

Лабораторна робота № 7

Дослідження методів кодування та стиснення зображень з вимірювальною інформацією в КІВС

Застосування методів кодування і стиснення двовимірних сигналів до відеозображень з вимірювальною інформацією

Можливість стиснення часових послідовностей відеозображень ґрунтується на наявній надлишковості цифрових даних, що описують яскравість та колір цих відеозображень. Розрізняють декілька видів надлишковості цифрових даних відеозображень [79, 80, 119]:

- статистична надлишковість, що обумовлена властивостями відеозображень як джерела цифрових двійкових даних і може бути виключена методами статистичного кодування цих даних на основі ентропійного підходу до опису відеозображень;

- просторова та часова надлишковість, обумовлена сильними кореляційними залежностями між цифровими даними про яскравість і колір сусідніх точок цифрового відеозображення та сусідніх відеозображень у часовій послідовності;

- психофізична надлишковість, обумовлена наявністю у описі відеозображення цифрових даних (наприклад, таких, що відповідають дрібним деталям об'єктів), які не можуть бути сприйняті людиною через обмеження її зору;

Відповідно до цього розрізняють:

- стиснення відеозображень без втрат відеоінформації про її зміст та наявні об'єкти, що використовує перші два види надлишковості;

- стиснення з втратами відеоінформації, що використовує третій вид надлишковості цифрових даних відеозображення.

Стиснення без втрат відеоінформації при окремому його застосуванні до часових послідовностей відеозображень не може забезпечити потрібних значень коефіцієнту стиснення (коефіцієнту зменшення об'єму цифрових даних відеозображення, тобто відношення об'єму цифрових даних нестиснутого відеозображення до об'єму цифрових даних стиснутого відеозображення). Тому таке стиснення може використовуватися тільки як складова частина більш складних методів стиснення відеозображень з втратами частини відеоінформації.

Стиснення з втратами частини відеоінформації може забезпечити потрібні коефіцієнти стиснення часових послідовностей відеозображень у декілька десятків та сотень разів, але при цьому виникають похибки відеоінформації на відеозображеннях, відновлених після стиснення. Допустима величина цих похибок визначається потрібною візуальною якістю відновлених відеозображень при їх спостеріганні людиною (застосування стиснутих відеозображень у мультимедійних системах, телебаченні, базах даних тощо) або допустимими похибками визначення геометричних параметрів та параметрів руху об'єктів (застосування стиснутих відеозображень у інформаційно-вимірювальних системах та інших подібних технічних системах).

Використання відеозображень для вимірювання геометричних параметрів та параметрів руху об'єктів містить певне протиріччя. Воно полягає у тому, що

підвищення точності визначення цих параметрів вимагає більшої розподільчої здатності відеозображення, що приводить до збільшення кількості дискретних точок та об'єму цифрових даних відеозображення. Але збільшення цього об'єму збільшує час формування, введення в цифрову ЕОМ та накопичення цих відеозображень, що суттєво знижує швидкодію засобів вимірювання механічних величин. Для вирішення цього протиріччя і може бути застосовано стиснення часових послідовностей відеозображень, яке зменшує об'єм цифрових даних та підвищує швидкодію засобів вимірювання механічних величин при заданій точності цих вимірювань. Але при цьому потрібно контролювати величину похибок вимірювальної відеоінформації про геометричні параметри та параметри руху об'єктів.

Відповідна похибка Δ_c обумовлена кодуванням і перетворенням цифрових даних відеозображень в ПФВЗ з метою зменшення їх об'єму (стиснення) при введенні, накопиченні та зберіганні в цифровій ЕОМ. Коефіцієнт зменшення об'єму цифрових відеозображень $k_v = V_0 / V_{\text{інт}}$ ($V_0 = N \cdot M \cdot N_{\text{дк}} \cdot N_k \cdot k_{\text{п}}$ – початковий об'єм цифрових даних, $V_{\text{інт}}$ – об'єм цих даних після перетворення, N, M – ширина та висота відеозображення у дискретних точках (д.т.), $N_{\text{дк}}$ – кількість двійкових розрядів, що використовуються для кодування яскравості точок цифрового відеозображення, N_k – кількість каналів, що використовуються в цифровому кольоровому відеозображенні для передачі інформації про яскравість і колір поверхні ОБ, $k_{\text{п}} = 1$ біт – коефіцієнт пропорційності для випадку двійкового кодування цифрового відеозображення). Зв'язок між середньоквадратичним значенням цієї похибки $\sigma_{\Delta_c} = \sigma_{\Delta_{ca}} \cdot l_{\text{п}} / h_{\text{п}}$ і k_v може бути визначений на основі [79, 80] ($\sigma_{\Delta_{ca}}$ – середньоквадратичне значення похибки відтворення амплітуди відеозображення в результаті кодування, $h_{\text{п}}$ – середнє значення висоти перепадів амплітуди відеосигналу, які відповідають контурам ОБ, $l_{\text{п}}$ – середнє значення довжини проєкцій цих перепадів на задану координатну вісь в площині відеозображення).

Для цього вводиться поняття швидкості створення інформації в ПФВЗ $R_{\text{ВЗ}}(\sigma_{\Delta_{6a}})$ [79], що фактично визначає кількість біт, необхідну для передачі інформації про одну д.т. цифрового відеозображення. Цифрове відеозображення можна вважати двовимірним марковським джерелом з експоненційною кореляційною функцією і нормальним законом розподілу амплітуди. Якщо $\Delta_{6a} \ll A_{\text{max}}$, то згідно [79] маємо:

$$R_{\text{ВЗ}}(\sigma_{\Delta_{6a}}) = \frac{k_{\text{пл}}}{2} \log_2 \left(\frac{\sigma_{\text{ВЗ}}^2 \cdot (1 - \rho_x^2) \cdot (1 - \rho_y^2)}{\sigma_{\Delta_{6a}}^2} \right),$$

де A_{max} – максимальне значення амплітуди цифрових відеозображень (при кодуванні амплітуди за допомогою 8-ми розрядів двійкового коду $A_{\text{max}} = 255$ дискретних рівнів (д.р.)), $\sigma_{\text{ВЗ}}^2$ – дисперсія амплітуди відеозображення, ρ_x і ρ_y – коефіцієнти кореляції відліків у рядках і стовпцях відеозображення, $k_{\text{пл}} = 1$

біт/(д.т.) – коефіцієнт пропорційності, що визначає розмірність $R_{ВЗ}$ для випадку двійкового кодування цифрового відеозображення.

Тоді коефіцієнт зменшення об'єму напівтонового цифрового відеозображення можна визначити таким чином:

$$k_V = N_{дк} / R_{ВЗ}.$$

Визначимо на основі наведених формул необхідний коефіцієнт зменшення об'єму цифрових відеозображень. Для цифрових відеозображень ОВ типове значення дисперсії $\sigma_{ВЗ}^2 = 60$ (д.р.)², а типове значення коефіцієнтів кореляції $\rho_x = \rho_y = 0,95$. Допустима величина середньоквадратичного значення похибки дорівнює: $\sigma_{\Delta c_a} = 5,0$ д.р., $R_{ВЗ} = 0,23$ біт/(д.т.), $k_V = 34,8$ рази. Зауважимо, що для реальних відеозображень ОВ коефіцієнт k_V є в дещо меншим, ніж той, що визначений для ідеального випадку за наведеними вище формулами.

Похибки вимірювальної інформації при стисненні на основі ДКП за JPEG-алгоритмом

Розгляд похибок будемо вести на прикладі найпоширенішого методу стиснення та формату графічних файлі JPEG, що може застосовуватися як до стиснення окремих відеозображень, так і до кадрів у складі часових послідовностей відеозображень [80, 114, 118, 204, 236].

Цьому методу відповідає JPEG-алгоритм стиснення відеозображень, який представлено на рис. 5.1.

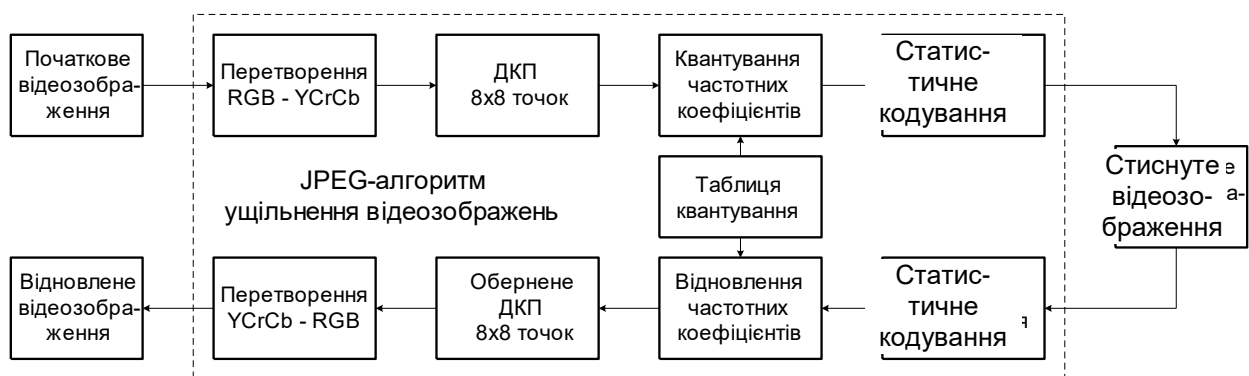


Рис. 5.1. Схема JPEG-алгоритму стиснення відеозображень

Стиснення цифрових відеозображень за JPEG-алгоритмом включає чотири етапи [118, 121, 204, 206, 207].

Перший етап – перетворення початкового відеозображення з метою розподілу інформації про яскравість і колір точок відеозображення, субдискретизація інформації про колір. Розподіл інформації про яскравість і колір точок зображення виконується шляхом перетворення кольорової схеми RGB в схему $YCrCb$ [114, 117, 207]:

$$\begin{cases} Y &= 0,299R + 0,587G + 0,114B, \\ Cb &= -0,169R - 0,331G + 0,500B + 128, \\ Cr &= 0,500R - 0,419G - 0,081B + 128, \end{cases}$$

де R , G , B – червона, зелена і синя складові частини інформації про колір дискретної точки відеозображення,

Cr , Cb – інформація про колір точки, відокремлена від інформації про яскравість (кольорорізницева інформація),

Y – інформація про яскравість цієї точки.

Зворотне перетворення виконується відповідно до формул:

$$\begin{cases} R &= Y + 1,402(Cr - 128), \\ G &= Y - 0,344(Cb - 128) - 0,714(Cr - 128), \\ B &= Y + 1,772(Cb - 128). \end{cases}$$

Субдискретизація інформації про колір полягає в збереженні тільки кожного другого або четвертого дискретного значення Cb і Cr .

У випадку застосування JPEG-алгоритму для розв'язання задач, пов'язаних із геометричними вимірюваннями на відеозображеннях, слід відзначити:

1. Існує обернена операція для відновлення цифрових даних.

2. В багатьох пристроях цифрові дані початкового відеозображення вже розділені на дані про яскравість і колір. Наприклад, у більшості відеокамер ці дані формуються окремо, їх перетворення в цифрову форму при введенні в комп'ютер також виконується окремо [122, 237, 238].

3. Для геометричних вимірювань використовуються, як правило, напівтонові зображення, отримані шляхом перетворення кольорових відеозображень.

В результаті можна вважати, що перший етап JPEG-алгоритму не приводить до виникнення похибок геометричних вимірювань, крім похибки виконання обчислень.

Другий етап JPEG-алгоритму – розподіл відеозображення на блоки (матриці) розміром $K \times K$ дискретних точок (зазвичай, $K = 8$) і виконання дискретного косинусного перетворення (ДКП) для кожного блоку. Із матриці розміром $K \times K$, що містить інформацію про яскравість дискретних точок відеозображення $f(n, m)$, утворюється матриця розміром $K \times K$, що містить частотні коефіцієнти ДКП $F(u, v)$, тобто спектр відеозображення. В результаті інформація про яскравість і колір (кожна окремо) переводиться в частотну область.

Обчислення частотних коефіцієнтів (пряме ДКП) виконується за формулою [79, 207, 236]:

$$F(u, v) = \frac{1}{\sqrt{2K}} \cdot C(u) \cdot C(v) \sum_{n=0}^{K-1} \sum_{m=0}^{K-1} f(n, m) \cdot \cos \frac{(2n+1)u\pi}{2K} \cdot \cos \frac{(2m+1)v\pi}{2K},$$

де $u = 0, 1, \dots, K-1$, $v = 0, 1, \dots, K-1$,

$n = 0, 1, \dots, K-1$, $m = 0, 1, \dots, K-1$,

$C(u)$, $C(v)$ – постійні коефіцієнти,

$$C(u) = \begin{cases} 1/\sqrt{2}, & u = 0, \\ 1, & u > 0, \end{cases} \quad C(v) = \begin{cases} 1/\sqrt{2}, & v = 0, \\ 1, & v > 0. \end{cases}$$

Відновлення інформації про яскравість точок (обернене ДКП) виконується за формулою:

$$f(n, m) = \frac{1}{\sqrt{2K}} \cdot \sum_{u=0}^{K-1} \sum_{v=0}^{K-1} C(u) \cdot C(v) \cdot F(u, v) \cdot \cos \frac{(2n+1)u\pi}{2K} \cdot \cos \frac{(2m+1)v\pi}{2K}.$$

Наведені формули фактично є визначенням прямого і оберненого ДКП, але їх застосування при обробці відеозображень не вигідно, так як потрібна велика кількість обчислювальних операцій. Існує швидкий алгоритм обчислення прямого і оберненого ДКП, який реалізується через стандартні операції множення матриць [239]:

$$F = C_1 \cdot Y \cdot C_1^T,$$

$$Y = C_1^T \cdot F \cdot C_1,$$

де C_1 – матриця постійних коефіцієнтів ДКП,

$$C_1(u, v) = \begin{cases} \frac{1}{\sqrt{K}}, & v = 0, \\ \sqrt{\frac{2}{K}} \cdot \cos \frac{(2u+1)v\pi}{2K}, & v > 0, \end{cases}$$

C_1^T – транспонована матриця C_1 .

Так як ДКП має обернене перетворення (обернене ДКП), то другий етап JPEG-алгоритму не приводить до втрат відеоінформації і виникнення похибок, крім похибок виконання обчислень.

Третій етап JPEG-алгоритму – квантування частотних коефіцієнтів, отриманих в результаті ДКП:

$$F_q(u, v) = \text{int} \left\{ \frac{F(u, v)}{Q(u, v)} + 0,5 \right\},$$

де $\text{int}\{.\}$ – операція виділення цілої частини числа,

$Q(u, v)$ – вагові коефіцієнти в таблиці квантування частотних коефіцієнтів.

Відновлення частотних коефіцієнтів при відновленні відеозображень виконується за формулою:

$$F'(u, v) = F_q(u, v) \cdot Q(u, v).$$

В процесі квантування верхні частоти можуть бути вилучені із спектра цифрового відеозображення. Це залежить від співвідношення значень елементів таблиці квантування і значень частотних коефіцієнтів. Вилучення верхніх частот приводить до викривлення форми перепадів яскравості, що відповідають контурам об'єктів вимірювань (ОВ) на відеозображеннях. В результаті виникає ще одна складова частина похибки визначення координат точок контурів об'єктів на цифровому відеозображенні. Інші частотні коефіцієнти, що залишаються в спектрі, в результаті квантування можуть дещо змінитися і це теж є джерело похибок визначення координат точок.

Четвертий етап JPEG-алгоритму – кодування частотних коефіцієнтів після квантування за методом Хаффмена або арифметичне кодування для

вилучення інформаційної надлишковості. Ці методи кодування не приводять до втрат інформації і виникнення похибок.

Для відновлення стиснутого відеозображення всі вказані етапи виконуються в зворотному порядку.

Аналіз впливу стиснення відеозображень на вимірювальну інформацію

В результаті маємо амплітудно-частотну характеристику цифрового фільтра, що відповідає JPEG-алгоритму стиснення відеозображень (рис. 5.2).

Використовуючи формулу (5.3) і отриману амплітудно-частотну характеристику, маємо:

$$\sigma_{вих}^2 = \sigma_{вх}^2 \cdot \frac{\delta_D}{\pi} \int_0^{\omega_L} 1^2 \cdot d\omega = \sigma_{вх}^2 \cdot \frac{L}{K}. \quad (5.4)$$

Таким чином, формула (5.4) дозволяє розв'язати поставлену задачу, тобто знайти дисперсію шуму на виході JPEG-алгоритму.

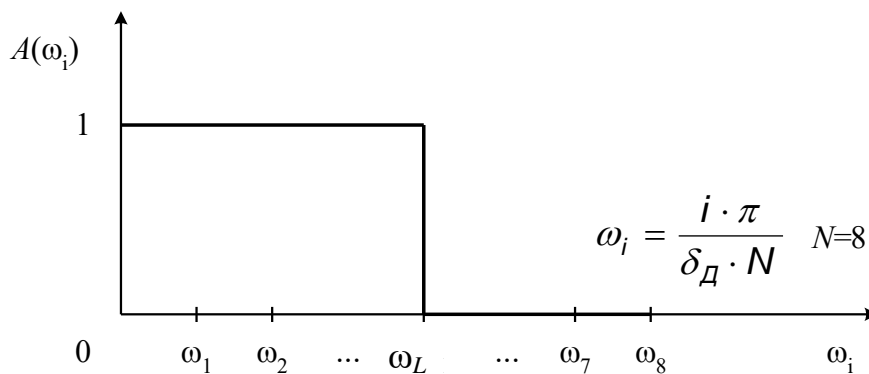


Рис. 5.2. Амплітудно-частотна характеристика JPEG-алгоритму

Так як $L = 0, 1, \dots, 7$; $K = 8$, то JPEG-алгоритм зменшує рівень шуму, який присутній на початковому зображенні.

Величина, на яку зменшується рівень шуму, залежить від співвідношення значень частотних коефіцієнтів і елементів таблиці квантування. Таким чином, перетворення вхідного шуму в JPEG-алгоритмі залежить від характеру зміни яскравості на конкретному типі відеозображень і від параметрів алгоритму.

При використанні формули (5.4) потрібно враховувати співвідношення максимальної частоти ω_{max} , що може бути присутня в рядку відеозображення, і верхньої межі смуги частот сигналу яскравості, яка визначається в процесі його формування і перетворення. Якщо верхня межа смуги частот сигналу яскравості менша за максимальну частоту ω_{max} , то потрібно визначити номер частотного коефіцієнту N' , що відповідає цій верхній межі. Тоді формула (5.4) набуває вигляду:

$$\sigma_{вих}^2 = \begin{cases} \sigma_{вх}^2 \cdot \frac{L}{K'}, & L < K', \\ \sigma_{вх}^2, & L \geq K'. \end{cases}$$

Таким чином, на основі теорії цифрової обробки сигналів визначено дисперсію шуму на виході JPEG-алгоритму. Встановлено, що при відповідному виборі параметрів JPEG-алгоритму можна зменшити рівень шуму на відеозображенні і, в результаті, зменшити похибки геометричних вимірювань.

Метод стиснення цифрових відеозображень на основі вейвлет-перетворення

На основі вейвлет-перетворення існує стандартний метод стиснення цифрових відеозображень JPEG-2000 [145, 204]. Метод стиснення відеозображень JPEG-2000 на сьогоднішній день вважається одним із кращих і по ефективності стиснення помітно перевершує найпоширеніший метод JPEG. Цей метод заснований на вейвлет-перетворенні цифрового відеозображення і обробці отриманих вейвлет-коефіцієнтів, які стискаються значно краще, ніж відліки початкового відеозображення. Разом з тим вейвлет-перетворення і обробка коефіцієнтів досить складні та вимагають великої обчислювальної потужності комп'ютера. Тому JPEG-2000 застосовують в тих випадках, коли потрібно одержати як можна кращу якість стиснутого відеозображення, а час виконання стиснення є не критичним. В зв'язку з цим великий інтерес представляють способи, що дозволяють поліпшити часові характеристики вейвлет-перетворень.

Суть алгоритму перетворення цифрового відеозображення на основі методу JPEG-2000 полягає в наступному. До початкового цифрового відеозображення застосовується дискретне вейвлет-перетворення із заданим числом рівнів декомпозиції. Дискретне вейвлет-перетворення – це фільтрація рядка (або стовпця) початкового відеозображення двома вейвлет-фільтрами. Це КІХ-фільтри: низькочастотний фільтр L та високочастотний фільтр H .

Оскільки після фільтрації частотний діапазон відеосигналу скорочується у два рази, профільтровані дані, відповідно до теореми Котельникова, можна прорідити удвічі, тобто виключити всі парні коефіцієнти. Потім з отриманих компонент складається новий рядок, як показано на рис. 4.1.

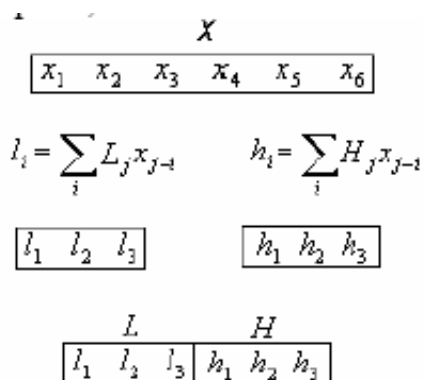


Рис. 4.1. Вейвлет-перетворення одновимірного сигналу

Для двовимірного сигналу існує окреме двовимірне вейвлет-перетворення. Але його досить складно реалізувати і на практиці його звичайно замінюють двома одновимірними. Така заміна не погіршує стиснення, але значно спрощує фільтрацію. В одновимірному випадку спочатку фільтрують

рядки, потім стовпці. Відеозображення при цьому розділяється на 4 частини (рис. 4.2).

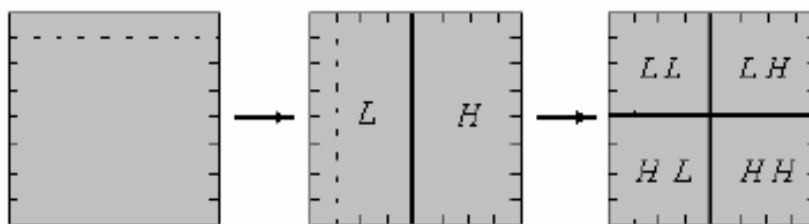


Рис. 4.2. Фільтрація двовимірного відеосигналу цифрового відеозображення

Потім до частини, що профільтована тільки низькочастотними фільтрами, застосовують ту ж саму процедуру. Такий алгоритм зветься алгоритмом Малла (рис. 4.3).

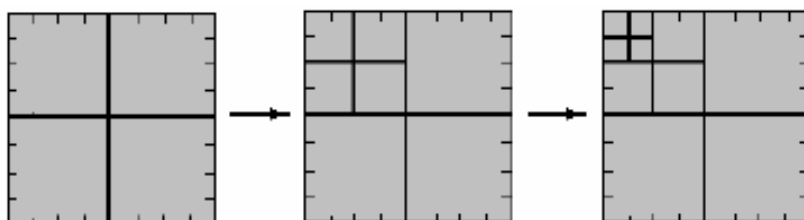


Рис. 4.3. Алгоритм Малла

Результат вейвлет-перетворення представлено на рис. 4.4.



Рис. 4.4. Результат вейвлет-перетворення

В даному випадку наведено дворівневе вейвлет-перетворення. Частина відеозображення у верхньому лівому куті профільтована тільки низькочастотними фільтрами та представляє собою згладжену зменшену копію оригіналу. Вся інша область відеозображення профільтована як низькочастотними, так і високочастотними фільтрами. На ній помітні великі ділянки близьких до нуля вейвлет-коефіцієнтів і окремі яскраві елементи, які означають різкі перепади яскравості на початковому відеозображенні. Ця частина дуже добре підлягає для стиску навіть звичайними алгоритмами.

Власне, одержання великої кількості нульових коефіцієнтів і є основною метою вейвлет-перетворення, саме ж перетворення стиску не дає.

Оптимальним за критерієм «якість-складність реалізації» є дворівневе вейвлет-перетворення. Значне збільшення числа рівнів не приводить до помітного підвищення ефективності стиску, а складність алгоритму істотно зростає.

Після перетворення здійснюється квантування коефіцієнтів. Саме на етапі квантування виникають основні інформаційні втрати, і саме за рахунок квантування можливо істотне зменшення об'єму подання відеозображення. Квантування може бути або рівномірним скалярним, або більш складним, наприклад, векторним. У випадку використання рівномірного скалярного квантування його параметри можуть мінятися залежно від квантуемого діапазону.

Далі застосовується послідовне кодування всіх частотних блоків LL2, LH2 тощо. Кодування здійснюється арифметичним методом за допомогою імовірнісної моделі і проводиться в наступній послідовності:

1) визначається розрядність коефіцієнтів блоку за формулою

$$N_{\max} = \text{ceil}(\log_2(C_{\max})),$$

де C_{\max} - максимальний за модулем коефіцієнт.

2) у всіх коефіцієнтів кодуються біти на місці N_{\max} , потім $N_{\max} - 1$ тощо.

Такий спосіб називається кодуванням поверхів бітів (bit plane), в алгоритмі JPEG-2000 кожний поверх бітів кодується в три стадії, які називаються significance pass, refinement pass, cleanup pass (рис. 4.5). Ймовірнісна модель для арифметичного кодування будується на тому факті, що великі та малі коефіцієнти блоку не перемішані рівномірно, а концентруються в окремих областях, і це дозволяє судити про значення кодуемого біта, розглядаючи величину найближчих вейвлет-коефіцієнтів.

Відмінною рисою JPEG-2000 є наявність так званого методу оптимізації довжини коду блоків (optimized truncation). Біти послідовно кодуються від значення N_{\max} аж до молодшого біта (LSB, least significant bit), причому закодовані дані відразу не посилають у вихідний потік, а зберігаються в пам'яті. Після того, як закодовані всі блоки, у пам'яті залишається набір закодованих даних (для дворівневого вейвлет-перетворення цей набір складається з 7 елементів). Задача методу – оптимізувати довжину елементів таким чином, щоб забезпечити необхідний ступінь стиснення та одночасно так підібрати співвідношення довжин елементів, щоб середньоквадратична похибка стиснення була найменшою.

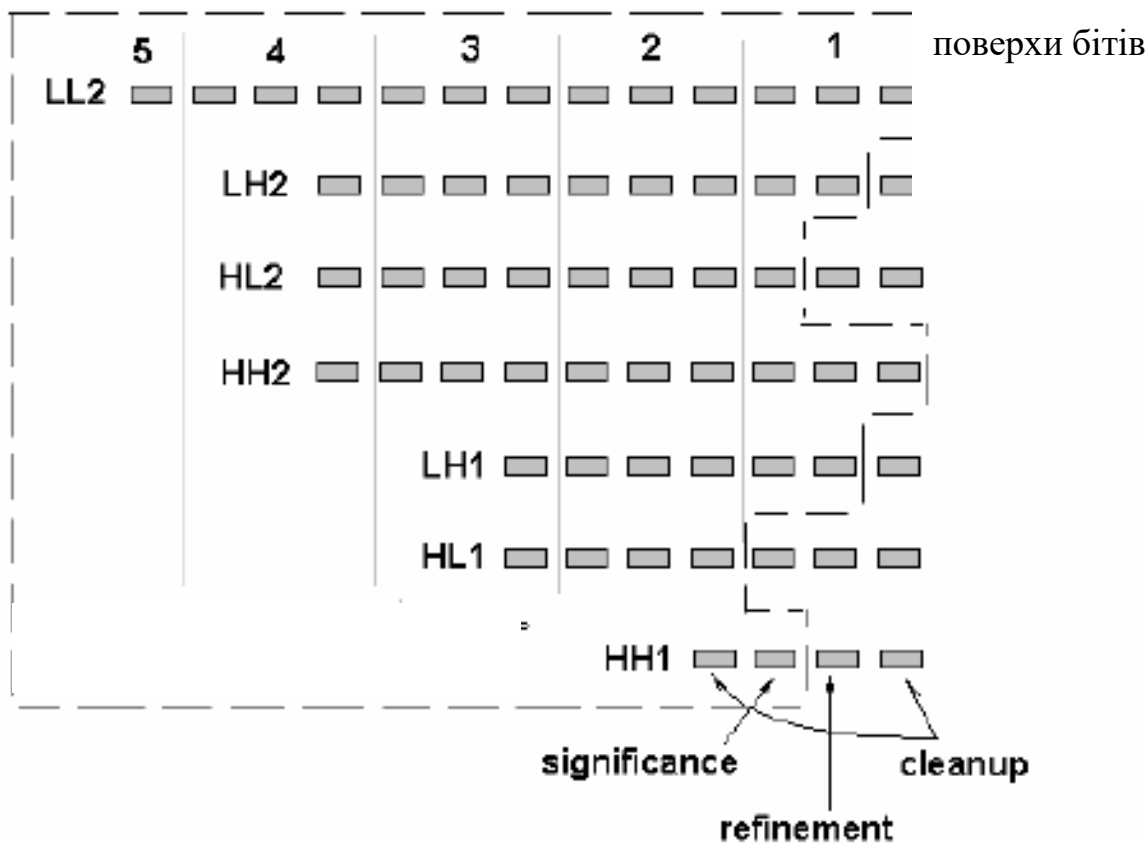


Рис. 4.5. Оптимізація довжин закодованих даних

У пам'яті залишаються тільки ті дані, які забезпечують кращу якість стиснуто відеозображення при заданому ступені стиснення.

Подання інформації, отриманої в результаті вейвлет-перетворення, дуже зручно тим, що воно забезпечує можливість одержання приблизних копій відеозображення без здійснення повного зворотного перетворення. Зворотне перетворення здійснюється в порядку, зворотному до порядку прямого перетворення. Роблячи обмежене число зворотних декомпозицій (об'єднань частотних діапазонів) та вважаючи, що всі не залучені в перетворення частотні діапазони містять винятково нульові елементи, ми легко можемо одержати або копію відеозображення в зменшеному масштабі, або відновлене відеозображення, але більш низької якості в порівнянні з відеозображенням, отриманим в результаті повного зворотного перетворення. З огляду на той факт, що блоки вейвлет-коефіцієнтів кодуються незалежно один від одного, ми одержуємо можливість часткового декодування не тільки на рівні перетворення, але і на рівні інтерпретації коду. Для одержання приблизної копії відеозображення досить декодувати лише частину інформації, а потім зробити часткове зворотне перетворення. Таким чином, формат зберігання відеозображення забезпечує масштабованість як за розподільчою здатністю, так за якістю.

Іншою важливою перевагою даного методу є можливість доступу до окремих елементів відеозображення без повного декодування його представлення. Забезпечується така можливість по-перше, розбивкою початкового відеозображення на не перехресні області, які кодуються як окремі

відеозображення, а по-друге, представленням коду окремої області у вигляді частин (шарів), кожна з яких є сумарним кодом коефіцієнтів, що відповідають деякій області. Відзначимо, що шари у свою чергу діляться на так звані пакети, що містять код блоків коефіцієнтів на різних рівнях декомпозиції. Для того, щоб декодувати будь-яку область відеозображення, досить визначити її розташування в площині відеозображення і які шари, що відносяться до цієї області, містять код блоків коефіцієнтів, необхідних для відновлення даної області.

Безумовно, таке представлення відеозображення не може бути вигідним з погляду ефективності стиснення. Дійсно, зі зменшенням розміру структурних елементів ефективність стиснення дещо знижується. Стандарт методу JPEG-2000 у цьому випадку залишає вибір: з одного боку, ми маємо можливість одержувати інформаційні представлення, що дозволяють досить швидко вилучати і редагувати частини відеозображення, з іншого боку, стандарт не перешкоджає створенню інформаційних представлень, ефективних по об'єму стиснутих даних.

Кольорові цифрові відеозображення містить у собі відразу кілька компонентів. Найчастіше, вони складаються з трьох компонентів: червоного *R*, зеленого *G* і синього *B* кольору. Більшої ефективності стиснення можна досягти у випадку, коли кодуємо компоненти представлені в яскраво-колірній формі. Для перетворення відеозображення зі стандартного колірної представлення *RGB* у яскраво-колірне представлення *YCrCb* (яскравість та кольорорізницеві сигнали). Стандартом методу JPEG-2000 передбачені обчислювальні процедури такого перетворення.

Для забезпечення завадостійкості інформаційного представлення та зручності доступу до даних в стандарті JPEG-2000 передбачена система маркерів і маркерних сегментів. Маркери відіграють роль розмежувачів інформаційного потоку; маркерні сегменти містять у собі параметри фрагментів інформації, обмежених маркерами. Дані, що починаються з маркера, як правило, можуть бути коректно проінтерпретовані без якої-небудь додаткової інформації, що забезпечує можливість часткового відновлення відеозображення, що було пошкоджено. Введення елементів завадостійкості забезпечує використання методу стиснення в телекомунікаційних додатках.

Результати чисельного моделювання та експериментальних досліджень вейвлет-стиснення цифрових відеозображень

Стиснення тестових відеозображень (рис. 4.6) виконувалося за допомогою безкоштовної версії програми ACDSee. Використовувався метод стиснення JPEG 2000, заснований на wavelet-перетворенні.

Було обрано декілька рівнів стиснення з коефіцієнтами стиснення (відношенням об'єму цифрових даних початкового відеозображення до об'єму стиснутих даних) 100:1, 50:1, 25:1, 10:1, 8:1, 4:1, 2:1 та 1:1. При стисненні 100:1 візуально якість відеозображення значно погіршувалась, тому такий режим був прийнятий як режим максимального стиснення, який варто використовувати для подібних відеозображень. Стиснення 1:1 було прийнято до розгляду для

оцінки максимально можливої якості, яка зберігається при вейвлет-перетворення цифрових відеозображень.

Стиснення проводилося для оригінальних кольорових відеозображень та для їх напівтонових (відтінки сірого) еквівалентів.

Для відновлених після стиснення відеозображень було виконано аналіз на наявність похибок відтворення кольору та геометричних параметрів об'єктів.

Похибки відтворення кольору об'єктів

Похибки відтворення кольору можна розглядати як загальну міру відхилення значень кольору стиснутого відеозображення відносно початкового:

$$\Delta = \sum_{i=1}^3 k_i \Delta_i,$$

де $i=1,2,3$ – номер каналів для передачі кольору (в даному випадку: R (червоний), G (зелений), B (синій)), відповідно $k_1 = 0.3$, $k_2 = 0.59$, $k_3 = 0.11$. Значення Δ_i обчислюються за формулою:

$$\Delta_i = \sqrt{\frac{1}{M \cdot N} \sum_{x=1}^M \sum_{y=1}^N (Y_{vost.}(x, y, i) - Y_{orig.}(x, y, i))^2},$$

де $Y_{vost.}(x, y, i)$, $Y_{orig.}(x, y, i)$ – показники яскравості кольору i у точці (x, y) відновленого та початкового відеозображень відповідно.

Похибки відтворення кольору об'єктів обчислювалися для кольорових відеозображень та їх напівтонових еквівалентів окремо. Основною відмінністю між цими підрахунками було те, що у кольорового відеозображення 3 кольорових канали, а у напівтонового 1.

Текст програми для підрахунку похибки відтворення кольору об'єктів для кольорового відеозображення:

```
file_original = 'st5_512x512.bmp'
file_vost = 'st5_512x512_4.bmp'
inf_original = imfinfo(file_original);
inf_vost = imfinfo(file_vost);
image_original = imread(file_original);
image_vost = imread(file_vost);

koef = [0.3 0.59 0.11]

delta = 0;

for color_channel = 1:3
    channel_original = double(image_original(:,:,color_channel));
    channel_vost = double(image_vost(:,:,color_channel));
    channel_quad_summ = 0;
    for i = 1:inf_original.Width
        for j = 1:inf_original.Height
            channel_quad_summ = channel_quad_summ +
                (channel_vost(i,j) - channel_original(i,j))^2;
        end
    end
    deltaT =
        sqrt(channel_quad_summ/(inf_original.Width*inf_original.Height))
    delta = delta + koef(color_channel) * deltaT;
```

end

В змінні `file_original`, `file_vost` заносимо назви відеозображень, які досліджуємо. В даному прикладі це `st5_512x512.bmp` (файл початкового відеозображення), `st5_512x512_4.bmp` (файл відеозображення, стиснутого з коефіцієнтом 4:1).

```
inf_original = imfinfo(file_original);  
inf_vost = imfinfo(file_vost);
```

В `inf_original` та `inf_vost` заносимо інформацію про відеозображення, використовуючи функцію `imfinfo()`. До цієї інформації належать: назва файлу, дата останнього редагування, розмір файлу у байтах, формат відеозображення, версія формату, висота, ширина, глибина кольору (кількість байт для зберігання кольору одного пікселя) та тип кольору.

```
image_original = imread(file_original);  
image_vost = imread(file_vost);
```

За допомогою функції `imread()` читаємо безпосередньо самі початкове та відновлене відеозображення і записуємо отримані дані у змінні `image_original`, `image_vost`.

Далі в циклі знаходимо значення виразу для похибки

$\sum_{x=1}^M \sum_{y=1}^N (A_{vost.}(x, y, i) - A_{orig.}(x, y, i))$ та поміщуємо його у змінну `channel_quad_summ`.

```
for i = 1:inf_original.Width  
    for j = 1:inf_original.Height  
        channel_quad_summ = channel_quad_summ +  
(channel_vost(i,j) - channel_original(i,j))^2;  
    end  
end
```

Тут `channel_vost` та `channel_original` – це двовимірні масиви, елементами яких є значення для кожного пікселя відеозображення відповідного кольору (червоного, зеленого чи синього). Їх розмірність рівна розмірам відеозображення.

```
deltaT =  
sqrt(channel_quad_summ / (inf_original.Width * inf_original.Height))
```

В змінну `deltaT` записуємо значення похибки для поточного кольору.

```
delta = delta + koef(color_channel) * deltaT;
```

В `delta` записуємо загальне значення похибки відтворення кольору об'єктів.

Приклад результат виконання програми:

```
>>  
file_original =  
st5_512x512.bmp  
file_vost =  
st5_512x512_4.bmp  
koef =  
    0.3000    0.5900    0.1100
```

```
deltaR =
    0.9533
deltaG =
    0.8644
deltaB =
    1.0118
delta =
    0.9072
```

Три значення deltaR , deltaG , deltaB відповідають похибкам для червоного, зеленого та синього кольорів відповідно, delta – загальне значення похибки відновленого кольорового відеозображення, deltaY – значення похибки відновленого напівтонового відеозображення.

За допомогою цієї програми було знайдено значення похибки відтворення кольору об'єктів для усіх п'яти відеозображень (див. рис. 4.6), стиснутих з усіма наведеними коефіцієнтами. Далі було побудовано графіки залежності похибки відтворення кольору об'єктів від коефіцієнту стиснення (рис. 4.7).

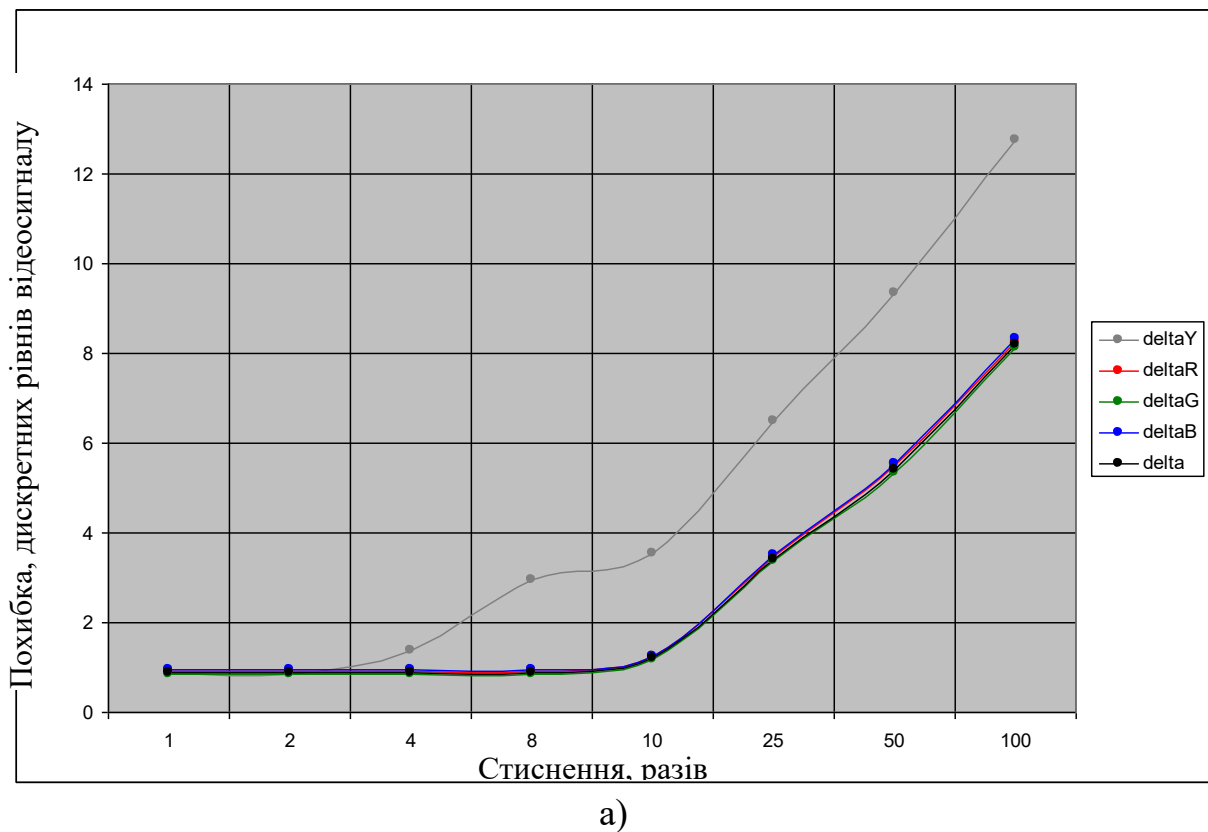
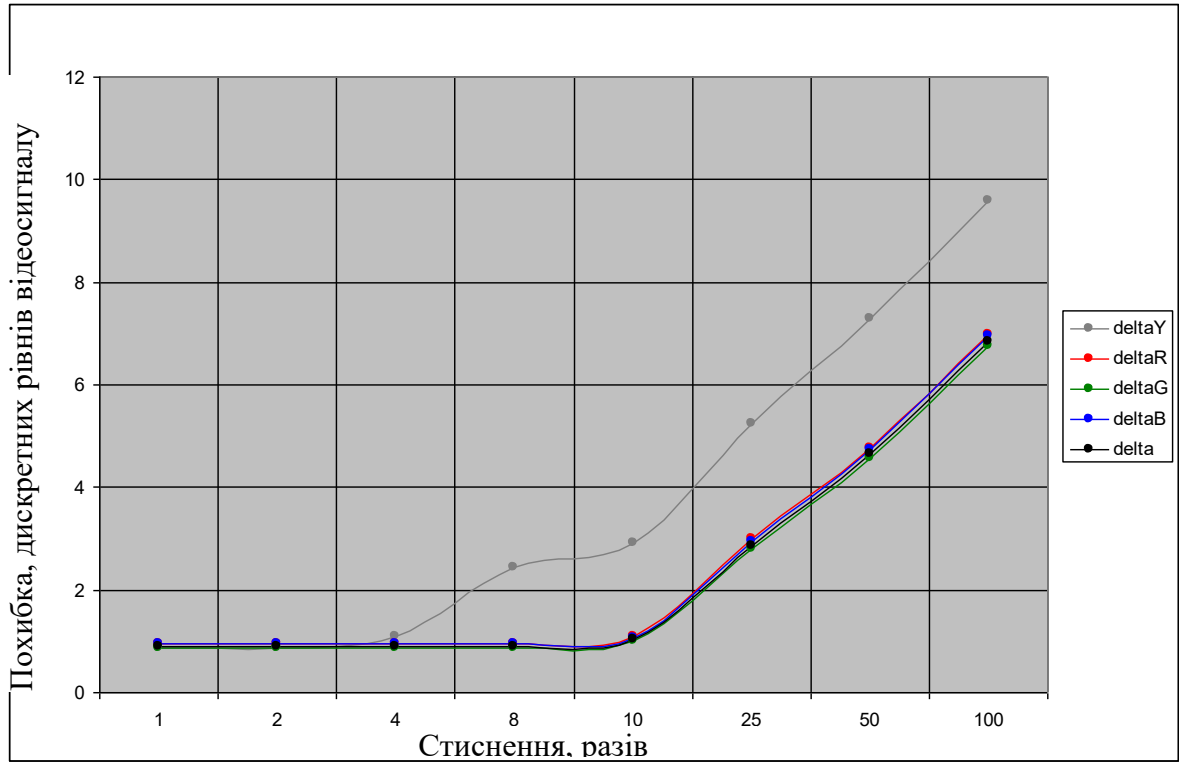
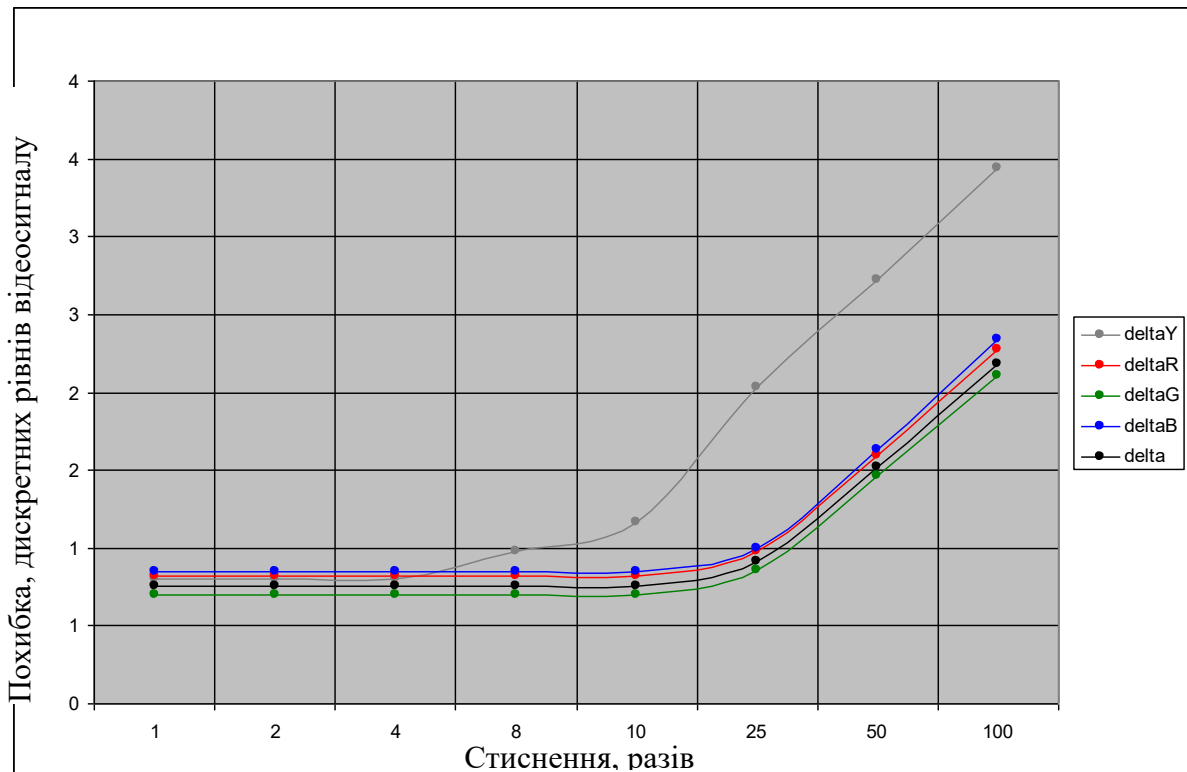


Рис. 4.7. Похибки відтворення кольору об'єктів в залежності від величини стиснення: а) відеозображення №1; б) відеозображення №2; в) відеозображення №3; г) відеозображення №4; д) відеозображення №5; deltaR , deltaG , deltaB – відповідають похибкам для червоного, зеленого та синього кольорів відповідно, delta – загальне значення похибки відновленого кольорового відеозображення, deltaY – значення похибки відновленого напівтонового відеозображення

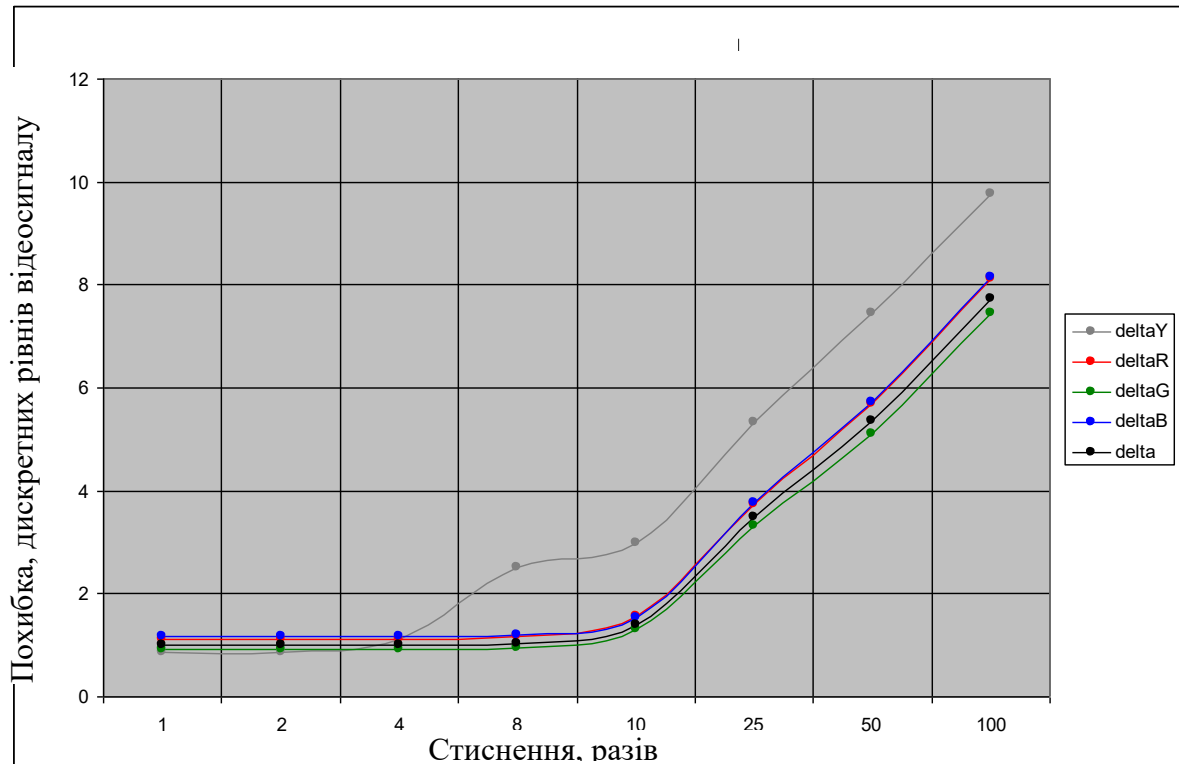


б)

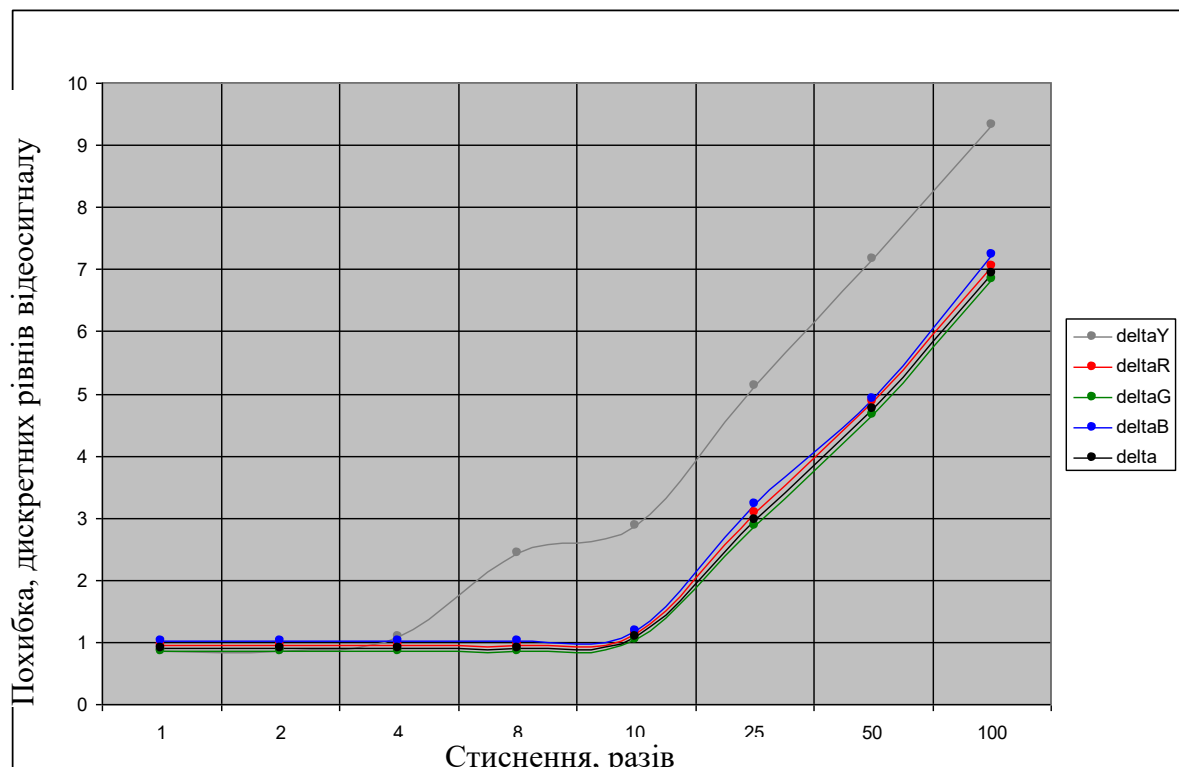


в)

Рис. 4.7 (продовження)



Г)



Д)

Рис. 4.7 (продовження)

На основі отриманих результатів можна зробити висновок про ефективність застосування вейвлет-перетворення та відповідного методу стиснення до цифрових відеозображень з вимірювальною інформацією про ГП

та параметри руху об'єктів. При стисненні кольорових та напівтонових відеозображень з коефіцієнтом 10:1 похибки майже непомітні. Прийнятним коефіцієнтом стиснення для вимірювань ГП є значення 50:1.

Похибки координат об'єктів на відновлених відеозображеннях

Оцінка похибок координат основана на аналізі ділянок відеозображення, дискретні точки яких мають однакові ознаки. Ці ділянки відповідають об'єктам вимірювань та їх отримано шляхом сегментації тестових відеозображень. Ознакою сегментації була величина яскравості дискретних точок цифрового відеозображення. Для відеозображень №1, №2, №4, №5 (див. рис. 4.6) точки, для яких яскравість належить діапазону [128, 255] дискретних рівнів відеосигналу, вважались світлими, інші з проміжку [0, 127] дискретних рівнів – темними. Для відеозображення №3 такими діапазонами були [51, 255] та [0, 50], що пояснюється низькою яскравістю самого відеозображення.

Міра координатного викривлення – це величина, яка знаходиться за формулою:

$$\Delta_{coord.} = \sqrt{\frac{1}{L} \sum_{i=1}^L (R_{compr.}(i) - R_{orig.}(i))^2},$$

де $R_{compr.}(i)$, $R_{orig.}(i)$ – координат контурних точок об'єкта, L – сумарна кількість контурних точок.

В даному випадку було обрано 20 точок, 10 з яких досліджувались на зміщення координати x , інші 10 – на зміщення координати y (рис. 4.8).

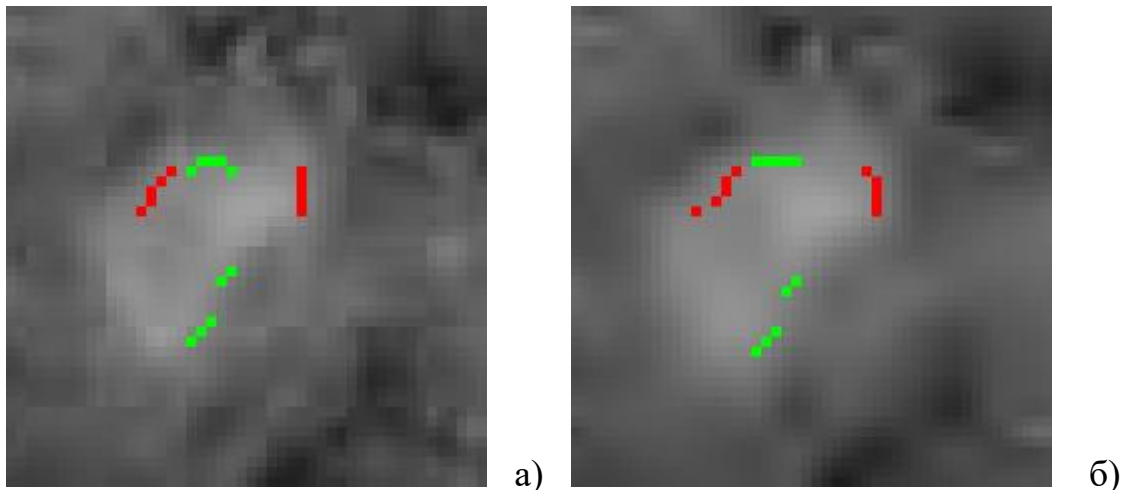


Рис. 4.8. Зміщення контурних точок при стисненні напівтонового еквівалента відеозображення №2: а) початкове відеозображення, б) відновлене відеозображення після стиснення в 50 разів

Похибкак $\Delta_{coord.}$ знаходилася за допомогою наступної програми:

```
dx_dy = [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1 0]
L = 20
summ = 0;
for i = 1:L
    summ = summ + (dx_dy(i))^2;
end
```

```
deltaCoord = sqrt (summ/L)
```

В масиві dx_dy зберігаються зміщення координат контурних точок.

В циклі

```
for i = 1:L  
    summ = summ + (dx_dy(i))^2;  
end
```

знаходиться значення виразу $\sum_{i=1}^L (R_{compr.}(i) - R_{orig.}(i))^2$, що присутній в формулі

для розрахунку похибки визначення координат.

$\text{deltaCoord} = \sqrt{\text{summ}/L}$ – середньоквадратична похибка визначення координат контурних точок об'єкта.

Результат виконання програми:

```
>>
```

```
L =  
    20
```

```
deltaCoord =  
    0.2236
```

```
>>
```

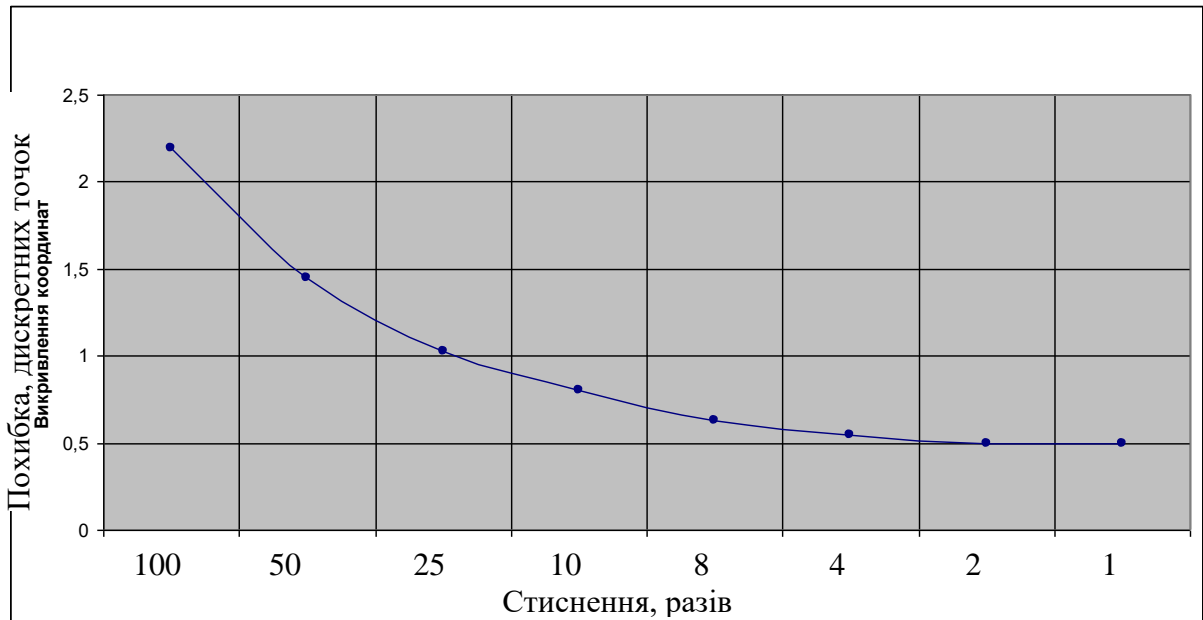
Так, було знайдено середньоквадратичні похибки координат контурних точок об'єктів для усіх напівтонових відеозображень з усіма коефіцієнтами стиснення.

Також було оцінено похибку визначення площі об'єктів на відеозображеннях, відновлених після стиснення:

$$\Delta_S = S_{compr.} - S_{orig.},$$

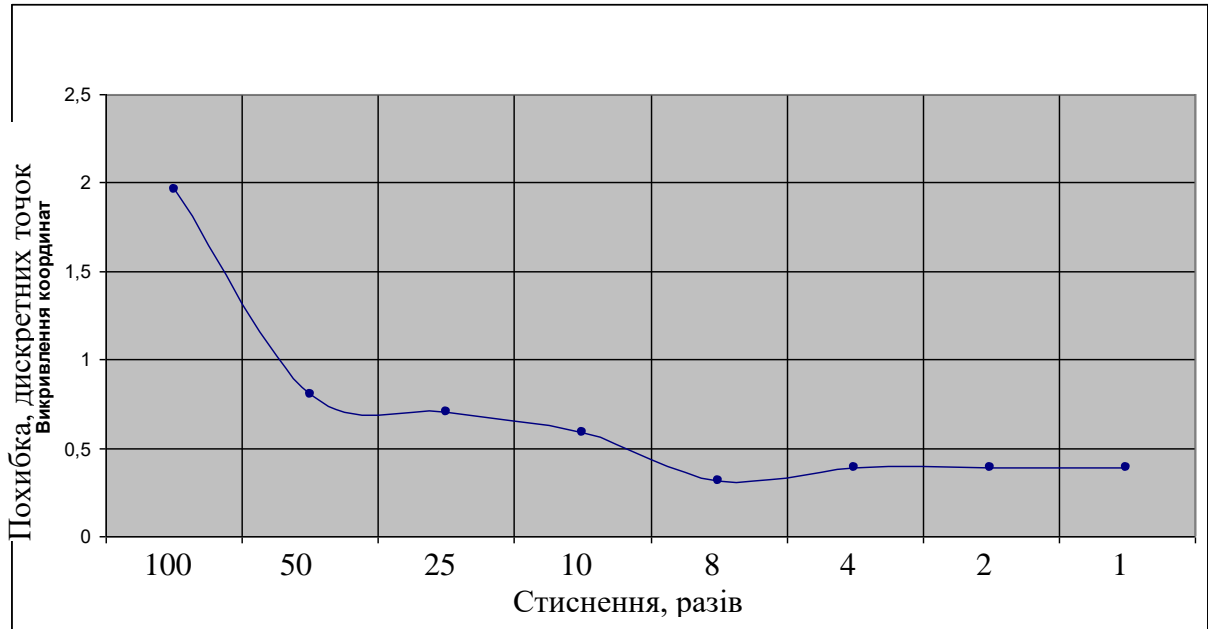
де $S_{compr.}$, $S_{orig.}$ – площі об'єктів на відновленому та початковому відеозображеннях. На практиці $S_{compr.}$, $S_{orig.}$ визначаються як кількість дискретних точок відеозображення, що належать об'єкту та знайдені шляхом сегментації.

На основі отриманих даних побудовано графіки залежності похибок визначення координат контурних точок та площі об'єктів в залежності від величини стиснення цифрового відеозображення (рис. 4.9, 4.10).

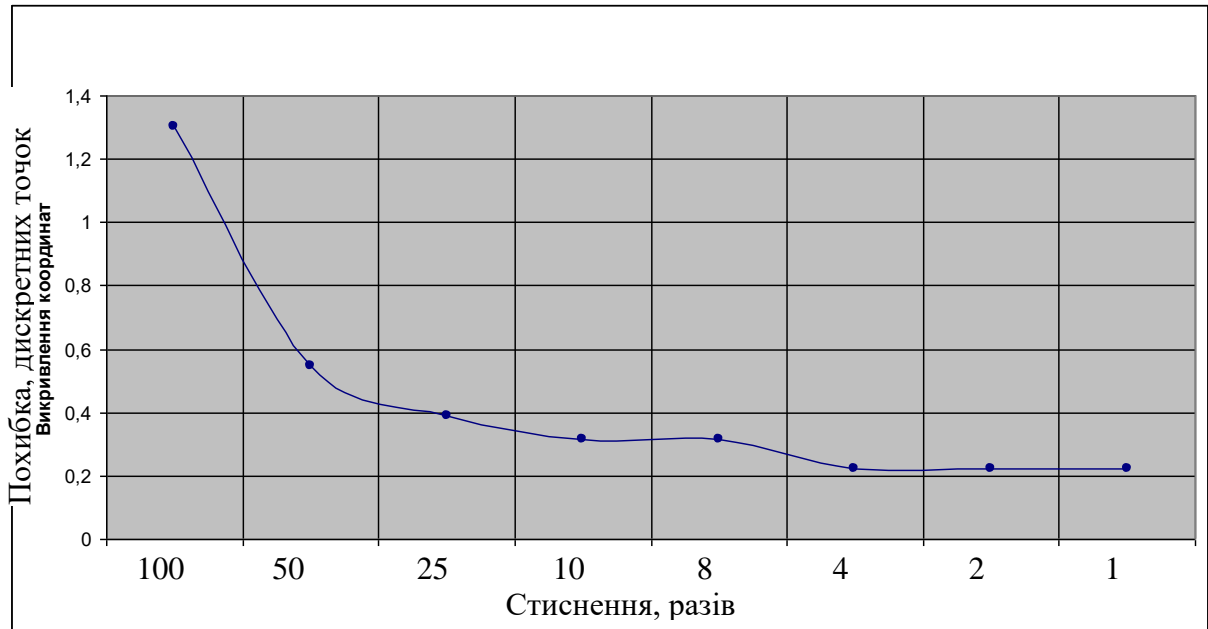


а)

Рис. 4.9. Похибки визначення координат контурних точок об'єктів в залежності від величини стиснення: а) відеозображення №1; б) відеозображення №2; в) відеозображення №3; г) відеозображення №4; д) відеозображення №5

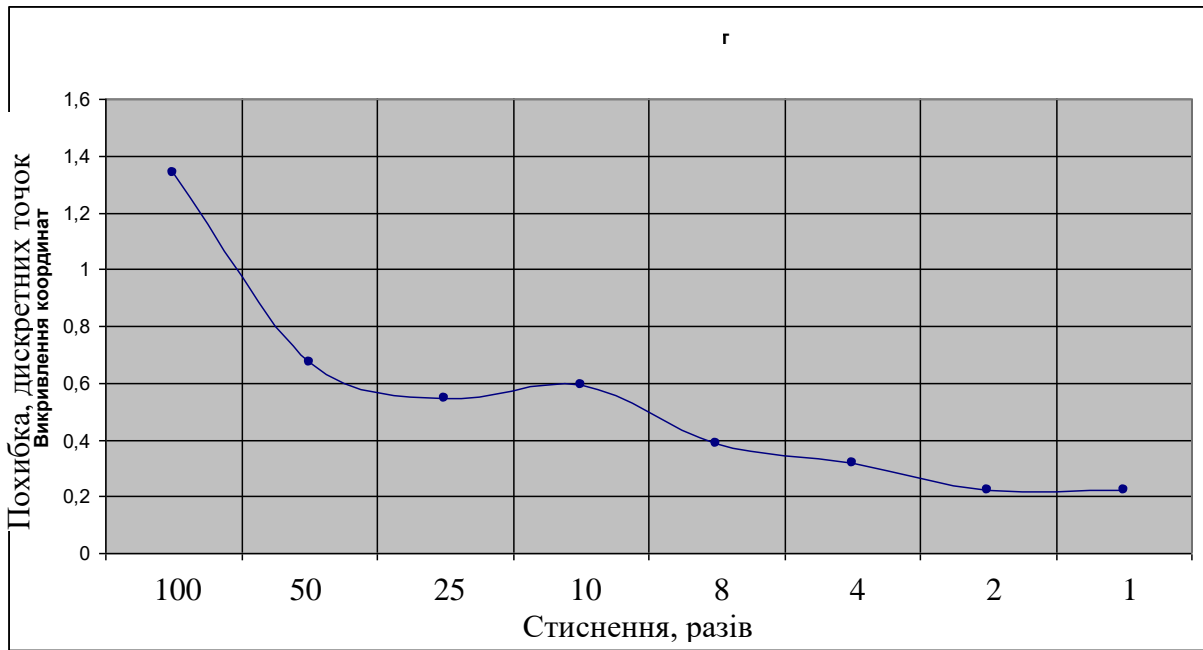


б)

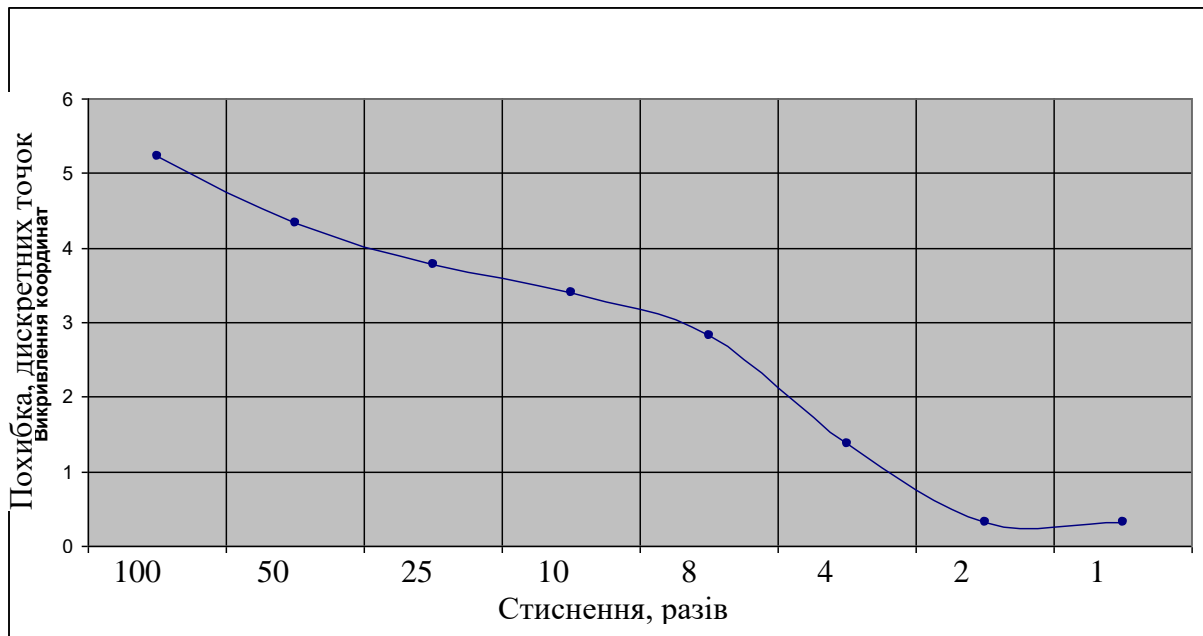


в)

Рис. 4.9 (продовження)

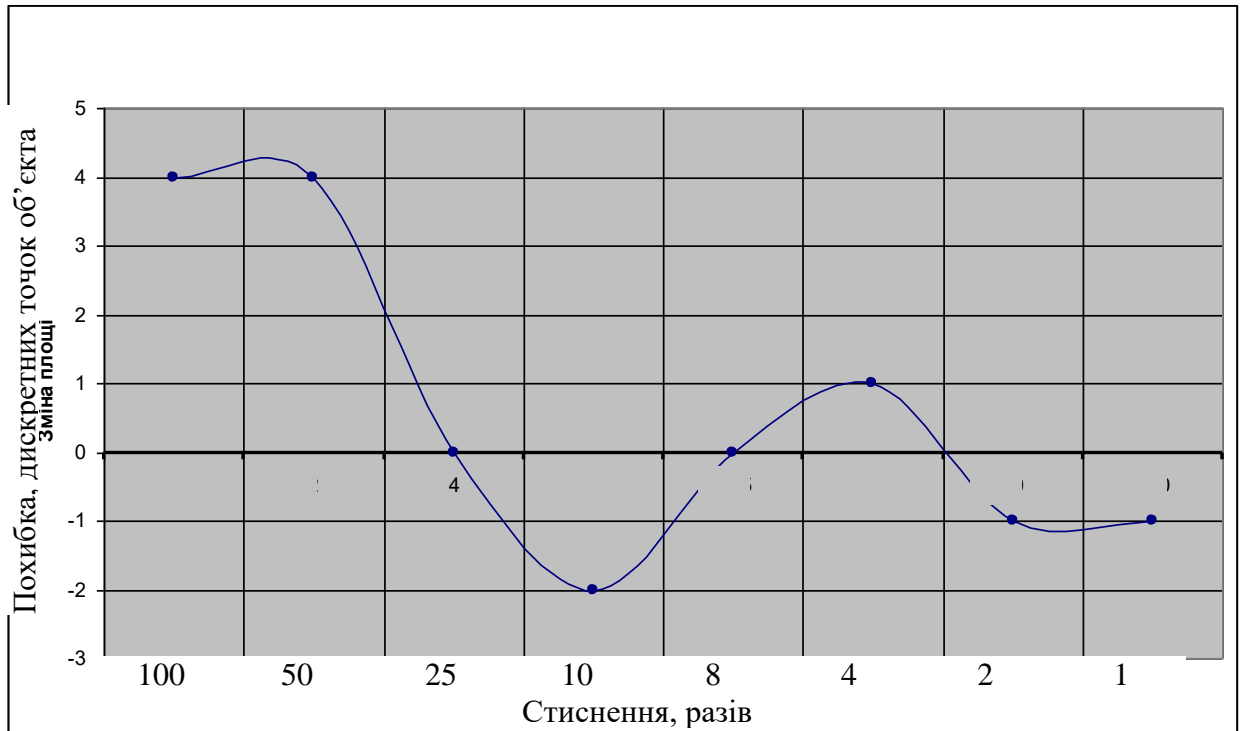


г)



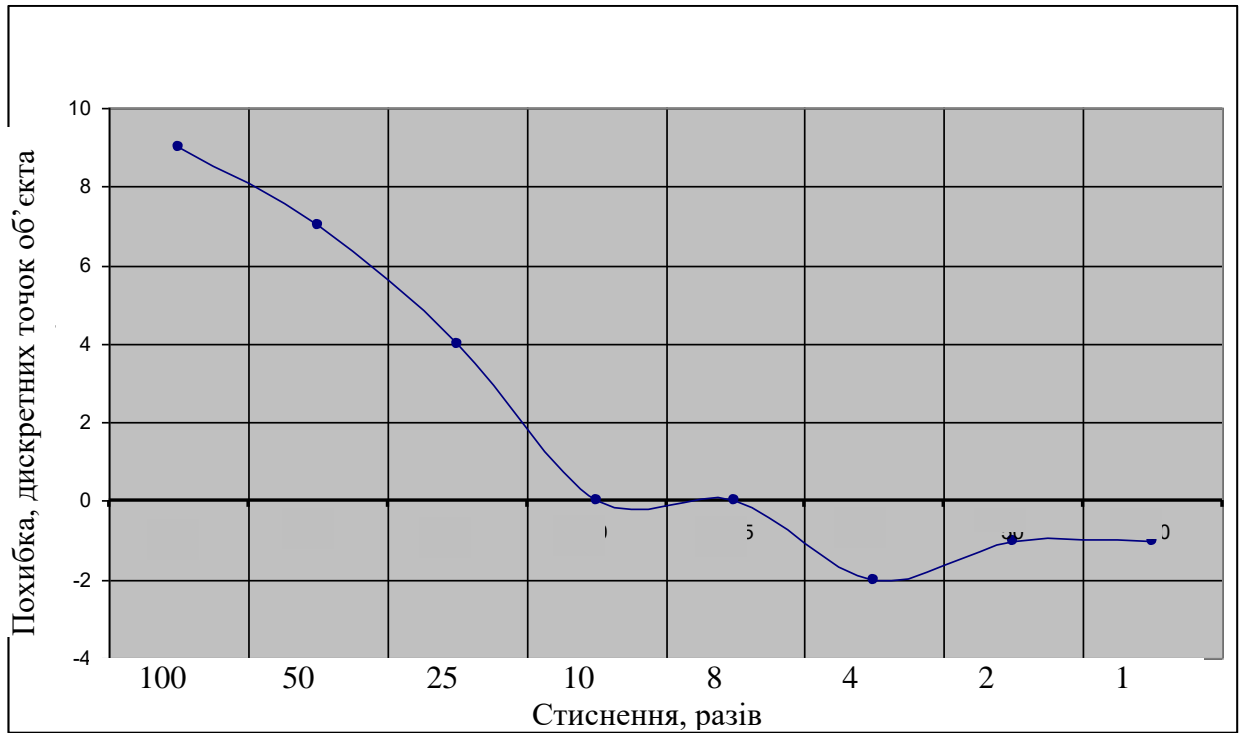
д)

Рис. 4.9 (продовження)

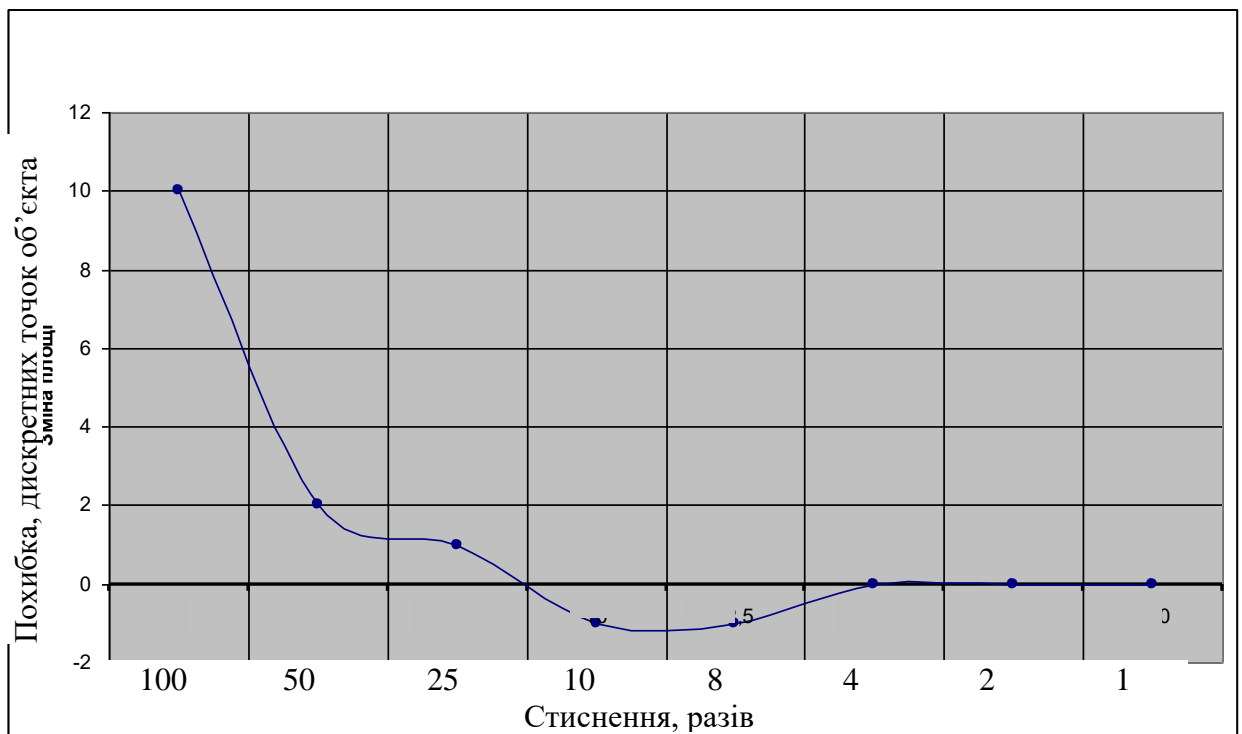


а)

Рис. 4.10. Похибки визначення площі об'єктів в залежності від величини стиснення: а) відеозображення №1; б) відеозображення №2; в) відеозображення №3; г) відеозображення №4; д) відеозображення №5

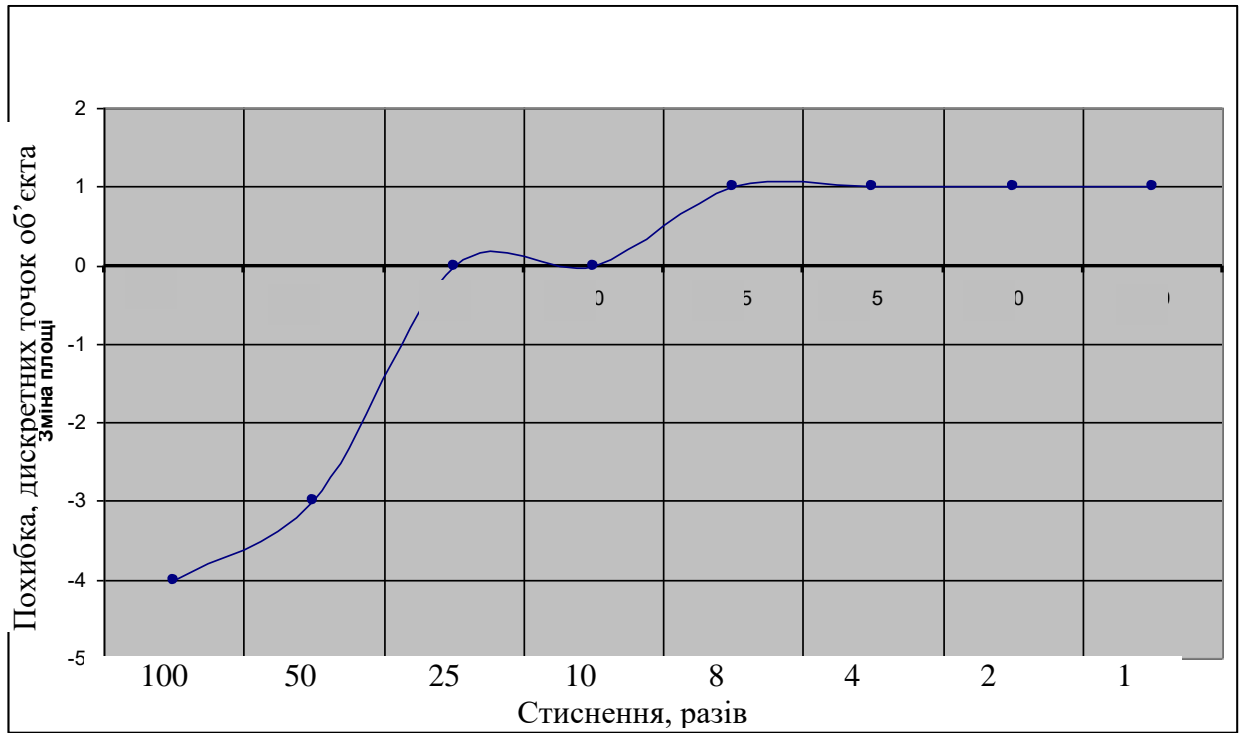


б)

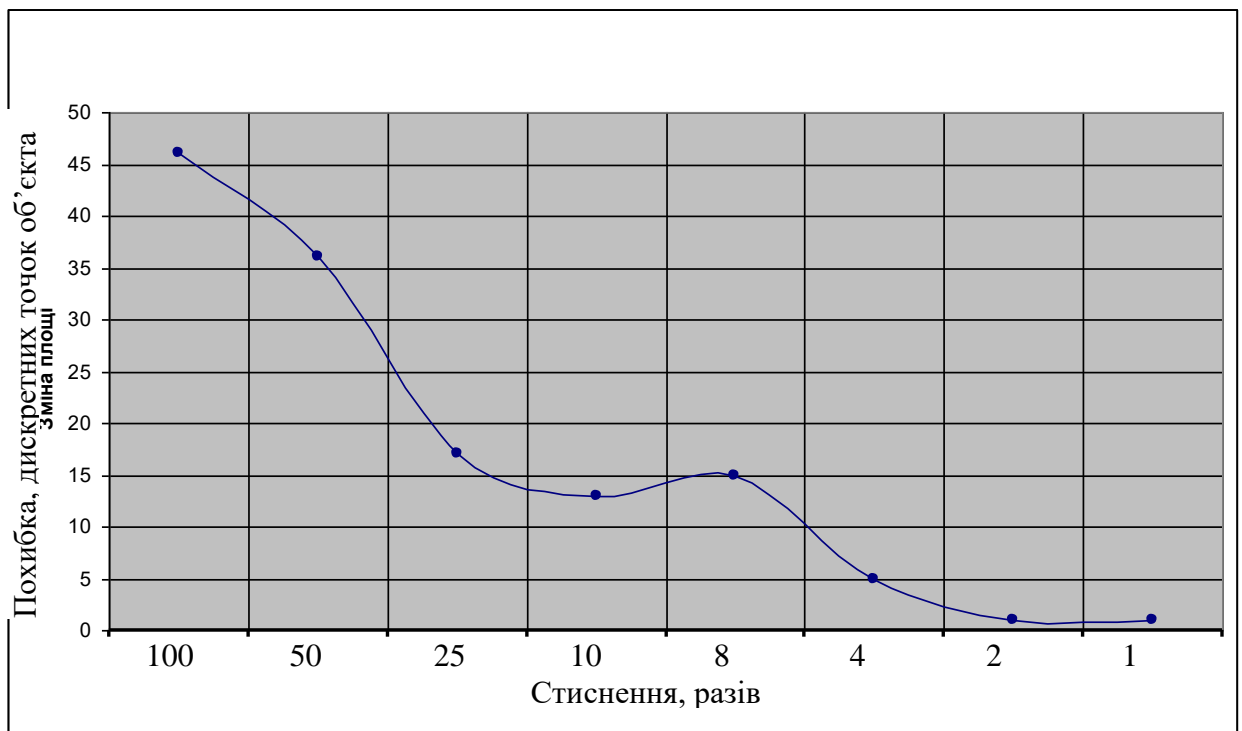


в)

Рис. 4.10 (продовження)



г)



д)

Рис. 4.10 (продовження)