення коефіцієнта стиснення для великих зображень не проходить даром – кодування великих зображень потребує більшого часу.

Для фрактальних методів час кодування зображення 512×512 приблизно у два рази більше, ніж зображення 256×256 . Але це не так вже і погано, якщо врахувати, що більше зображення містить у 4 рази більше пікселів.

Для вейвлет-методу з нуль-деревом без децимації час кодування більшого зображення фактично збільшується у 5-6 разів, хоча загальний час кодування залишається порядку 15 сек. (на Intel Core i5 3,0 ГГц). При децимації час кодування значно збільшується так як додається операція сортування.