

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Тернопільський національний економічний університет
Факультет комп'ютерних інформаційних технологій
Кафедра спеціалізованих комп'ютерних систем

БОЙЧУН Михайло Васильович

КОМП'ЮТЕРНО-ІНТЕГРОВАНА СИСТЕМА КОМПРЕСІЇ
ВІДЕОДАНИХ НА ОСНОВІ ЕНТРОПІЙНОЇ ФУНКЦІЇ
ЕКВІВАЛЕНТНОСТІ / COMPUTER-INTEGRATED VIDEO
COMPRESSION SYSTEM BASED ON ENTROPY FUNCTION OF
EQUIVALENCE

спеціальність: 151 – Автоматизація та комп'ютерно-інтегровані технології
магістерська програма – Автоматизація та комп'ютерно-інтегровані технології

Магістерська робота

Виконав студент групи АКІТм-21
М. В. Бойчун

Науковий керівник:
к.т.н., доцент А. І. Сегін

Магістерську роботу допущено до захисту:

" ____ " _____ 20__ р.

Завідувач кафедри

_____ Я.М. Николайчук

Тернопіль 2018

Тернопільський національний економічний університет
Факультет комп'ютерних інформаційних технологій
Кафедра спеціалізованих комп'ютерних систем
Освітній ступінь "магістр"

спеціальність: 151 – Автоматизація та комп'ютерно-інтегровані технології
магістерська програма – Автоматизація та комп'ютерно-інтегровані технології

ЗАТВЕРДЖУЮ

Завідувач кафедри СКС

_____ Я.М.Николайчук

“ ____ ” _____ 20__ р.

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ
БОЙЧУН Михайло Васильович

(прізвище, ім'я по-батькові)

1. Тема магістерської роботи

Комп'ютерно-інтегрована система компресії відеоданих на основі ентропійної функції еквівалентності / Computer-integrated video compression system based on entropy function of equivalence

керівник роботи _____ к.т.н., доцент А. І. Сегін

затвержені наказом по університету від "14" листопада 2017 р. № 804

2. Строк подання студентом закінченої магістерської роботи 16 листопада 2018р.

3. Вихідні дані до магістерської роботи:

1. Кодування інформації на основі алгебраїчних кодів

2. Стиснення чорно-білих зображень із втратами

3. Програмен забезпечення реалізації алгоритму ефективного кодування

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):

1. Імовірнісний підхід у теорії кодування відеоінформації.

2. Розробка імовірнісного методу на основі ентропійного аналізу для підвищення ефективності кодування зображень.

3. Аналіз ентропійних оцінок.

4. Розробка ефективного алгоритму кодування зображень на основі зміни градації яскравості.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень):

6. Консультанти розділів магістерської роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання 14 листопада 2017 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1	Імовірнісний підхід у теорії кодування відеоінформації.	11.2017р. – 01.2018р.	
2	Розробка імовірнісного методу на основі ентропійного аналізу для підвищення ефективності кодування зображень	02.2018р. – 04.2018р.	
3	Аналіз ентропійних оцінок.	05.2018р. – 07.2018р.	
4	Розробка ефективного алгоритму кодування зображень на основі зміни градації яскравості	08.2018р. – 1.10.2018р.	

Студент

(підпис)

М. В. Бойчун

Керівник магістерської роботи

(підпис)

к.т.н., доцент А. І. Сегін

РЕФЕРАТ

Робота виконана на 76 сторінках та містить 14 рисунків, 6 таблиць, 34 джерел за переліком посилань.

Мета роботи. Розробка алгоритму та програмного забезпечення ефективного кодування чорно-білих зображень на основі його ентропійних характеристик.

Методи дослідження базуються на методах інформаційного імовірнісного контекстно-залежного моделювання, теорії інформації та кодування і математичної статистики.

Результатами роботи є алгоритм ефективного кодування відеоінформації зі втратами та його програмна реалізація.

Рекомендації по використанню результатів роботи. Результати роботи можна застосувати при передачі, відображенні та зберіганні відеоінформації з допустимими втратами інформації.

Можливі напрямки розвитку. В роботі увага приділяється кодуванню найпростішого виду відеоінформації – чорно-білого зображення, що дає можливість отримати основи алгоритмів для кодування більш складних видів зображень, таких як кольорові картинки та відео файли.

Ключові слова: ЕНТРОПІЯ, ЕФЕКТИВНЕ КОДУВАННЯ, КЛАСИФІКАЦІЙНІ ОЗНАКИ.

ABSTRACT

Work is executed on 76 pages and contains 14 figures, 6 tables, 34 sources for references.

Purpose of the work. Development of algorithms and software efficient coding of black and white images based on characteristics of its entropy.

Research methods. Methods based on probabilistic methods informational context-dependent modeling, information theory and coding and mathematical statistics.

Job performances. The results of the algorithm is efficient video encoding with losses and its implementation.

Recommendations after the use of job performances. The results of the work can be applied for transmitting, displaying and storing video information with allowable loss of information.

Possible development directions. In baccalaureate work focuses on the simplest type of video encoding - black and white image that gives an opportunity to get the basics of algorithms to encode more complex types of images, such as color pictures and video files.

Keywords: ENTROPY, EFFICIENT CODING, CLASSIFICATION FEATURES.

ВСТУП

Актуальність теми. Розвитком комп'ютерних мереж, в тому числі й Інтернету, мобільного зв'язку, супутникового і ефірного телебачення та інших телекомунікаційних систем вимагає вдосконалення способів зберігання й передачі інформації в цифровому вигляді. Як відомо, відеоінформація є одним з найбільш об'ємних типів цифрової інформації. З врахуванням обмеженої пропускної здатності цифрових каналів, щоб гарантувати високу якість передачі зображень, необхідно забезпечити їх досить ефективне представлення.

З метою зниження обчислювальної складності на практиці довгий час застосовувалися спрощені методи інформаційного моделювання й генерації коду. Як моделі бралися найпростіші комбінаторні моделі, а генерація коду здійснювалася з використанням найбільш швидких реалізацій префіксного кодування. У наш час постановка завдання змінилася: на перший план стала виходити ефективність кодування. Сьогодні стає доцільним застосування більше складних технологій кодування, які дозволяють досягти максимально компактного інформаційного повідомлення.

Для одержання компактних інформаційних повідомлень застосовуються технології ефективного кодування. Використання цих технологій дозволяє істотно знизити вимоги до обсягу інформаційних носіїв, а також відчутно збільшує швидкість передачі інформації в каналах зв'язку. В зв'язку з цим, в останні часи значні зусилля направлені на розробку технологій ефективного кодування зображень.

Мета дослідження Метою даної роботи є аналіз існуючих технологій одержання компактних представлень відеоінформації з точки зору способу організації кодування й пошук можливих шляхів підвищення їхньої ефективності. У бакалаврській роботі розробляється алгоритми та програмне забезпечення кодування відеоінформації, яке базується на сполученні контекстно-залежного імовірнісного моделювання й арифметичного

кодування, що є найбільш ефективним з погляду компактності представлення відеоінформації.

Вибір напрямку дослідження заснований на результатах порівняльного аналізу існуючих алгоритмів ефективного кодування. З метою зниження обчислювальної складності на практиці довгий час застосовувалися спрощені методи інформаційного моделювання й генерації коду.

Предметом дослідження є способи цифрового представлення відеозображень з метою їх відображення, обробки, передавання та збереження електронними засобами

Об'єктом дослідження є алгоритми стиснення відеозображень на основі використання алгебраїчних кодів.

Методи дослідження базуються на методах інформаційного імовірнісного контекстно-залежного моделювання, теорії інформації та кодування і математичної статистики.

Одержані результати та їх новизна в тому, що розроблено програмне забезпечення для ефективного кодування чорно-білих зображень на основі їх ентропійних характеристик, що забезпечує стиснення їх цифрового представлення. Для покращення методу використано ентропійну оцінку еквівалентності, що дозволило приблизно в 2 рази більше стиснути відеодані при незначній втраті швидкодії.

Напрямки подальшого розвитку В роботі увага приділяється кодуванню найпростішого виду відеоінформації – чорно-білого зображення, що дає можливість отримати основи алгоритмів для кодування більш складних видів зображень, таких як кольорові картинки та відео файли.

РОЗДІЛ 1

ІМОВІРНІСНИЙ ПІДХІД У ТЕОРІЇ КОДУВАННЯ ВІДЕОІНФОРМАЦІЇ

1.1 Теоретичні основи визначення ентропії як границі довжини коду

Метою ефективного кодування є створення ефективних методів представлень інформаційної вибірки. Мова в більшості випадків іде про вибірку джерел дискретної інформації з кінцевим алфавітом. Ефективне кодування стає можливим завдяки наявності в інформації певних особливостей, тому однією з найбільш важливих задач є одержання як можна більше точного опису властивостей інформаційних джерел.

Існує кілька підходів до такого роду опису. Найчастіше використовуються комбінаторний та ймовірнісний.

У рамках комбінаторного підходу символи розглядаються не обособлено, а групами, іменованими інформаційними повідомленнями. Вважається, що з джерела інформації можуть надходити не всі можливі повідомлення, а тільки повідомлення з деякого виділеної множини. При цьому повідомлення, що належать даній множині, вважаються повністю рівнозначними, тобто їхня поява є рівноймовірною. Конкретний вибір множини допустимих повідомлень виконує роль інформаційного опису.

Комбінаторний підхід одержав досить широке поширення на практиці. Основною його перевагою є простота опису інформаційних особливостей, тому практичні реалізації методів ефективного кодування, в основі яких лежить даний підхід, мають низьку обчислювальну складність.

Недолік полягає в тому, що точність опису часто прямо залежить від потужності множини припустимих повідомлень. Тому для одержання досить точного опису потрібно розглянути дуже велику кількість повідомлень. Комбінаторний підхід також не дозволяє одержувати інформаційні характеристики для окремих символів усередині повідомлення.

Суть імовірнісного підходу полягає у використанні імовірнісних оцінок фактів появи різних символів на виході джерела інформації. У порівнянні з комбінаторним підходом імовірнісний підхід є більше точним способом опису властивостей джерел інформації. У той же час імовірнісний підхід менш вигідний з обчислювальної точки зору, в зв'язку з тим, що його застосування пов'язане з обчисленням складних імовірнісних оцінок. Таким чином, з одного боку, імовірнісний опис, як правило, більш точний в порівнянні з комбінаторним описом, з іншого боку, імовірнісний опис не завжди можна використати на практиці через існуючі обчислювальні обмеження. Постійне збільшення продуктивності обчислювальних систем робить останній фактор менш значимим, внаслідок чого імовірнісний підхід останнім часом все частіше береться за основу при розробці алгоритмів ефективного кодування.

Основоположником імовірнісного підходу є Шеннон. Він запропонував ввести характеристику невизначеності інформації $H(p_1, p_2, \dots, p_n)$, що залежить від імовірностей $p_1; p_2; \dots; p_n$ появи символів алфавіту A потужності N на виході інформаційного джерела. Шеннон запропонував, щоб міра невизначеності, за аналогією з фізичною характеристикою називалася ентропією й мала наступні властивості:

величина $H(p_1, p_2, \dots, p_n)$ інваріантна щодо перестановок аргументів p_1, p_2, \dots, p_n ;

функція $H(p_1, p_2, \dots, p_n)$ безперервна по кожному з своїх аргументів $[p_1, p_2, \dots, p_n]$;

функція $A(N) = H(1/N, \dots, 1/N)$ з кількістю елементів $1/N$ рівна N , монотонно зростає по N

виконується співвідношення: $H(p_1; p_2; \dots; p_k; p_{k+1}; \dots; p_{k+1}) = H(p_1; p_2; \dots; p_k; p_s) + p_s H(p_{k+1}/p_s, \dots, p_{k+1}/p_s)$, де p_s – сума від 1 до (p_{k+1}) .

Як показав Шеннон, функція, що задовільняє зазначеним властивостям, має вигляд $H(p_1; p_2; \dots; p_n) = -K \sum p_i \log_m p_i$, де K та m – деякі позитивні

константи. Для зручності пропонується вибирати K рівним одиниці, а в якості m брати основу системи представлення інформації. Як показує практика, такий вибір багато в чому виправданий. Наприклад, у системі представлення інформації з основою 2 невизначеність появи одного із двох символів, що з'являються з рівною ймовірністю, виявляється рівною 1 біту. Це добре узгоджується з визначенням біта як одиниці інформації, необхідної для вибору однієї із двох рівноймовірних подій. Таким чином, остаточна формула для обчислення ентропії інформаційного джерела з імовірнісним розподілом появи символів $\{p_i\}_{i=1}^N$ виглядає в такий спосіб:

$$H(p_1, p_2, \dots, p_N) = - \sum_{i=1}^N p_i \log_m p_i. \quad (1.1)$$

Формула (1.1) дозволяє визначити ентропію джерела, що перебуває в деякому конкретному стані. Ентропія джерела в цілому – безвідносно до конкретного стану – може бути визначена далеко не завжди. Можливість визначення ентропії залежить від імовірностей тих або інших станів і відповідних цих станів ансамблів перехідних імовірностей (перехід зі стану в стан здійснюється при породженні джерелом чергового символу). Розглянемо окремий випадок, коли інформаційне джерело S може перебувати в одному з T станів, причому чітко відомі всі ймовірності переходів джерела з будь-якого стану i у стан j (p_{ij}). Якщо існує межа $P^\infty = \prod_{i=1}^{\infty} P$, де P – матриця з компонентами $p_{i,j}$, величина ентропії джерела S може бути обчислена по формулі:

$$H(S) = - \sum_{i=1}^T p_i \sum_{j=1}^T p_{i,j} \log_m p_{i,j}. \quad (1.2)$$

Тут p_i – імовірність знаходження джерела S в i -му стані ($p_i \in$ значення i -елемента будь-якого рядка матриці P^∞), а m – основа системи представлення інформації.

Множина станів інформаційного джерела в сукупності з ансамблем перехідних імовірностей прийнято називати моделлю станів або марковською моделлю. Для визначення ентропії джерела інформації в рамках даної моделі необхідно правильно підібрати її структуру й параметри, які повинні повною мірою відповідати джерелу. На практиці це досить важко, тому що параметри інформаційних джерел, як правило, апріорно не відомі. Визначити їх можна тільки приблизно, тому при рішенні практичних задач доводиться застосовувати емпіричні способи оцінки ентропії. Як правило, мова йде про усереднену по часу величину ентропії.

Нехай джерело послідовно перебуває в станах s_1, s_2, \dots, s_n , які відповідають розподілу ймовірностей появи символів

$$\{p_i^{(s_1)}\}_{i=1}^N, \{p_i^{(s_2)}\}_{i=1}^N, \dots, \{p_i^{(s_n)}\}_{i=1}^N.$$

Через $p_{i_1}, p_{i_2}, p_{i_n}$ позначимо ймовірність появи на виході джерела послідовності символів з індексами i_1, i_2, \dots, i_n (символ i_k породжується джерелом у стані s_k). Очевидно, що

$$p_{i_1, i_2, \dots, i_n} = p_{i_1}^{(s_1)} p_{i_2}^{(s_2)} \dots p_{i_n}^{(s_n)}. \quad (1.3)$$

Напівемпіричною ентропією джерела назвемо величину

$$-\frac{\sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1, i_2, \dots, i_n} \log_m p_{i_1, i_2, \dots, i_n}}{n}. \quad (1.4)$$

З урахуванням тотожності (1.3) формула (1.4) зводиться до більш зручного для обчислення виду:

$$\begin{aligned}
 & - \frac{\sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1, i_2, \dots, i_n} \log_m p_{i_1, i_2, \dots, i_n}}{n} = \\
 & = \frac{- \sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1}^{(s_1)} p_{i_2}^{(s_2)} \dots p_{i_n}^{(s_n)} \log_m (p_{i_1}^{(s_1)} p_{i_2}^{(s_2)} \dots p_{i_n}^{(s_n)})}{n} = \\
 & = \frac{- \sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1}^{(s_1)} p_{i_2}^{(s_2)} \dots p_{i_n}^{(s_n)} (\log_m p_{i_1}^{(s_1)} + \log_m p_{i_2}^{(s_2)} + \dots + \log_m p_{i_n}^{(s_n)})}{n} = \\
 & = \frac{- \sum_{i=1}^N p_i^{(s_1)} \log_m p_i^{(s_1)} - \sum_{i=1}^N p_i^{(s_2)} \log_m p_i^{(s_2)} - \dots - \sum_{i=1}^N p_i^{(s_n)} \log_m p_i^{(s_n)}}{n} = \\
 & = \frac{- \sum_{j=1}^n \sum_{i=1}^N p_i^{(s_j)} \log_m p_i^{(s_j)}}{n}.
 \end{aligned}$$

Префікс "напів" означає, що для отримання ентропії частково використовуються апріорні дані про імовірнісні характеристики джерела (відомий розподіл імовірностей появи символів у різних станах), а частина, що залишилася інформації – конкретна послідовність станів – виходить емпіричним шляхом. У реальних задачах, як правило, подібної апріорної інформації нема, тому доцільно ввести поняття *емпіричної ентропії*, визначивши її величину по формулі:

$$\frac{- \sum_{j=1}^n \sum_{i=1}^N r_i^{(s_j)} \log_m r_i^{(s_j)}}{n}, \tag{1.6}$$

У цьому випадку $r_i^{(sj)}$ – емпірична оцінка ймовірності появи символу з індексом i при породженні j -ї вибірки джерела інформації. Причина, по якій особлива увага приділяється способам обчислення ентропії, полягає в тому, що величина ентропії є чисельною границею ефективності кодування вибірки інформаційного джерела. Знаючи ентропію, можна оцінити ефективність того або іншого методу або алгоритму. В ідеалі ефективність повинна збігатися з величиною ентропії. При цьому, якщо природа інформаційного джерела не є імовірнісною, тобто чітко виділити ті або інші імовірнісні стани не можна, мова може йти тільки про ентропію, що обчислюється частково на емпіричній основі. В силу останньої обставини, надалі при вживанні терміна ентропія найчастіше буде матися на увазі напівемпірична ентропія.

Обчислення ентропії на практиці можна здійснювати в процесі послідовного аналізу станів інформаційного джерела, що безпосередньо необхідно для формул (1.5) і (1.6). Як видно з наведених формул, величина ентропії складається з величин ентропії в станах, у яких послідовно перебуває джерело інформації. Таким чином, задача оптимального моделювання послідовної інформаційної вибірки зводиться до задачі створення оптимальної моделі породження символів у кожному конкретному стані, що є істотним спрощенням. Як буде показано нижче, подібне спрощення припустимо й відносно механізму генерації коду, тобто воно фактично припустимо відносно методу кодування в цілому.

1.2 Обґрунтування оптимальної довжина коду, необхідної для декодування даних

Найпростіший варіант кодування вибірки інформаційного джерела – установлення відповідності між конкретними символами алфавіту й кодами, що мають цілу довжину в одиницях подання інформації. Природною є вимога, щоб код, отриманий у результаті такого кодування, був декодованим,

тобто по будь-якій комбінації кодів символів можна було б відновити закодоване повідомлення. Необхідна умова декодування була запропоновано й доведена Макмілланом. Формулювання виглядає таким чином: якщо система кодів із цілими довжинами $\{l_i\}_{i=1}^N$ декодовані, то виконується нерівність

$$\sum_{i=1}^N m^{-l_i} \leq 1, \quad (1.7)$$

де m – основа системи представлення інформації.

Виконання нерівності (1.7) обумовлює виконання більше важливої нерівності:

$$\sum_{i=1}^N p_i l_i \geq - \sum_{i=1}^N p_i \log_m p_i. \quad (1.8)$$

Для доказу досить скористатися опуклістю функції $\log_m(x)$. Таким чином, ефективність декодованого посимвольного кодування обмежена величиною ентропії імовірнісного розподілу появи символів на виході інформаційного джерела.

Більше складний і в той же час найбільше ефективний варіант кодування передбачає одержання кодів не для окремих символів, а для цілих повідомлень. Коди декодованих повідомлень, очевидно, також повинні задовольняти нерівності Макміллана. При цьому оцінка ефективності з розрахунку на один символ повідомлення ускладнюється необхідністю проведення усереднення по довжині повідомлення. Доведено, що ефективність кодування вибірки інформаційного джерела зазначеним способом також обмежується величиною ентропії.

Використовуючи раніше введені позначення, джерело, що послідовно перебуває в станах s_1, s_2, \dots, s_n , характеризується імовірнісними розподілами

$$\{p_i^{(s_1)}\}_{i=1}^N, \{p_i^{(s_2)}\}_{i=1}^N, \dots, \{p_i^{(s_n)}\}_{i=1}^N.$$

Через $p_{i_1}, p_{i_2}, \dots, p_{i_n}$ позначається ймовірність появи повідомлення " i_1, i_2, \dots, i_n ", через $l_{i_1}, l_{i_2}, \dots, l_{i_n}$ – довжина коду повідомлення. Ефективність кодування повідомлень довжини n з розрахунку на один символ визначається по формулі

$$\frac{\sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1, i_2, \dots, i_n} l_{i_1, i_2, \dots, i_n}}{n}.$$

Коди для повідомлень довжини n повинні бути декодованими, тому можна скористатися нерівністю (1.8):

$$\frac{\sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1, i_2, \dots, i_n} l_{i_1, i_2, \dots, i_n}}{n} \geq \frac{\sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1, i_2, \dots, i_n} \log_m p_{i_1, i_2, \dots, i_n}}{n}.$$

Права частина нерівності являє собою вираз для визначення ентропії. Таким чином, ефективність кодування з розрахунку на один символ не може перевершувати величину ентропії повідомлення, що приходить на один символ. Використовуючи формулу (1.5), одержимо більше зручну з обчислювальної точки зору оцінку:

$$\frac{\sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1, i_2, \dots, i_n} l_{i_1, i_2, \dots, i_n}}{n} \geq \frac{\sum_{j=1}^n \sum_{i=1}^N p_i^{(j)} \log_m p_i^{(j)}}{n}.$$

Якщо джерело, що породжує повідомлення, є стаціонарним (ймовірнісний розподіл $\{p_i\}_i^N = 1$ не залежить від позиції символу в повідомленні), нерівність приймає спрощений вигляд:

$$\frac{\sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1, i_2, \dots, i_n} l_{i_1, i_2, \dots, i_n}}{n} \geq - \sum_{i=1}^N p_i \log_m p_i.$$

Оскільки код виходить для повідомлення в цілому, не можна говорити про коди для конкретних символів, що становлять повідомлення. Проте виникає необхідність визначати деякий вклад у результуючу довжину коду повідомлення, складеним з кожного символом, що входить в нього. Фізичний зміст такого вкладу – середнє збільшення довжини коду повідомлення, обумовлене входженням у нього даного символу. Позначимо через x_i вклад, внесений символом з індексом i (у загальному випадку величина вкладу являє собою дійсне число). Через $x_{i_1}, x_{i_2}, \dots, x_{i_n}$ позначимо вклад, внесений повідомленням $s = i_1, i_2, \dots, i_n$. Виходячи зі змісту поняття "вклад", для випадку стаціонарної незалежної вибірки логічно припустити виконання наступних властивостей:

$$x_{i_1, i_2, \dots, i_n} = x_{i_1} + x_{i_2} + \dots + x_{i_n}$$

(адитивність)

$$|l_{i_1, i_2, \dots, i_n} - x_{i_1, i_2, \dots, i_n}| \leq M(n),$$

де $l_{i_1}, l_{i_2}, \dots, l_{i_n}$ – довжина реального коду повідомлення (ціле число), а $M(n)$ – ненегативна функція, така що $M(n)/n \rightarrow 0$ при $n \rightarrow \infty$

Асимптотична поведінка функції $M(n)/n$ дозволяє зробити висновок:

$$\sum_{i=1}^N m^{-x_i} \leq 1. \quad (1.9)$$

Таким чином, отримане узагальнення нерівності Макміллана на випадок вкладів символів у результуючу довжину повідомлень.

Узагальнена нерівність (1.9) дозволяє обчислити точне значення величини вкладу символу x_a з індексом a для випадку оптимального кодування. Для обчислення точного значення необхідно вирішити задачу про мінімізацію суми:

$$\sum_{i=1}^N p_i x_i,$$

яка визначає середню ефективність кодування, за умови виконання нерівності (1.9). У точці мінімуму похідна від зазначеної суми перетворюється в нуль:

$$\left(\sum_{i=1}^N p_i x_i \right)'_{x_a} = 0. \quad (1.10)$$

Підстановка рішення задачі про мінімізацію в нерівність (1.9), повинно перетворювати його в рівність. Звідси будь-який оптимальний внесок x_k легко виражається через внески інших символів:

$$x_k = -\log_m \left(1 - \sum_{i=1}^{k-1} m^{-x_i} - \sum_{i=k+1}^N m^{-x_i} \right).$$

Підставляючи вираз для x_k у рівняння (1.10), одержуємо

$$\left(\sum_{i=1}^{k-1} p_i x_i - p_k \log_m \left(1 - \sum_{i=1}^{k-1} m^{-x_i} - \sum_{i=k+1}^N m^{-x_i} \right) + \sum_{i=k+1}^N p_i x_i \right)'_{x_a} = 0$$

Після взяття похідної рівняння отримує наступний вигляд:

$$p_a - p_k \frac{m^{-x_a}}{m^{-x_k}} = 0, \text{ або } p_a m^{-x_k} = p_k m^{-x_a}.$$

Індекс k може приймати одне з N значень, тобто реально для кожного фіксованого індексу $a \in N$ рівнянь. Просумувавши ці рівняння, одержимо

$$p_a \sum_{k=1}^N m^{-x_k} = \left(\sum_{k=1}^N p_k \right) m^{-x_a}$$

що з урахуванням виконання рівності у виразі (1.9) рівносильне

$$p_a = m^{-x_a}.$$

Звідси отримаємо формулу для x_a :

$$x_a = -\log_m p_a. \quad (1.11)$$

Таким чином, оптимальний вклад символу, що з'являється з імовірністю p у довжину результуючого коду становить $\log_m p$ одиниць інформації в системі представлення з основою m . Такий висновок може бути використаний для обчислення оптимальної довжини коду в рамках тієї або іншої імовірнісної моделі. Одержуючи оцінку ймовірності появи чергового символу на деякому етапі кодування, можна точно визначити оптимальну довжину відповідного інформаційного опису.

1.3 Методи генерації коду

Серед кодів, що задовольняють нерівності Макміллана, особливе місце займають префіксні коди. Система кодів називається префіксною, якщо жоден з кодів меншої довжини, що належить системі, не є початком (префіксом) іншого коду більшої довжини із цієї ж системи. Очевидна перевага префіксного кодування полягає в тому, що одержаний код може бути легко декодований без введення додаткових розділюючих символів. Завдяки властивості префікса для того, щоб визначити черговий закодований символ (повідомлення), досить проаналізувати початок відповідної чергової порції коду. При цьому довжина аналізованої порції ніколи не перевищує довжину коду чергового закодованого символу (повідомлення).

Геометричне трактування систем префіксних кодів є m -арні дерева. Властивість префікса гарантує відсутність циклів у графі, ребрам якого зіставлені різні значення інформаційної одиниці. Таким чином, граф є деревом зі ступенем розгалуження, що збігає з основою системи представлення інформації m . Слід зазначити, що нумерація ребер може бути здійснена довільним чином; значення має тільки конкретна структура дерева, а точніше – набір відстаней від кореневого вузла до листових вузлів. Ці відстані відповідають довжинам кодів префіксної системи. Крафт показав, що виконання нерівності (1.7) є гарантією існування кодового дерева зі структурою, що відповідає набору довжин $\{p_i\}_i^N = 1$, що фігурують у нерівності. Інакше кажучи, якщо система довжин задовольняє нерівності (1.7), можна побудувати систему префіксних кодів з відповідними довжинами. Дане твердження дозволяє відмовитися від розгляду систем кодів, відмінних від префіксних. Будь-яка система дешифрованих кодів задовольняє нерівності (1.7), а отже, вона може бути без втрати ефективності замінена системою префіксних кодів. Нерівність (1.7) стосовно систем префіксних кодів називають також нерівністю Крафта.

Блокове кодування повідомлень довжини n , породжених деяким інформаційним джерелом є удосконаленням префіксного. Як і раніше, позначимо через

$$\{ p_i^{(s_j)} \}_i^N = 1$$

імовірнісний розподіл появи j -го символу повідомлення (s_j – відповідний стан джерела), через $p_{i_1}, p_{i_2}, \dots, p_{i_n}$ – імовірність появи повідомлення i_1, i_2, \dots, i_n . Відповідно до твердження Крафта, можна побудувати систему префіксних кодів із довжинами

$$l_{i_1, i_2, \dots, i_n} = \lceil -\log_m p_{i_1, i_2, \dots, i_n} \rceil.$$

Для доказу досить підставити ці довжини в нерівність Крафта й переконатися в тому, що воно виконується. Оцінимо ефективність кодування з розрахунку на один символ повідомлення:

$$\begin{aligned} \frac{\sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1, i_2, \dots, i_n} l_{i_1, i_2, \dots, i_n}}{n} &= \frac{\sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1, i_2, \dots, i_n} \lceil -\log_m p_{i_1, i_2, \dots, i_n} \rceil}{n} \leq \\ &\leq \frac{\sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1, i_2, \dots, i_n} (-\log_m p_{i_1, i_2, \dots, i_n} + 1)}{n} = \\ &= \frac{-\sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1, i_2, \dots, i_n} \log_m p_{i_1, i_2, \dots, i_n}}{n} + \frac{1}{n}. \end{aligned}$$

Використовуючи альтернативний вираз для ентропії (1.5), одержимо:

$$\frac{\sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1, i_2, \dots, i_n} l_{i_1, i_2, \dots, i_n}}{n} \leq \frac{-\sum_{j=1}^n \sum_{i=1}^N p_i^{(s_j)} \log_m p_i^{(s_j)}}{n} + \frac{1}{n}.$$

Для випадку стаціонарного джерела з розподілом імовірностей

$\{p_i\}_{i=1}^N = 1$ маємо:

$$\frac{\sum_{1 \leq i_1, i_2, \dots, i_n \leq N} p_{i_1, i_2, \dots, i_n} l_{i_1, i_2, \dots, i_n}}{n} \leq - \sum_{i=1}^N p_i \log_m p_i + \frac{1}{n}.$$

Збільшуючи довжину повідомлення n , можна досягнути ефективності кодування як завгодно близької до ентропії джерела інформації. Таким чином, знаючи апріорні ймовірності появи різних символів на виході джерела в кожен конкретний момент часу, можна організувати кодування даного джерела, наближене до оптимального кодування на кожен наперед задану величину, за умови, що є достатній об'єм інформаційної вибірки.

Алгоритм побудови системи префіксних кодів з довжинами, що залежать від імовірностей по формулі $[-\log_m p]$, був запропонований Шенноном. Алгоритм працює наступним чином. Імовірності появи повідомлень p_1, p_2, \dots, p_N розташовуються в порядку зменшення імовірностей (тут N – потужність множини повідомлень).

$$p_1 \geq p_2 \geq \dots \geq p_N.$$

Як код повідомлення з індексом i беруться перші $[-\log_m p_i]$ m -арних розрядів числа $\sum_{j=1}^{i-1} p_j$ так названої накопиченої ймовірності. Оскільки довжини кодів у такій системі не спадають зі зменшенням імовірності й імовірності появи повідомлень із індексами $i+1, i+2, \dots, N$ відрізняються від імовірності появи повідомлення з індексом i принаймні на $m^{[-\log_m p_i]}$, код повідомлення з індексом i не є початком кодів повідомлень із індексами $i+1, i+2, \dots, N$. Таким чином, система кодів є префіксною. Розглянемо геометричне

трактування алгоритму Шеннона. Інтервал $[0,1)$ може бути розбитий на N підінтервалів

$$[a_i, b_i) = \left[\sum_{j=1}^{i-1} p_j, \sum_{j=1}^i p_j \right),$$

відповідним повідомленням з індексами $i = 1, 2, \dots, N$. Для ідентифікації i -го повідомлення необхідно вибрати деяке число з підінтервалу $[a_i, b_i)$, представлене як можна меншою кількістю m -арних розрядів. Для цього потрібно побудувати на інтервалі $[0,1)$ одновірну сітку з постійним періодом, що містить m^{l_i} точок (місце розташування будь-якої точки визначається l_i -розрядним числом), так само одна з яких (ідентифікуюча) належить інтервалу $[a_i, b_i)$. Ясно, що шукана сітка повинна мати період, що не перевищує p_i , тобто $\frac{1}{m^{l_i}} < p_i$. Звідки отримуємо $l_i = \lceil -\log_m p_i \rceil$.

Алгоритм Шеннона дозволяє генерувати коди, довжина яких відрізняється від оптимальних значень, обумовлених по формулі (1.11), менше ніж на одну інформаційну одиницю. Таким чином, різниця між ефективністю кодування й ентропією (так названа надмірність) не перевищує одиниці.

Алгоритм Шеннона має досить високу ефективність, однак він не є оптимальним алгоритмом побудови системи префіксних кодів. Для знаходження оптимального алгоритму необхідно при фіксованому наборі ймовірностей $\{p_1, p_2, \dots, p_N\}$ вирішити задачу про мінімізацію суми $\sum_{i=1}^N p_i l_i$ за

умови виконання нерівності Крафта $\sum_{i=1}^N m^{-l_i} \leq 1$ (тут l_i – довжина коду

повідомлення з індексом i). Ще один алгоритм побудови кодів запропонований Хаффманом.

На практиці алгоритм Хаффмана реалізується наступним чином. На початковому етапі кожному повідомленню ставиться у відповідність вага, яка дорівнює оцінці ймовірності появи даного повідомлення. Повідомлення містяться в список, що впорядковується по спаданню ваг. Надалі елементи списку обробляються з використанням ітеративної процедури. На кожному кроці (ітерації) m останніх елементів списку об'єднуються в новий елемент, що потім розміщується в списку на відповідному місці замість поєднаних елементів. Новому елементу списку ставиться у відповідність вага, яка дорівнює сумі ваг елементів, що заміщають. Кожна ітерація закінчується впорядкуванням отриманого нового списку, що завжди містить як мінімум на один елемент менше, ніж попередній список. Паралельно з роботою зазначеної процедури здійснюється послідовна побудова m -арного дерева. На кожному кроці алгоритму будь-якому елементу списку відповідає кореневий вузол m -арного дерева, що складає з вершин, які відповідають елементам, утворених об'єднанням відповідних елементів. При об'єднанні m елементів списку відбувається об'єднання відповідних дерев в одне нове m -арне дерево, у якому кореневий вузол відповідає новому елементу, що поміщається в список, заміщаючи елементи, і якому відповідають дочірні вузли цього кореневого вузла. Алгоритм завершує роботу, коли в списку залишається один елемент, що відповідає кореневому вузлу побудованого дерева. Для гарантії коректного завершення роботи алгоритму вихідний розмір списку повинен мати довжину, що представляється у вигляді $n \cdot (m - 1) + 1$. Якщо кількість повідомлень не відповідає цій довжині, список доповнюється до необхідного розміру за рахунок додавання в кінець фіктивних елементів, що мають нульові ваги. Побудоване в результаті описаної процедури дерево називається деревом Хаффмана. Система префіксних кодів може бути отримана шляхом присвоєння конкретних m -арних значень ребрам цього дерева.

Алгоритм Хаффмана має найбільшу ефективність серед алгоритмів побудови префіксних кодів по заданому імовірнісному розподілі. Очевидний

недолік алгоритму – більша обчислювальна складність. Алгоритм доцільно використати в тих випадках, коли імовірнісний розподіл залишається незмінним для досить великого об'єму інформаційної вибірки. Застосування алгоритму Хаффмана у випадку, коли статистичні характеристики інформаційного джерела швидко змінюються, сильно утруднено необхідністю здійснення частих змін структури кодового дерева. Галлагером був запропонований ефективний спосіб зміни структури дерева Хаффмана, який не потребує його повної перебудови. Застосування даного способу хоча й приводить до істотного спрощення генерації коду, проте не завжди дозволяє досягти необхідної швидкості обробки інформації. Тим же недоліком володіють і практичні реалізації алгоритму Шеннона.

РОЗДІЛ 2.

РОЗРОБКА ІМОВІРНІСНІСНОГО МЕТОДУ НА ОСНОВІ ЕНТРОПІЙНОГО АНАЛІЗУ ДЛЯ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ КОДУВАННЯ ЗОБРАЖЕНЬ

2.1 Квантування відеозображень

Ефективність кодування природного напівтонового зображення (наприклад, фотографічного) методами, описаними в попередньому розділі, в середньому становить декілька бітів на одне закодоване значення колірної матриці. Стосовно до синтезованих зображень (такі зображення можуть містити текст, малюнки, графічні об'єкти й т.п.) ефективність є трохи вищою й іноді становить менше одного біта на закодоване значення. Однак існують технології кодування, що дозволяють домогтися істотно кращого результату. Вища ефективність досягається за рахунок того, що при кодуванні припускаються деякі інформаційні спотворення. Щоб спотворення були як можна менш помітні, їхні характеристики вибираються з врахуванням особливостей психовізуального сприйняття зображення людиною.

Спотворення найчастіше виникають внаслідок квантування значень інформаційної вибірки. Ідея квантування полягає в заміні всіх значень із групи (інтервалу) деяким єдиним значенням. Даний прийом дозволяє зменшити кількість різних значень, що зустрічаються в матриці, що дозволяє кодувати її меншою кількістю символів. Інформаційні спотворення, виникають через розбіжність згрупованих значень зі значенням, отриманим у процесі їхнього квантування. Повне відновлення інформації стає, таким чином, неможливим.

Розвитком зазначеного підходу є метод у якому групуються не окремі значення, а цілі вектора, складені зі значень, що перебувають на близьких позиціях у матриці. Квантоване значення в даному випадку відповідає деякій множині векторів значень. Таке рішення представляє значно більше

можливостей для інформаційного опису. Воно назване векторне квантування, а розглянутий раніше спрощений варіант називається скалярним, квантуванням.

Неважко помітити, що векторне квантування частково виконує функції контекстно-залежного моделювання: об'єднання векторів значень, що перебувають на близьких позиціях, дозволяє врахувати можливі імовірнісні взаємозв'язки між ними. Тому векторне квантування, як альтернативний метод врахування інформаційних закономірностей, є предметом для окремого розгляду, і буде залишений за рамками даної роботи. Надалі мова йтиме тільки про ті методи кодування, у яких використовується скалярне квантування.

Зупинимося більш детально на деяких нюансах практичної реалізації скалярного квантування. Як відзначалося раніше, об'єднання значень у групі повинне здійснюватись з врахуванням вимог конкретної задачі. У тих випадках, коли специфіка оброблюваної інформації не дозволяє однозначно вибрати спосіб групування значень, використовується найбільш просте рішення – квантовані значення розбиваються на рівні по довжині інтервали. Даний метод квантування називається рівномірним.

Позначимо через Δ довжину інтервалу при рівномірному квантуванні (параметр квантування). Сукупність інтервалів представимо у вигляді $[x_0 + \Delta - i, x_0 + \Delta - (i + 1))$. Значення x_0 належить дійсному інтервалу $[-\Delta/2, \Delta/2)$ і задає базовий зсув інтервалів при квантуванні. При цьому i являє собою номер інтервалу та приймає довільні цілочисельні значення.

Номер інтервалу виступає в ролі об'єкту, що кодується. У процесі декодування по ньому відновлюється шуканий інтервал значень (деквантування). Оскільки конкретні значення відновленню не підлягають, при деквантуванні вони заміняються деяким фіксованим числом. В якості даного числа використовується математичне очікування значень інтервалу. У випадку, коли значення розподілені рівномірно, математичне очікування відповідає середині інтервалу, що визначається за виразом $x_0 + \Delta \cdot i + \Delta/2$.

Квантована інформаційна вибірка часто представлена числами зі знаком, симетрично розподіленими щодо нуля. Маленькі по абсолютній величині значення звичайно не грають визначальної ролі – їхнє переміщення залишається практично непомітним для нашого сприйняття. У подібних ситуаціях базовий зсув інтервалу x_0 має сенс вибирати рівним $\Delta/2$. При такому виборі значення в околі $[-\Delta/2, \Delta/2)$ будуть замінятися нулями (так звана мертва зона). З метою поліпшення схеми квантування розмір мертвої зони збільшують, залишаючи незмінними розміри інших інтервалів. Крім зменшення довжини коду, це дозволяє спростити процедуру квантування: якщо вибрати розмір мертвої зони рівним 2Δ , визначення номера інтервалу при квантуванні зведеться до ділення квантованого значення на параметр квантування. Дане рішення використовується в багатьох практичних додатках, зокрема, у стандартах кодування відеоінформації JPEG, JPEG2000, MPEG, H.261, H.263.

Інформаційні спотворення, що виникають у результаті скалярного квантування, зазвичай помітно погіршують якість зображення, роблячи його східчастим. В ідеалі спотворення не повинні приводити до істотної зміни основного зображення й не повинні вносити в зображення нові, що раніше не існували деталі. Із цієї причини доцільно квантувати не самі значення колірної матриці, а деякі їхні узагальнені характеристики, зміна яких не веде до кардинальної зміни самого зображення. Як показує практика, найбільшою ефективністю володіють методи, у яких як об'єкт квантування виступають параметри двомірних інтерполяційних функцій, що використовуються для наближеного опису зображень.

Застосування інтерполяції саме по собі є ефективним методом одержання ощадливих подань інформації. Ефективність залежить від кількості вільних параметрів інтерполяційної функції й від того, наскільки точно ця функція наближає закодовані дані до реальних. На практиці найбільше часто використовується особливий різновид інтерполяційного методу – базисне розкладання.

2.2 Побудова інформаційного критерію

При обробці багатомірних даних у переважній більшості випадків вирішується завдання їхнього агрегування. В якості агрегатів зазвичай використовуються середні величини, зокрема, середні статистичні. Серед останніх особливими властивостями володіють середнє арифметичне й середнє гармонійне. Їхнє відношення має властивості ентропії, значення якої можна інтерпретувати як міру невизначеності у виборі елементів масиву. Однак тут нас буде цікавити інший аспект зазначеного відношення, а саме – відношення як міра структурних розходжень значень компонент одномірного масиву. Оскільки розходження лежать в основі поняття інформації, то природною є спроба використання цієї міри для побудови інформаційного критерію. Для формального викладу зробимо необхідні позначення.

Нехай X матриця з позитивними елементами

$$X \in \mathbf{R}_+^{n \times m} = \{X = (x_{ij}) \mid x_{ij} > 0 \ i = \overline{1, n}; \ j = \overline{1, m}\},$$

що має n рядків і m стовпців і нехай

$$\mathbf{p} \in \Delta_n = \{\mathbf{p} = (p_1, p_2, \dots, p_n)^T \mid \sum_{i=1}^n p_i = 1, p_i \geq 0, \forall i = \overline{1, n}\}.$$

Для кожної матриці $X \in \mathbf{R}_+^{n \times m}$ можна визначити функцію

$$H(\mathbf{p}, X) = \ln(\mathbf{p}^T X \cdot X^{-T} \mathbf{p}) - \ln m, \quad (2.1)$$

де X^{-T} – транспонована матриця, елементи якої є зворотними значеннями елементів матриці X .

При фіксованих значеннях елементів матриці X функція (2.1) буде залежати від компонентів вектора \mathbf{p} . Щоб підкреслити цей факт, будемо для $H(\mathbf{p}, X)$ використовувати також позначення $H(\mathbf{p})$. Можна показати, що $H(\mathbf{p})$ має основні властивості ентропії. Якщо рядки матриці X різні за

структурою значень своїх елементів, то максимальне значення (2.1) досягається на векторі \mathbf{p}^* , відмінному від рівномірного

$$\mathbf{p}^o = \left(\overbrace{\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n}}^n \right)^T = \frac{1}{n} \mathbf{e}.$$

Компоненти вектора \mathbf{p}^* мають цікаву властивість: два максимальні за своїми значеннями компоненти відповідають двом об'єктам (рядкам матриці X), які найбільшою мірою відрізняються один від одного структурою значень своїх елементів (найбільш деструктивні). Цю властивість вектора \mathbf{p}^* можна використати при розробці алгоритмів класифікації.

Можливість використання властивостей вектора \mathbf{p}^* для рішення завдання кластеризації квітів ірису на три класи – *virginic*, *versicol* й *setosa* покажемо на класичному прикладі Фішера. На рис.2.1 показаний графік значень компонент вектора \mathbf{p}^* для всієї вибірки. Як видно найбільші значення компонентів вектора \mathbf{p}^* відповідають двом елементам: [7,2 3,6 6,1 2,5] ; [4,3 3,0 1,1 0,1].

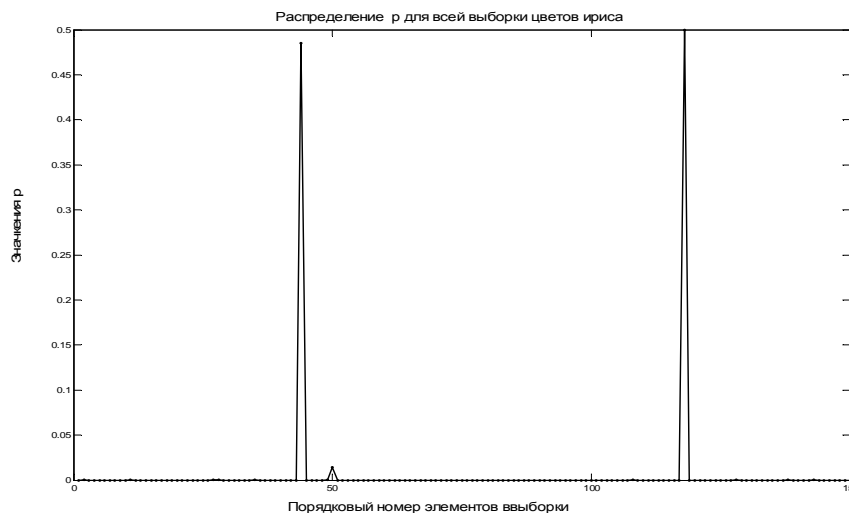


Рисунок 2.1 Графік зміни компонент вектора \mathbf{p}^* для всієї вибірки характеристик квітів ірису.

На першому кроці з використанням міри відстані

$$\rho(\mathbf{x}, \mathbf{y}) = \ln(\mathbf{x}^{-T} \mathbf{y}) + \ln(\mathbf{y}^{-T} \mathbf{x}) \quad (2.2)$$

виявилось можливим розбити всю сукупність характеристик квітів ірису на два кластери по ступені їхньої близькості до виділених елементів по алгоритму. Отримані в результаті кластери, перший – квіти *virginic* й *versicol*, другий - квіти *setosa*. На другому кроці після проведення аналогічної процедури для даних першого кластера спостерігалися 9 помилок, які розподілилися між класами (квітами) як 5 до 4.

Загальні результати класифікації, представлені у вигляді матриці перемішування

$$P = \begin{pmatrix} 45 & 5 & 0 \\ 4 & 46 & 0 \\ 0 & 0 & 50 \end{pmatrix}.$$

Їх можна віднести до одних із кращих результатів кластеризації квітів ірису, що демонструє можливість використання інформаційних властивостей вектора \mathbf{p}^* для побудови алгоритмів класифікації багатомірних даних. Можна продовжити дослідження в цьому напрямку для різних вирішальних правил у процедурах класифікації, однак, залишаючи осторонь деталі, підкреслимо, що запропонований спосіб виявлення розходжень є основою для розробки ефективних алгоритмів розпізнавання.

Повернемося до основної мети нашого дослідження з формування інформаційного критерію. Розглянемо окремий випадок, коли матриця X в (2.1) складається з одного вектора-стовпця $\mathbf{x} = (x_1, x_2, \dots, x_n)^T \in \mathbf{R}_+^{n \times 1}$. Для цього випадку представимо (2.1) у вигляді

$$h(\mathbf{x}) = \ln(\mathbf{x}^{-T} \mathbf{p} \cdot \mathbf{p}^T \mathbf{x}) = \ln \left(\sum_{i=1}^n \frac{p_i}{x_i} \cdot \sum_{i=1}^n p_i x_i \right). \quad (2.3)$$

Легко перевірити, що

$$1. h(\mathbf{x}) = h(\lambda \mathbf{x}) \quad \forall \lambda > 0;$$

$$2. h(\mathbf{x}) = 0 \quad \forall \mathbf{x} = \lambda \cdot \mathbf{e} = \lambda \cdot \overbrace{(1, 1 \dots 1)}^n, \quad \lambda > 0; \quad (2.4)$$

$$3. h(\mathbf{x}) = h(\mathbf{x}^{-1}).$$

На підставі цих властивостей будемо функцію (2.3) називати мірою розходжень компонент вектора \mathbf{x} або інформаційною мірою, оскільки, там, де є розходження, там є й інформація. Далі, $\forall \mathbf{w} \in \Omega \subset \mathbf{R}_+^{n \times 1}$ визначимо

$$h(\mathbf{x}/\Omega) = \min_{\mathbf{w} \in \Omega} \ln((\mathbf{p}./\mathbf{w})^{-T} \mathbf{x} \cdot \mathbf{x}^{-T} (\mathbf{p} \cdot \mathbf{w})) \quad (2.5)$$

і шуканий інформаційний критерій

$$I(\mathbf{x}) = h(\mathbf{x}) - h(\mathbf{x}/\Omega) = \ln(\mathbf{p}^T \mathbf{x} \cdot \mathbf{x}^{-T} \mathbf{p}) - \min_{\mathbf{w} \in \Omega} \ln((\mathbf{p}./\mathbf{w})^{-T} \mathbf{x} \cdot \mathbf{x}^{-T} (\mathbf{p} \cdot \mathbf{w})), \quad (2.6)$$

де $(./)$ і (\cdot) – позаелементні операції ділення й множення відповідно.

Область Ω визначається умовами розв'язуваного завдання. Приведемо приклади використання формули (2.6). Почнемо із простого випадку. Нехай перетворення полягає в заміні вихідного вектора $\mathbf{x} = (x_1, x_2, \dots, x_n)^T \in \mathbf{R}_+^{n \times 1}$ вектором $\mathbf{y} = (y_1, y_2, \dots, y_n)^T \in \mathbf{R}_+^{n \times 1}$. Тоді область обмежень Ω визначиться співвідношеннями $\Omega = \{\mathbf{w} \in \mathbf{R}_+^{n \times 1} \mid w_i = y_i \quad \forall i = \overline{1, n}\}$, а величина $I(\mathbf{x})$ – рівністю:

$$I(\mathbf{x}) = \ln(\mathbf{p}^T \mathbf{x} \cdot \mathbf{x}^{-T} \mathbf{p}) - \ln((\mathbf{p}./\mathbf{y})^T \mathbf{x} \cdot \mathbf{x}^{-T} (\mathbf{p} \cdot \mathbf{y})), \quad (2.7)$$

тут необхідно, щоб максимально можлива кількість інформації містилася в тотожному перетворенні й дорівнювала:

$$I(\mathbf{x}) = \ln(\mathbf{p}^T \mathbf{x} \cdot \mathbf{x}^{-T} \mathbf{p}), \quad (2.8)$$

Розглянемо більше складні й практично важливі приклади.

2.3. Оцінка інформативності ознак

У теорії розпізнавання образів оцінку інформативності ознаки одержують як відношення результатів розпізнавання об'єктів контрольної вибірки в повному просторі ознак до результатів розпізнавання, проведеного без обліку оцінюваної ознаки. Із цього визначення необхідно, щоб оцінка інформативності ознаки, залежала від вирішального правила. Крім того, ця оцінка залежить від обсягу навчальної вибірки. Як показано, для одержання її достовірного значення, об'єктів у кожному класі повинно бути в десятки разів більше кількості досліджуваних ознак.

З точки зору змісту поняття "інформативність", можна дати наступне визначення: інформативна ознака – це ознака, що має близькі значення на елементах (об'єктах) одного класу й істотно різні значення на елементах різних класів.

Звідси необхідно для ефективного вирішення завдання розпізнавання в алгоритмах класифікації перейти до використання ознак, що володіють вказаною властивістю. Область припустимих значень Ω визначимо в такий спосіб. Представимо всю сукупність елементів навчальної вибірки, яка є відомою, у вигляді рядків матриці $X \in \mathbf{R}_+^{n \times m}$. Нехай k – кількість розпізнаваних класів, $\xi(x_{ij})$ – номер класу, якому відповідає значення x_{ij} j -ї ознаки на i -ому елементі вибірки. Тоді інформативність j -ї ознаки (стовпця X_j матриці X) можна оцінити на основі рішення завдання (2.6) з областю визначення Ω у вигляді

$$\Omega = \{ \mathbf{w} \in \mathbf{R}_+^{n \times 1} \mid w_u = w_v \Leftrightarrow \xi(x_{uj}) = \xi(x_{vj}) \forall j = \overline{1, m} \}. \quad (2.9)$$

Відзначимо, що обумовлена в такий спосіб інформативність ознаки $I(X_j)$ не залежить від одиниць його виміру й враховує тільки відносні значення розподілу ознак на елементах класів розпізнавання.

Роботоздатність запропонованого методу покажемо при рішенні ряду задач. Оцінимо інформативність чотирьох ознак квітів ірису при розбивці їх на 3 класи (продовження вище наведеного приклада). Область (2.9) буде складатися із всіх векторів w , для компонентів яких виконуються співвідношення:

$$w_1 = w_i \forall i = \overline{1,50}; w_{51} = w_i \forall i = \overline{51,100}; w_{101} = w_i \forall i = \overline{101,150}.$$

У таблиці 2.1 представлені результати оцінки інформативності ознак квітів ірису, а на рисунку 2.2 – графік значень їхніх характеристик. Низька інформативність перших двох ознак обумовлена їхньою невеликою варіативністю, тоді як для останніх двох, навпаки, спостерігається висока варіативність.

Таблиця 2.1 – Результати оцінки інформативності ознак квітів ірису

Властивості ознаки	Ознаки			
	чашилисток		маточка	
	довжина	ширина	довжина	ширина
Інформативність	0,0126	0,0079	0,3205	0,8158
Відносний діапазон зміни	0,6116	0,3980	5,0146	12,1539

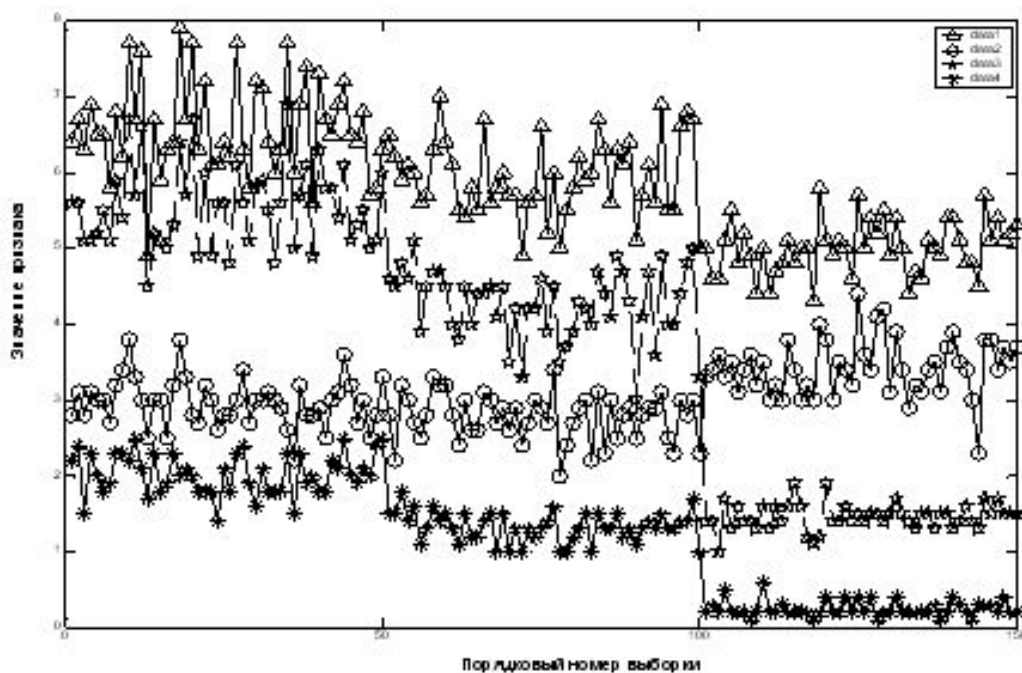


Рисунок 2.2 – Графіки зміни ознак квітів ірису

В останньому рядку таблиці представлена величина відносного діапазону зміни ознаки, обумовлена як сума відносин модуля попарних різниць середніх значень ознаки в класі до їх мінімального середнього значення. Як видно, наведені оцінки інформативності добре погодяться з оцінками варіативності ознак.

2.4 Оптимальна градація яскравостей зображень

Дуже часто, у завданнях класифікації й розпізнавання образів, ознаки, що описують об'єкти спостереження мають різну природу, наприклад, кількісні і якісні. Їхнє спільне використання при класифікації даних, як правило, пов'язане із серйозними труднощами. У зв'язку із цим виникає завдання перетворення кількісних ознак у якісні, або іншими словами, завдання розбивки кількісних ознак на градації. Причому така розбивка повинне бути оптимальною з погляду потреб розв'язуваного завдання. У даній роботі пропонується метод градації ознак на основі інформаційного критерію (2.6). Це завдання складніше, ніж визначення ознак розпізнавання, оскільки її рішення передбачає не тільки визначення значень критерію (2.6), але й визначення значень порогів градації. Залишаючи осторонь деталі, намітимо шлях рішення цього завдання й приведемо приклади.

Нехай $\mathbf{x} = \{x_i\}_1^n \in \mathbf{R}_+^{n \times 1}$ – вектор-стовбець дійсних позитивних чисел упорядкованих по зростанню. Потрібно розбити всі n його значень за ступенями близькості на k груп по n_s значень у кожній ($s = \overline{1, k}$, $n_1 + n_2 + \dots + n_k = n$). Позначимо $\sum n_s = 1 + \dots + n_s$. Тоді для завдання (2.7) область Ω є:

$$\Omega = \{\mathbf{w} \in \mathbf{R}_+^{n \times 1} \mid w_i = w_{\sum n_s} : w_{\sum n_{s-1}+1} \leq w_i < w_{\sum n_s} \quad \forall s = \overline{1, k}\}.$$

З формули (2.6) видно, що як цільова функція використовується функція:

$$f(\mathbf{w}) = \sum_{i=1}^n \frac{x_i}{w_i} \cdot \sum_{i=1}^n \frac{w_i}{x_i},$$

мінімізація якої по області Ω дозволяє легко визначити екстремальне значення $\mathbf{w}^* = \arg \min_{\mathbf{w} \in \Omega} f(\mathbf{w})$. Однак область Ω залежить від значень Σ_{n_s} , обумовлених порядковими номерами порогів градацій $x_{\Sigma_{n_s}}$ $s = \overline{1, k}$. Ці пороги можна знайти з умов мінімізації їхніх внесків у значення цільової функції $f(\mathbf{w})$. Зазначені внески v_n визначаються з наступного очевидного співвідношення:

$$\sum_{i=1}^{n+1} u_i \cdot \sum_{i=1}^{n+1} \frac{1}{u_i} - \sum_{i=1}^n u_i \cdot \sum_{i=1}^n \frac{1}{u_i} = \Sigma_1 \cdot \Sigma_2 - (\Sigma_1 - u_n) \left(\Sigma_2 - \frac{1}{u_n} \right) = \frac{\Sigma_1}{u_n} + \Sigma_2 u_n - 1 = v_n \quad (2.10)$$

Легко побудувати алгоритм визначення значень $x_{\Sigma_{n_s}}$ на основі методу динамічних скупчень й оцінок внесків (2.10). З використанням цього підходу, розроблено ефективний метод градації значень. Його працездатність покажемо на конкретних прикладах.

У табл.2.2 наведені результати градацій цих чисел. Відзначимо, що при $k=2$ відносна величина порогу 0,21 близька до золотого перетину 0,168.

Як видно з табл. 2.2, результати градації за інформаційним критерієм у порівнянні з рівномірним розподілом зміщені вліво. Це можна пояснити тим, що значення цільової функції залежить від відносного а не абсолютного збільшення аргументів.

Таблиця 2.2 – Результати градації перших 100 чисел натурального ряду

Кількість градацій до	Граничні значення $x_{\Sigma_{n_s}}$
2	$\frac{21}{50}$
3	$\frac{[10; 40]}{[33; 66]}$
4	$\frac{[7; 24; 54]}{[25; 50; 75]}$
5	$\frac{[5; 16; 35; 62]}{[20; 40; 60; 80]}$
10	$\frac{[3; 9; 17; 27; 38; 50; 62; 74; 87]}{[10; 20; 30; 40; 50; 60; 70; 80; 90]}$

На рис. 2.3 представлені результати 4-х рівневої (2 біти) градації яскравостей пікселів зображення "Роза". В цілях співставлення на всіх 4-х зображень (включаючи вихідне) сума яскравостей пікселів однакова.

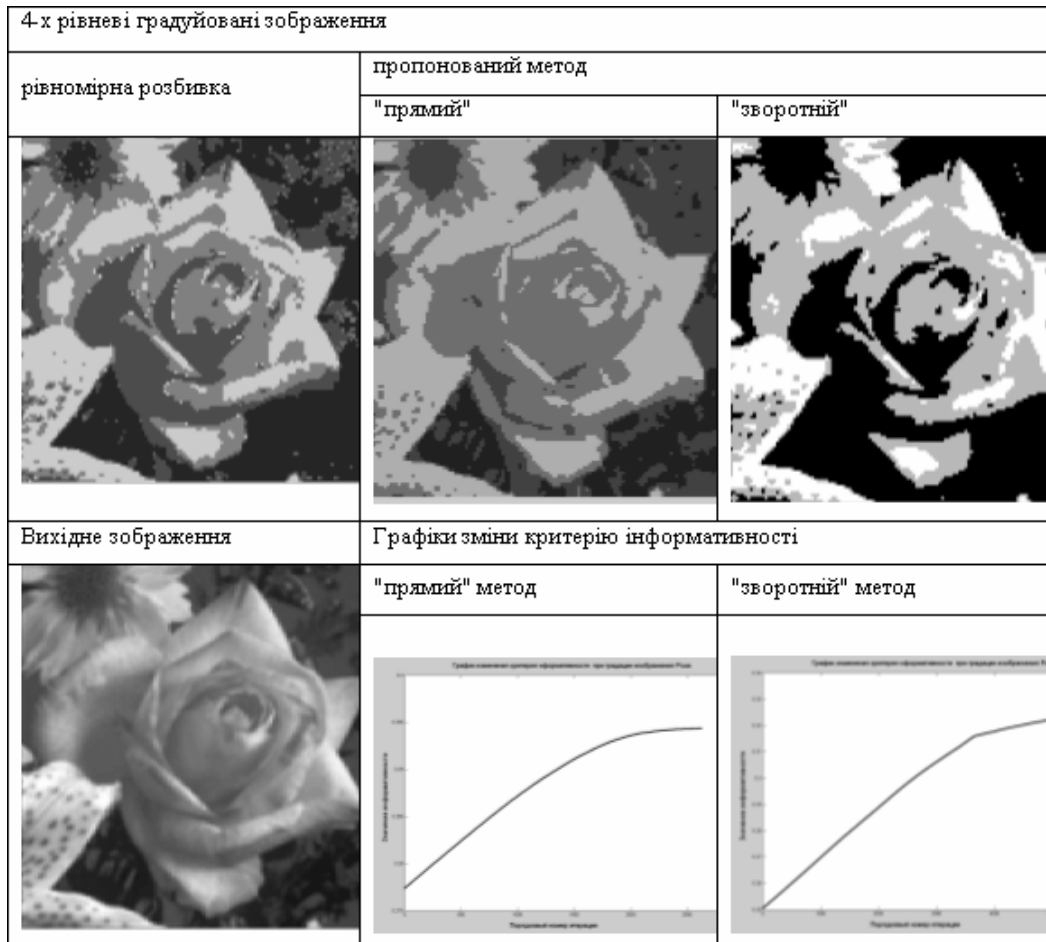


Рисунок 2.3. Результати градації зображення "роза".

Як видно, при рівномірному, не оптимальному, розбитті спостерігаються нечисленні артефакти, "прямий" метод показав гарну чутливість до малих значень пікселів, "зворотний" – виділив всі яскраві ділянки вихідного зображення.

Таким чином, запропонований інформаційний критерій дозволяє одержати інформацію про структуру значень компонент елементів (рядків, стовпців) масиву аналізованих даних, що може бути використана для

рішення різних завдань їхнього аналізу – класифікації, оцінки інформативності ознак та їхньої градації.

РОЗДІЛ 3. АНАЛІЗ ЕНТРОПІЙНИХ ОЦІНОК

3.1. Оцінки ентропії Хартлі, Колмогорова, 3σ та Шенона

Як практичну міру ентропії дискретного джерела інформації Р. Хартлі [18] запропонував функцію логарифма кількості можливих станів ДІ:

$$H = \log S^n = n \cdot \log S ; \quad (3.1)$$

де H – ентропія;

S – число незалежних рівноймовірних станів ДІ;

n – довжина вибірки.

Більшість технологічних об'єктів, які в даному випадку розглядаються як стохастичні ДІ, формують дані, що підлягають гаусовому закону розподілу імовірностей. Для таких ДІ переважна більшість відліків лежить в діапазоні 3σ в околі математичного сподівання. Відповідно, ентропія буде визначатися згідно виразу:

$$H = \log 3\sigma . \quad (3.2)$$

При кодуванні неперервних ДІ із заданою точністю квантування по рівню і кроці дискретизації по часу А. Н. Колмогоровим [19] запропонована нова міра інформації – ε -ентропія, яка визначається кількістю елементів ε при переході ДІ в різні стани:

$$H_\varepsilon \leq \frac{T}{\Delta t} + \log \frac{C}{\varepsilon} ; \quad (3.3)$$

де Δt – крок дискретизації, що забезпечує точність квантування ε ;

C – діапазон квантування;

T – інтервал часу спостереження ДІ.

Якщо визначити кількість значень, яке може бути отримане в F -просторі за час T у вигляді:

$$\varphi(t) = 2^{\frac{T}{\Delta t}};$$

отримаємо:

$$H_\varepsilon \leq \log \left(\frac{C}{\varepsilon} \cdot 2^{\frac{T}{\Delta t}} \right).$$

В частковому випадку, коли $\frac{C}{\varepsilon} = 2^m$ і $\frac{T}{\Delta t} = 2^n$

$$H_\varepsilon \leq \log_2(2^m \cdot 2^n) = m + n.$$

Оцінки ентропії ДІ у вигляді міри Хартлі, 3σ і ε -ентропії Колмогорова вирішуються в цілих числах в тому випадку, коли діапазон квантування станів ДІ вибирається кратним цілому степеню числа два. В іншому випадку, коли $S \neq 2^k$ ($k = 1, 2, 3, \dots$) необхідно користуватися оцінками:

$$H = n \cdot \hat{E}[\log S] = n \cdot \lceil \log S \rceil;$$

$$H = \hat{E}[\log 3\sigma];$$

$$H_\varepsilon = \hat{E} \left[\log \left(\frac{C}{\varepsilon} \cdot 2^{\frac{T}{\Delta t}} \right) \right]; \quad (3.4)$$

де \hat{E} – символ цілочисельної функції з округлення до більшого цілого;

$\lceil \rceil$ – ознака операції округлення до більшого цілого.

Приведені оцінки ентропії ДІ базуються на умові, що кожен S_j стан джерела кодується n -розрядним двійковим кодом однакової довжини.

Розглянуті оцінки відповідають ДІ з рівноймовірними станами і, як правило, є максимальними, оскільки не враховують надлишковість ДІ за рахунок не рівноймовірності розподілу різних станів S_j ДІ.

Для ДІ з нерівноймовірними станами К. Шеноном [20] введена міра ентропії

$$H = -k \cdot \sum_{i=1}^n p_i \cdot \log p_i ; \quad (3.5)$$

де k – додатній коефіцієнт, що враховує основу логарифма; p_i – ймовірність S_j -го стану дискретного ДІ.

Були також здійснені спроби використання теорії інформації для оцінки вимірних значень, тобто кількісної характеристики фізичної величини або, в більш широкому змісті, джерела інформації.

3.2 Ентропійні оцінки Лонго, Шульца, Олівера, Мідлтона, Таллера, Боюна, Лапи, Голдмана

Дж. Лонго [21] запропонував розширити шенонівську теорію інформації введенням “якісних” параметрів, які б характеризували корисність інформації, тоді як шенонівське визначення використовує тільки імовірнісні (повторюваність) стану ДІ без будь-якої семантики. Кожному стану ДІ Дж. Лонго ставить у відповідність коефіцієнти корисності $u_i \geq 0$.

Розподіл коефіцієнтів корисності не залежить від розподілу імовірностей, а визначається з семантичних міркувань. Тоді кількість інформації буде визначатися за виразом:

$$I(u, p) = -k u \cdot \log p ;$$

де k – стала величина;

$p = p_i$ – імовірність S_j -го стану.

Ентропія корисності визначається як середнє статистичне значення усіх можливих повідомлень, з врахуванням коефіцієнтів u_i :

$$H(u, p) = -k \sum_{i=1}^n [u_i p_i \cdot \log p_i] . \quad (3.6)$$

Якщо всі $u_i = const$, ентропія корисності переходить в звичайну шенонівську ентропію ДІ.

Для некорисних станів, імовірність яких $p_i > 0$, вважаються $u_i = 0$. Якщо з умови $u_i > 0$ випливає, що $p_i = 0$, то це означає, що корисні стани не можливі.

Г. Шульц [22] намагався узагальнити шенонівську ентропію і тим самим створити поняття оціночної ентропії. Він видозмінив ентропію як середнє значення об'єму інформації $\log \frac{1}{p_i}$ окремих станів, щоб її можна було б використовувати не тільки для передачі, але й для оцінки значимості інформації. Шульц встановив оціночну шкалу, по якій найбільш «несприятливому» стану надавалось найнижче значення, а найбільш сприятливому стану – найвище. Для зручності користування значення шкали вибиралися кратними степеню числа 2.

$$W_0 = 2^0 = 1, W_1 = 2^1 = 2, W_2 = 2^2 = 4, \dots, W_n = 2^n.$$

З цих значень утворювалися оціночні коефіцієнти

$$\text{від } w_0 = \frac{W_0}{\sum_{i=1}^n W_i} \text{ до } w_n = \frac{W_n}{\sum_{i=1}^n W_i}.$$

Вони еквівалентні послідовності частот p_i .

Для забезпечення додаткової умови, яка вимагає, щоб при однаковій семантичній оцінці всіх станів ДІ утворювалась би шенонівська ентропія, використовується формула:

$$H(p, w) = H(p_1, p_2, \dots, p_n, w_1, w_2, \dots, w_n) = - \sum_{i=1}^n \left[\frac{p_i w_i}{\sum_{j=1}^n p_j w_j} \cdot \log \frac{p_i w_i}{\sum_{j=1}^n p_j w_j} \right]. \quad (3.7)$$

Легко помітити, що функція $H(p, w)$ симетрична відносно змінних p і w . Для $p = const$ вона приймає вигляд:

$$H(w) = -\sum_{i=1}^n w_i \log w_i ;$$

а для $p_1 = p_2 = \dots = p_n = w_1 = w_2 = \dots = w_n$ має місце граничний випадок:

$$H(n) = -n \sum_{i=1}^n \log \frac{1}{n} = \log n ;$$

де n – кількість можливих станів ДІ, що співпадає з оцінкою Хартлі. Функцію $H(p, w)$ можна виразити через абсолютні значення W_i і абсолютні частоти P_i .

$$H(P, W) = -\sum_{i=1}^n \left[\frac{P_i W_i}{\sum_{j=1}^n P_j W_j} \right] \cdot \log \frac{P_i W_i}{\sum_{j=1}^n P_j W_j} . \quad (3.8)$$

При цьому вираз:

$$H(p, w) = H(P, W) = H(v) = -\sum_{i=1}^n v_i \log v_i$$

формально відповідає шенонівській ентропії:

$$H(p) = -\sum_{i=1}^n p_i \cdot \log p_i$$

Необхідно відмітити, що в виборі оціночних коефіцієнтів існує певна довільність, яка вносить суб'єктивність в оцінку.

Для дискретного ергодичного ДІ, в якого кореляційні зв'язки існують тільки між двома сусідніми значеннями послідовності x_1, x_2, \dots, x_n ентропія визначається виразом:

$$H(x) = -\sum_j p(x_j) \cdot H_i = -\sum_{i=1}^n p(x_i) \sum_{j=1}^n p\left(\frac{x_j}{x_i}\right) \cdot \log p\left(\frac{x_j}{x_i}\right)$$

або

$$H(x) = -\sum_{j=1}^n \sum_{i=1}^n p(x_i, x_j) \cdot \log p(x_j/x_i);$$

де $p(x_j/x_i)$ – умовна імовірність появи значення x_j при умові, що попереднім було значення x_i ;

$p(x_i, x_j)$ – сумісна імовірність появи пари символів x_i, x_j .

В загальному випадку ентропія ергодичних ДІ визначається виразом:

$$H(x) = -\sum_{i=1}^n \sum_{j=1}^n \dots \sum_{s=1}^n p(x_i, x_j, \dots, x_s) \cdot \log p(x_i/x_j/\dots/x_s), \quad (3.9)$$

де $p(x_i, x_j, \dots, x_s)$ – імовірність послідовності станів i, j, \dots, s ; n – загальна кількість станів джерела; $p(x_i/x_j/\dots/x_s)$ – умовна імовірність появи i -го стану при умові, що попередніми були j, \dots, s стани, при чому j -ий передував i -му, а s – найбільш віддалений стан, який має кореляційний зв'язок з i -м.

Для джерел з незалежними, але нерівноймовірними станами Б.Олівером [23] отримана оцінка загальної кількості можливих комбінацій станів:

$$N = \frac{n!}{\prod_j S_j}; \quad (3.10)$$

або в логарифмічному вигляді:

$$\log N = \log n! - \sum_j S_j;$$

яке на основі наближення Стірлінга:

$$\log N \rightarrow \log \sqrt{2\pi n} + n \cdot \log n - n \sum_{j=1}^n \log \sqrt{2\pi S_j} - n \sum_{j=1}^n p(j) \cdot \log p(j) - n \cdot \log n + n$$

зводиться до виразу:

$$H = \lim \frac{\log N}{n} = -\sum p(j) \cdot \log p(j), \quad (3.11)$$

що співпадає з шенонівським визначенням ентропії.

Д. Мідлтон [24] також досліджував дискретні ДІ, які формують послідовність символів довільної довжини, розподілених в визначеному

порядку в часі. Для реалізації дискретної випадкової послідовності $X = \{x_i\}$, кожен із символів x_i якої може приймати одне з l_i різних значень ($l \leq l_i \leq L$; $i = 1, 2, \dots, n$), отримано вираз для апіорної невизначеності $x_1, \dots, x_i, \dots, x_n$

$$H(X) = -\sum_{l_1}^L \dots \sum_{l_n}^L p(X) \log p(X); \quad (3.12)$$

де додавання проводиться по всім можливим значенням кожного з символів x_i послідовності.

Для ДІ з статистично залежними станами Д. Мідлтон визначив вираз середньої умовної ентропії:

$$H(X/Y) = \sum_{l_1}^L \dots \sum_{l_n}^L \sum_{m_1}^M \dots \sum_{m_k}^M p(x_1, \dots, x_n, y_1, \dots, y_m) \cdot \log p(x_1, \dots, x_n / y_1, \dots, y_m); \quad (3.13)$$

де x_i, y_i – статистично залежні стани ДІ. З останнього виразу випливає, що для розрахунку ентропії таких ДІ необхідно знати сумісні щільності імовірностей різного порядку $W_1(y_1, t_1), W_1(y_1, t_1; y_2, t_2), \dots, W_1(y_1, t_1; \dots; y_n, t_n)$.

При вивченні статистичних моделей показано, що на практиці ДІ не є настільки статистично складними, щоб їх описувати багатомірними розподілами.

Зокрема, для повного опису ергодичних стаціонарних ДІ достатньо знати двомірні розподіли і відповідно статистично середні. Природно, що ентропія і швидкість створення повідомлень такими джерелами, за рахунок кореляційних зв'язків між різними послідовностями символів і нерівномірності розподілу імовірності символів, виявляються меншими в порівнянні з оцінкою інформаційної міри Хартлі (3.1).

В роботі В. Таллера [25] підкреслюються переваги, які можна отримати шляхом кореляційного аналізу і усунення внутрішньої кореляції повідомлень, що формуються джерелом. Показано, що якщо в деякий момент часу ДІ, що має S станів, переходить тільки в S_j можливих станів, то аналіз дійсного

інформаційного змісту його повідомлень приводиться до меншого об'єму інформації в порівнянні з функцією найбільшої інформації:

$$H \leq k2BT \left(1 + \frac{S}{N} \right); \quad (3.14)$$

де $H = k \cdot n \log S_{ave}$;

S_{ave} – середнє значення станів ДІ;

BT – інформаційна база повідомлень, що формується;

N – значення шуму.

При цьому встановлені вимоги найбільшої інформації базуються на твердженні, що неможливо аналітичне продовження функції інформації з точністю більшою ніж $\frac{1}{S}$, на протязі інтервалу між відліками.

Таким чином, величина $\frac{1}{S}$, як показано в роботі В. Талера, набуває змісту інтервалу кореляції між відліками відповідних станів ДІ.

Досліджуючи оцінки інформації дискретних випадкових величин Боюн [26, 27] запропонував величину δ -ентропію, яка характеризує невизначеність значення випадкового процесу і його динаміку. Приведена оцінка δ -ентропії визначається виразом:

$$h_{\Delta} = \frac{|f'_{cep}(t)|}{|f'_{max}(t)|}; \quad (3.15)$$

де $f'_{cep}(t)$, $f'_{max}(t)$ – відповідно середнє і максимальне значення похідних зміни кількості станів джерела.

Приведена δ -ентропія не залежить від вибраних кванта по рівню і кроку дискретизації, а повністю визначається законом розподілу похідних.

В роботі Л. Г. Лапи [28] відмічено, що для визначення ймовірнісних показників джерел інформації (ДІ) необхідні детальні характеристики кореляційних зв'язків між різними символами. Там же підкреслюється, що чим більша кореляція між станами ДІ, тим більш нерівномірний розподіл умовної ймовірності $p(x_i/x_j)$ у виразі визначення ентропії ДІ з залежними станами, що приводить до зменшення ентропії джерела.

Л. Г. Лапа також звертає увагу на те, що функція ентропії Хартлі користується виключно одновимірними розподілами, в той час як КФ використовує двовимірні ймовірнісні розподіли. Тому принципово оцінка ентропії на базі кореляційної функції змістовніша, ніж функція ентропії Хартлі.

До аналогічного твердження приходять Ж. Макс [29], визначаючи зв'язок коефіцієнта кореляції з оцінками ентропії. Ф. Г. Ланге [30] доводить, що середнє значення двовимірної щільності ймовірності (кореляція) несе більше інформації про ентропію, ніж середні одновимірної щільності.

С. Голдман [31] також розширює теорію інформації Шенона, застосовуючи поняття КФ і відмічає, що кореляційний аналіз є зв'язуючим елементом між спектральною теорією і теорією інформації.

3.3. Оцінки ентропії з врахуванням кореляційних властивостей джерела інформації

Складність обчислення багатовимірних розподілів суттєво обмежує можливість використання шенонівських оцінок ентропії для ДІ з незалежними та статистично залежними станами при інженерному розрахунку їх властивостей, особливо якщо дослідження інформаційних параметрів конкретних джерел ведуться комп'ютерною системою в реальному масштабі часу. Тому реалізація безпосереднього зв'язку КФ з розподілами ймовірностей станів і ентропією ДІ є досить перспективною. Такий зв'язок легко встановлюється для неперервних джерел [32].

$$h(X, Y) = - \int_{-\infty}^{\infty} \omega(x) \log_2 \omega(x) dx - \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \omega(x) \log_2 \omega(y/x) dx dy = h(X) + h(Y/X); \quad (3.16)$$

де $h(X) = - \int_{-\infty}^{\infty} \omega(x) \log_2 \omega(x) dx$ – диференціальна ентропія;

$h(Y/X) = - \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \omega(x, y) \log_2 \omega(y/x) dx dy$ – умовна диференціальна ентропія

повідомлення Y ;

$\omega(x, y)$ – щільність сумісного розподілу x і y ;

$\omega(x)$ – щільність розподілу x ;

$\omega(y/x)$ – умовна щільність розподілу y відносно x .

Багаторазовими дослідженнями [33] реальних об'єктів управління і джерел інформації показано, що стохастичні параметри технологічних процесів на локальних проміжках часу досить точно описуються моделлю гаусового закону розподілу ймовірностей.

Оцінка ентропії джерел з корельованими станами, які мають гаусовий закон розподілу, визначається за виразом [32]:

$$h[x(t), x(t + \tau)] = \log_2 \left(2\pi\sigma_x^2 \sqrt{1 - \rho_{xx}^2(\tau)} \right) + \frac{1}{1 - \rho_{xx}^2(\tau)} \log_2 e - \frac{\rho_{xx}^2(\tau)}{1 - \rho_{xx}^2(\tau)} \log_2 e = \log_2 \left(2\pi e \sigma_x^2 \sqrt{1 - \rho_{xx}^2(\tau)} \right)$$

Чисельний розрахунок ентропії дискретного джерела інформації з нерівноймовірними корельованими станами здійснюється у відповідності з виразом:

$$H[x_i, x_{i+j}] = \log_2 2\pi e + \log_2 \sigma_x^2 + \frac{1}{2} \log_2 [1 - \rho_{xx}^2(j)]; \quad (3.17)$$

де перший елемент є константою інформаційної міри, що пов'язана з типом закону розподілу випадкової величини, другий елемент визначає дисперсію випадкових станів ДІ:

$$\sigma_x^2 = \frac{1}{N-1} \sum_{i=1}^N \left(x_i - \frac{1}{N} \sum_{i=1}^N x_i \right)^2,$$

а третій елемент характеризує взаємну ентропію корельованих нерівноймовірних станів ДІ за допомогою квадрату нормованої функції автокореляції:

$$\rho_{xx}(j) = \frac{\frac{1}{N} \sum_{i=1}^N \left(x_i - \frac{1}{N} \sum_{i=1}^N x_i \right) \cdot \left(x_{i+j} - \frac{1}{N} \sum_{i=1}^N x_i \right)}{\frac{1}{N-1} \sum_{i=1}^N \left(x_i - \frac{1}{N} \sum_{i=1}^N x_i \right)^2}. \quad (3.18)$$

Розрахунок ентропії ДІ на основі оцінки нормованої автокореляційної функції $\rho_{xx}(j)$ є незручним при обчисленні в зв'язку з необхідністю центрування послідовності x_i згідно з (3.4). Для оцінки ентропії ДІ за допомогою структурної функції приведемо вираз (3.3) до вигляду:

$$\begin{aligned} H(x_i, x_{i+j}) &= \log_2 2\pi e \sqrt{D_{\hat{x}}^2 - D_{\hat{x}}^2 \cdot \rho_{xx}^2(j)} = \\ &= \log_2 2\pi e + \frac{1}{2} \log_2 \left(\left[D_{\hat{x}} - R_{xx}(j) \right] \cdot \left[D_{\hat{x}} + R_{xx}(j) \right] \right); \end{aligned} \quad (3.19)$$

$$\text{де } R_{xx}(j) = D_{\hat{x}} \cdot \rho_{xx}(j).$$

Кореляційна функція для стаціонарних процесів легко виражається через коваріаційну функцію $K_{xx}(j)$, яка значно простіша в обчисленні через відсутність центрування.

Відповідно, визначення ентропії зручніше здійснювати через автоковаріаційну функцію згідно наступного виразу:

$$\begin{aligned}
H(x_i, x_{i+j}) &= \log_2 2\pi e \sqrt{D_x^2 - D_x^2 \cdot \rho_{xx}^2(j)} = \\
&= \log_2 2\pi e + \frac{1}{2} \log_2 ([D_x - K_{xx}(j)] \cdot [D_x + K_{xx}(j)]).
\end{aligned}$$

Аналітичний зв'язок між структурною і автокореляційною функціями показаний в [34]. Підставимо, виражене через $C_{xx}(j)$ значення $R_{xx}(j)$, і отримаємо оцінку:

$$H(x_i, x_{i+j}) = \log_2 2\pi e + \frac{1}{2} \log_2 \left[\frac{C_{xx}(j)}{2} \left(2D_x - \frac{C_{xx}(j)}{2} \right) \right]. \quad (3.20)$$

Якщо оцінити середнє, отримаємо робочу формулу міри ентропії:

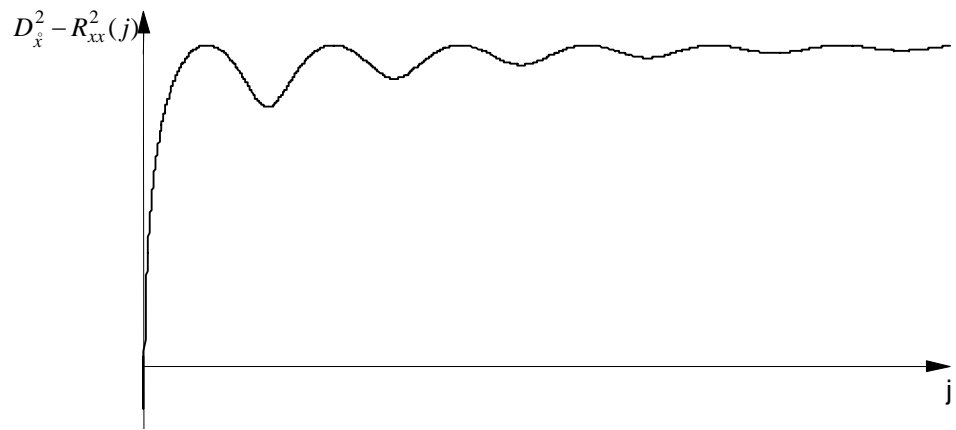
$$\overline{H(x_i, x_{i+j})} = \frac{1}{2T} \sum_{j=1}^T \log_2 (2\pi e)^2 \left[\frac{C_{xx}(j)}{2} \left(2D_x - \frac{C_{xx}(j)}{2} \right) \right]$$

або

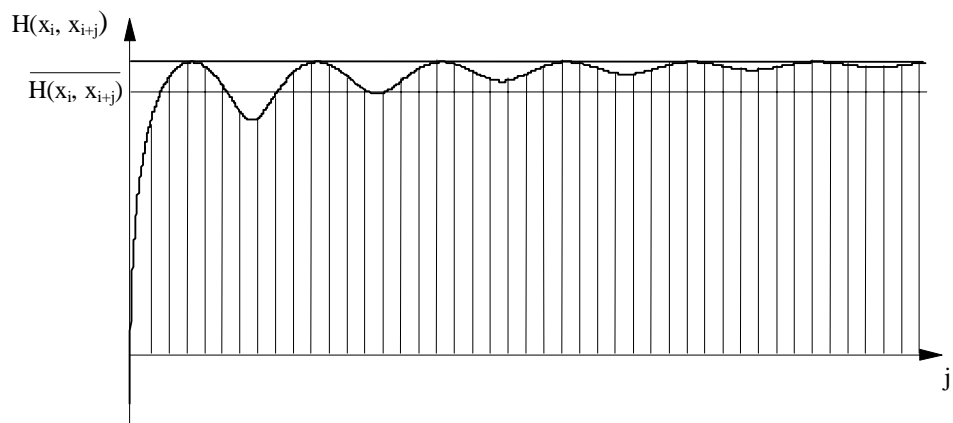
$$\overline{H(x_i, x_{i+j} | x_i)} = \frac{1}{2T} \sum_{j=1}^T \log_2 (2\pi e)^2 \left[C_{xx}(j) \left(D_x - \frac{C_{xx}(j)}{4} \right) \right]; \quad (3.21)$$

де T – інтервал усереднення.

На рис. 3.1, б наведено графіки ентропії $H(x_i, x_{i+j})$ та оцінки ентропії $\overline{H(x_i, x_{i+j})}$ для ДІ з кореляційною функцією $R_{xx}(j) = Ae^{-\alpha(j \cdot \Delta t)} \cos[\omega(j \cdot \Delta t)]$, де α – коефіцієнт згасання; Δt – крок дискретизації; A – амплітуда випадкового процесу; ω – колова частота.



а)



б)

Рис. 3.1. Графік функції $D_x^2 - R_{xx}^2(j)$ – а); графіки ентропії $H_{xx}(x_i, x_{i+j})$ і оцінки ентропії $\overline{H_{xx}(x_i, x_{i+j})}$ – б).

Простіше, порівняно з структурною функцією, обчислюється модульна функція автокореляції $G_{xx}(j)$, яка в випадку гаусових процесів має статистичний зв'язок з функцією кореляції, звідки:

$$\rho_{xx}(j) = 1 - \frac{\pi}{4} g_{xx}(j);$$

де $g_{xx}(j) = \frac{G_{xx}(j)}{\sigma_x^2}$ – нормована модульна функція автокореляції.

Тоді отримаємо оцінку ентропії у вигляді:

$$\overline{H(x_i, x_{i+j})} = \frac{1}{T} \sum_{j=1}^T \log_2 \left[\frac{\pi e \sqrt{\pi}}{2} G_{xx}(j) \cdot \sqrt{8 - \pi g_{xx}^2(j)} \right]; \quad (3.22)$$

яка може бути ефективно використана на практиці в задачах ідентифікації джерел інформації.

Модульна функція має простий аналітичний зв'язок з функцією еквівалентності, на основі якого, з (3.8) отримаємо оцінку ентропії, виражену через функцію еквівалентності $F_{xx}(j)$:

$$\overline{H(x_i, x_{i+j})} = \frac{1}{T} \sum_{j=1}^T \log_2 \left[\pi^2 e \frac{M_x - F_{xx}(j)}{\sigma_x} \sqrt{\frac{8\sigma_x}{\pi} - [M_x - F_{xx}(j)]^2} \right]. \quad (3.23)$$

Аналогічні оцінки можна отримати, якщо використати статистичні зв'язки нормованої кореляційної функції із знаковою $H_{xx}(j)$ та полярною $P_{xx}(j)$ кореляційними функціями:

Відповідно оцінки ентропії будуть мати вигляд:

$$\overline{H(x_i, x_{i+j})} = \frac{1}{T} \sum_{j=1}^T \log_2 \left[2\pi e \sigma_x \cdot \sqrt{1 - \sin^2\left(\frac{\pi}{2} H_{xx}(j)\right)} \right]; \quad (3.24)$$

$$\overline{H(x_i, x_{i+j})} = \frac{1}{T} \sum_{j=1}^T \log_2 \left[\frac{2\pi e}{\sigma_x} \cdot \sqrt{2 - \pi \cdot P_{xx}^2(j)} \right]. \quad (3.25)$$

Ентропійна модель, виражена через $\check{F}_{xx}(j)$, обчислюється з більшою швидкістю, завдяки простішій реалізації функції “менше з двох” ніж операцій множення, віднімання, піднесення до квадрату. Ентропійну модель на основі $H_{xx}(j)$ доцільно використовувати для обробки даних ДІ, які формують цифрові сигнали із значеннями +1 і -1.

Аналіз функції $H(x_i, x_{i+j})$ та оцінок $\overline{H(x_i, x_{i+j})}$ різних дискретних ДІ показує, що при заданому діапазоні квантування станів максимальною ентропією характеризуються дискретні послідовності максимальної довжини.

Це добре узгоджується з логічними уявленнями про інформаційну ємність дискретних послідовностей і властивостями сигнальних моделей ДІ, що

дає можливість достатньо ефективно використовувати при інженерних розрахунках введені міри (3.21) – (3.25).

Реалізуючи обчислення оцінки $\overline{H}(x_i, x_{i+j})$ на ковзному інтервалі можна алгоритмічно просто здійснити ідентифікацію інформаційних станів технологічних об'єктів контролю та управління в реальному масштабі часу.

Даний клас моделей, по відношенню до кореляційних, здійснюють інтегральну оцінку імовірності переходів між станами і завдяки представленню в логарифмічному просторі забезпечують менші об'єми даних.

Таким чином, оберемо ентропійну оцінку еквівалентності (3.23), що дає можливість більш точно здійснити оцінку ентропії за рахунок врахування кореляційних властивостей між пікселями зображення і як остаточний результат – забезпечує більше стиснення відеоданих.

РОЗДІЛ 4

РОЗРОБКА ЕФЕКТИВНОГО АЛГОРИТМУ КОДУВАННЯ ЗОБРАЖЕНЬ НА ОСНОВІ ЗМІНИ ГРАДАЦІЇ ЯСКРАВОСТІ

4.1 Загальна структура та алгоритм програми

Для написання програми було обрано мову програмування Delphi 7.0, тому що вона містить всі необхідні компоненти для створення віконних додатків, а також багато модулів для роботи із зображеннями.



Рисунок 4.1 – Загальна структура програми.

Розробимо схему алгоритму основної програми, в якій будемо використовувати функції, приведені нижче у розділі:

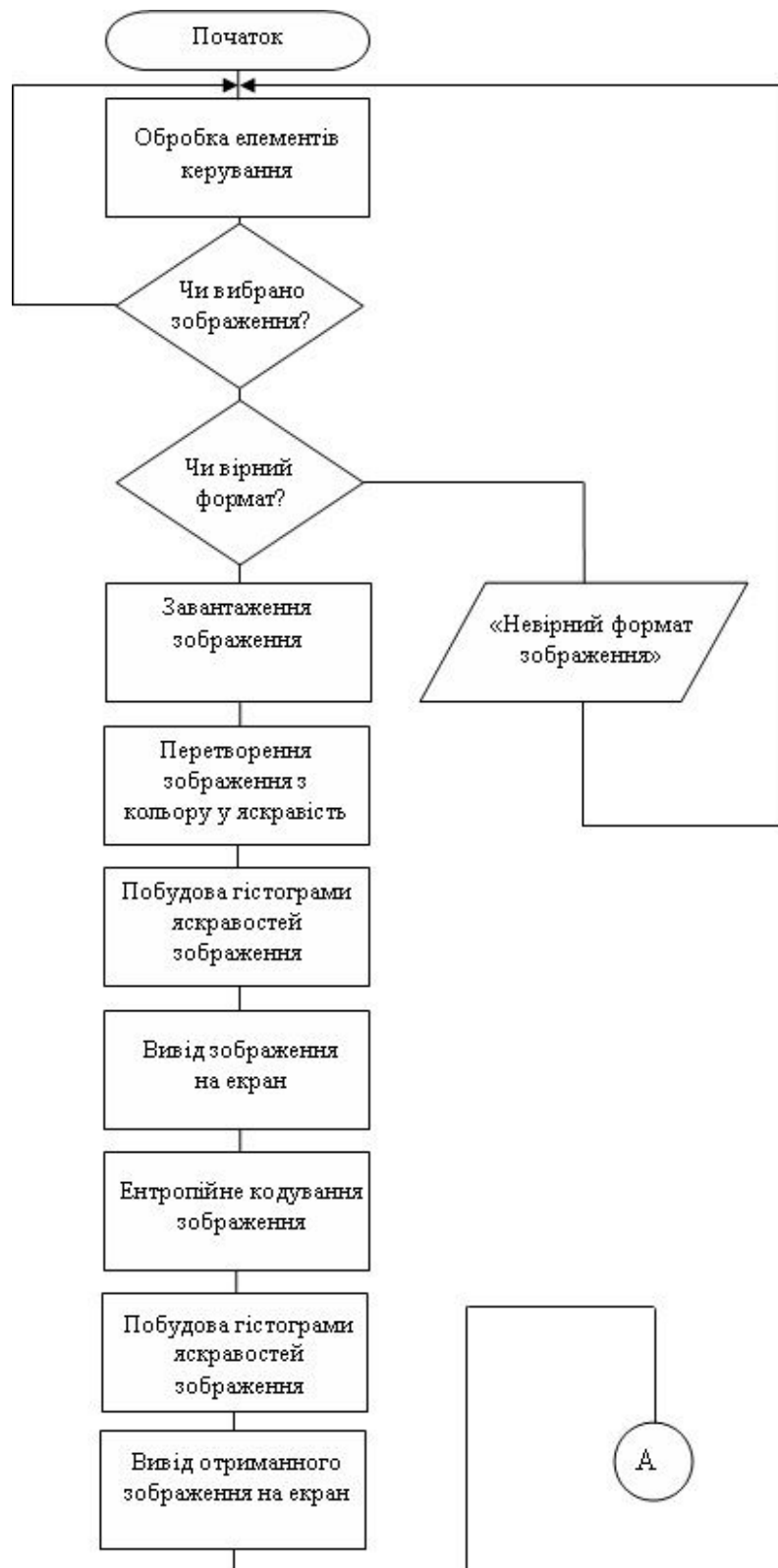


Рисунок 4.2 – Схема алгоритму основної програми

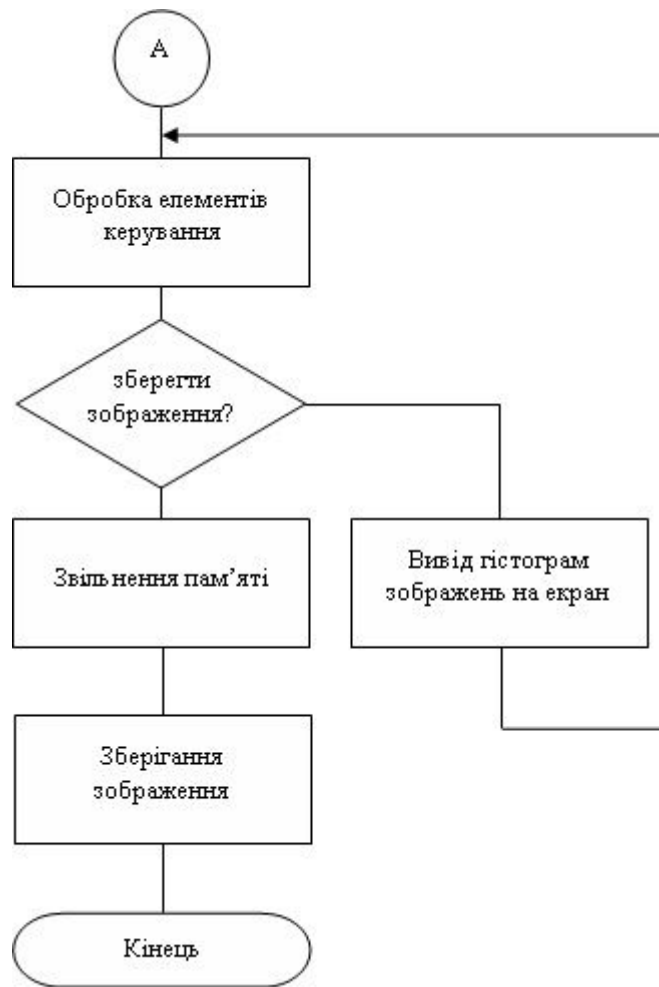


Рисунок 4.2 – Продовження схеми алгоритму основної програми

У тому числі, для завантаження нашого зображення ми можемо скористуватися стандартним класом `TBitmap`, який дозволяє одразу преобразувати `bmp`-зображення у матричний вид. При цьому всі параметри зображення зберігаються у відповідних масивах та змінних класу, які ми потім зможемо використовувати для проведення квантування та кодування зображення. Натомість, ми запропонуємо алгоритм, який одразу буде завантажувати зображення із преобразуванням його у 256 градацій сірого відтінку.

Відповідно до особливостей представлення чорно-білих зображень (256 градацій сірого кольору), кожен піксель такого зображення має відповідати рівню яскравості кольору початкового зображення. Тобто для того, щоб ми змогли перетворити кольорове зображення, нам потрібно порахувати яскравість

кожного пікселю зображення, та задати колір відповідної RGB-складної як рівень яскравості попереднього пікселю (нагадаємо, що у чорно-білому зображенні усі компоненти RGB мають однакове значення).

Рівень яскравості кожного пікселя розраховується по формулі

$$L_{ij} = (30 * R_{ij} + 59 * G_{ij} + 11 * B_{ij}) / 100 \quad (4.1)$$

Таким чином, кожному пікселю початкового зображення потрібно змінити значення на отримане за формулою (3.1), що приведе до зміни зображення як показано на рисунку 3.3.



Рисунок 3.3 – Зміна кольорового зображення на чорно-біле.

Вектор яскравості (гістограма):

$$Y = \{y_i\}^n \in N_+^n \quad (4.2)$$

де n – кількість градацій яскравості в зображенні.

Компонента вектора яскравості – це кількість пікселів з даним значенням яскравості:

$$y_k = \sum_{i=1}^h \sum_{j=1}^w (x_{ij} = \{L_{ij} = k; 1; 0\}), \quad k=1..n \quad (4.3)$$

На основі поданих формул, побудуємо гістограму яскравостей зображення, яка заповнюється наступним чином.

Складається масив з 256 елементів (кожен елемент якого відповідає одному рівню яскравості обраного зображення), кожен елемент якого на початковому етапі дорівнює нулю.

Масив заповнюється наступним чином: визначається яскравість піксель зображення, який має координати в матриці $[1,1]$, значення яскравості - номер елемента даного масиву, додати до цього елемента масиву 1, перейти до наступного пікселя зображення.

Таким чином, ми отримуємо гістограму яскравості нашого зображення.

Для Розробка функції обчислення порогового значення відрізка масиву яскравостей зображення, введемо умовні позначення, для спрощення орієнтування у функції GetMedian:

Вхідні дані: GisYar – масив гістограми яскравостей зображення, a1 - початковий елемент відрізка, на якому буде виконуватися квантування яскравостей, a2 – кінцевий елемент відрізка, Коef - коефіцієнт ентропійності, отриманий на основі методу, викладеного у другому розділі/

Вихідні дані: Elem – номер елементу масиву, який буде пороговим значенням (тобто відрізок початкового масив GisYar $[a1; a2]$ буде поділено елементом Elem на 2 кластери).

Тимчасові дані: MasTemp - масив довжиною $(a2-a1)$, кожним елементом якого буде сума попередніх його елементів, Sum - проміжна сума.

Для розробки рекурсивної процедури розділення масиву гістограми яскравостей та складання масиву відповідностей елементів палітри введемо умовні позначення процедури DelenieNa2.

Вхідні дані: a1 - початковий елемент відрізка масиву, на якому буде виконуватися процедура, a2 - кінцевий елемент відрізка, level - рівень глибини рекурсії, Maxlevel - максимальна глибина рекурсії для обраної кількості кольорів.

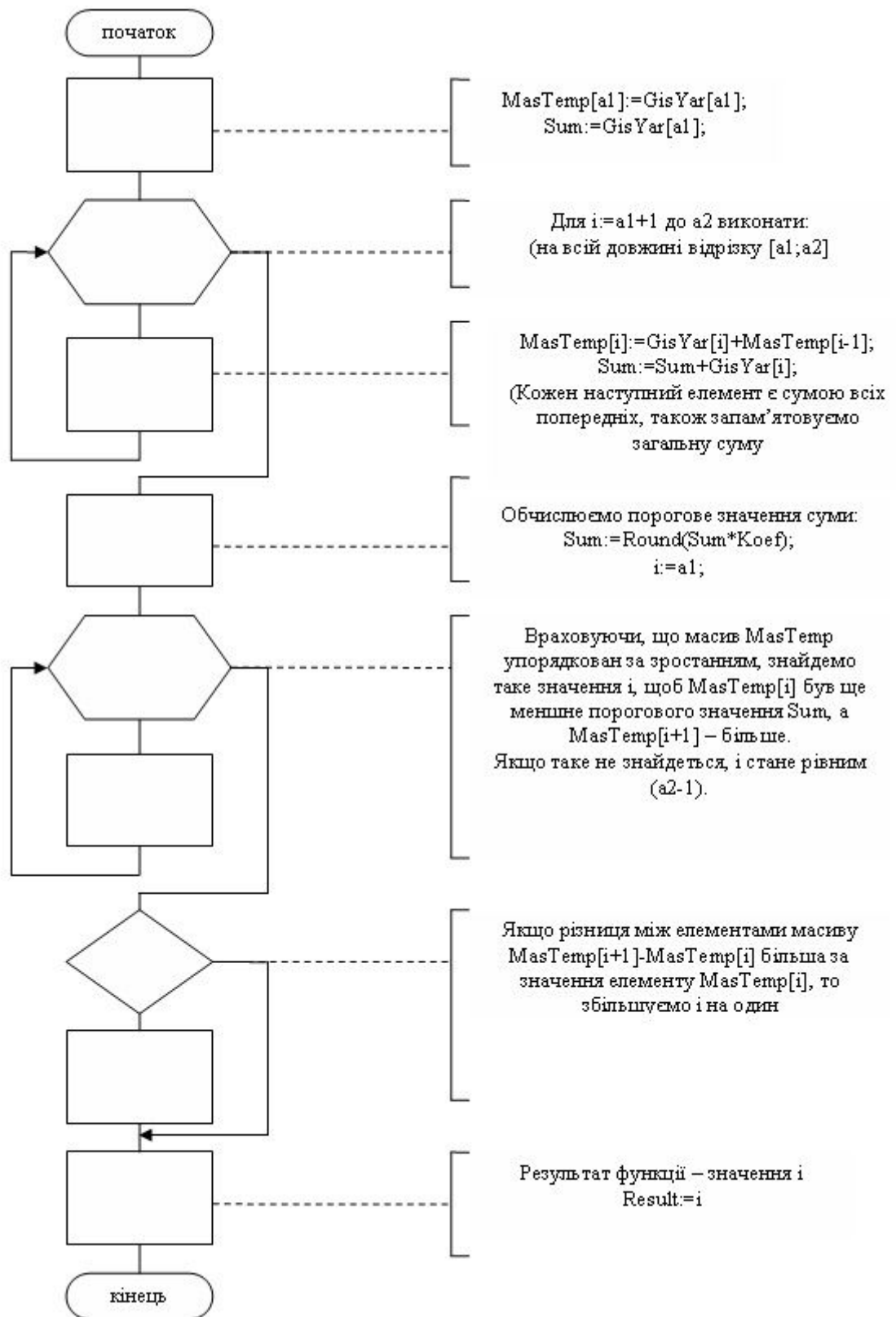


Рисунок 4.4 – Алгоритм функції обчислення порогового значення.

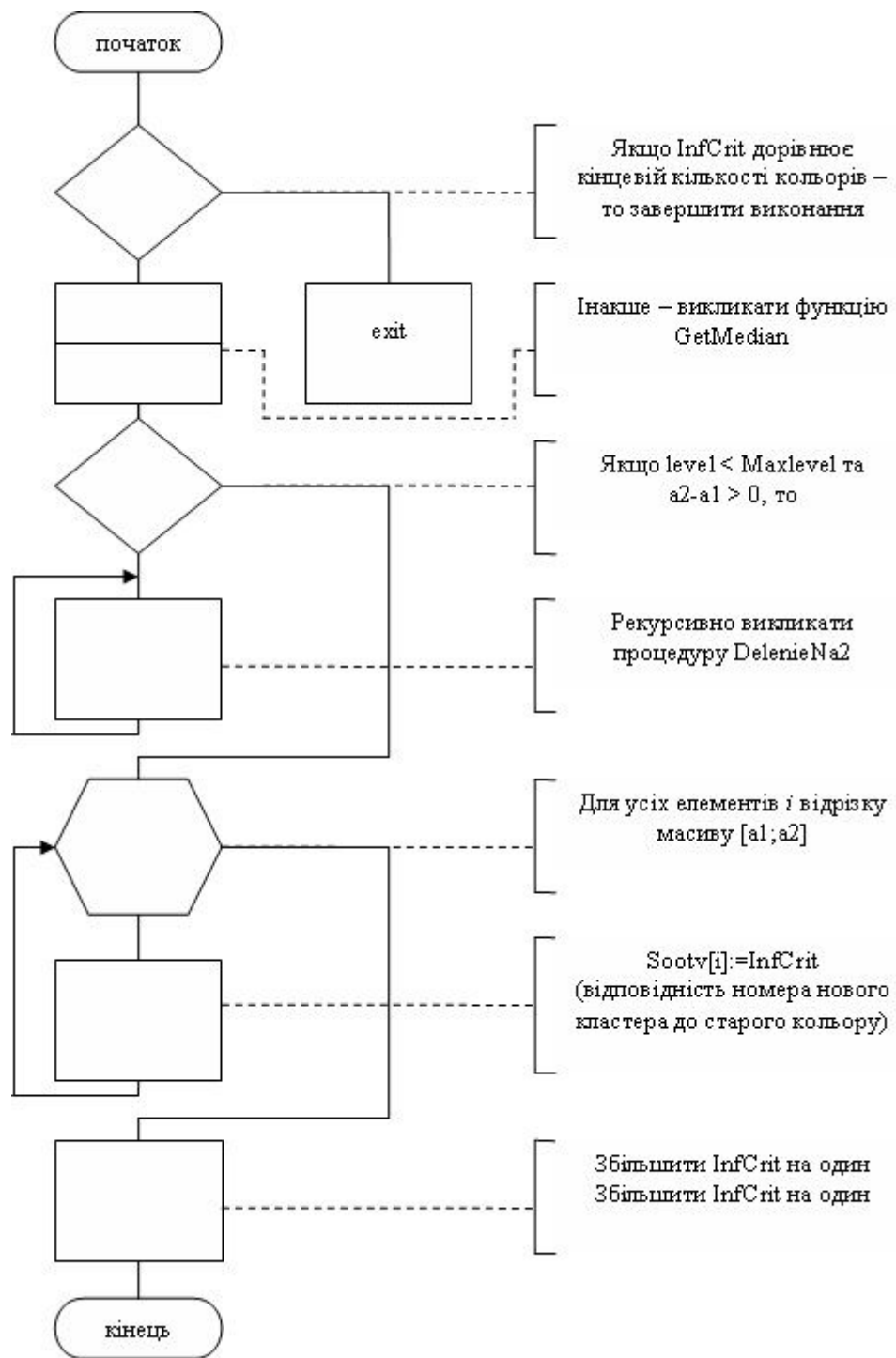


Рисунок 4.5 – Алгоритм рекурсивної процедури складення масиву відповідностей.

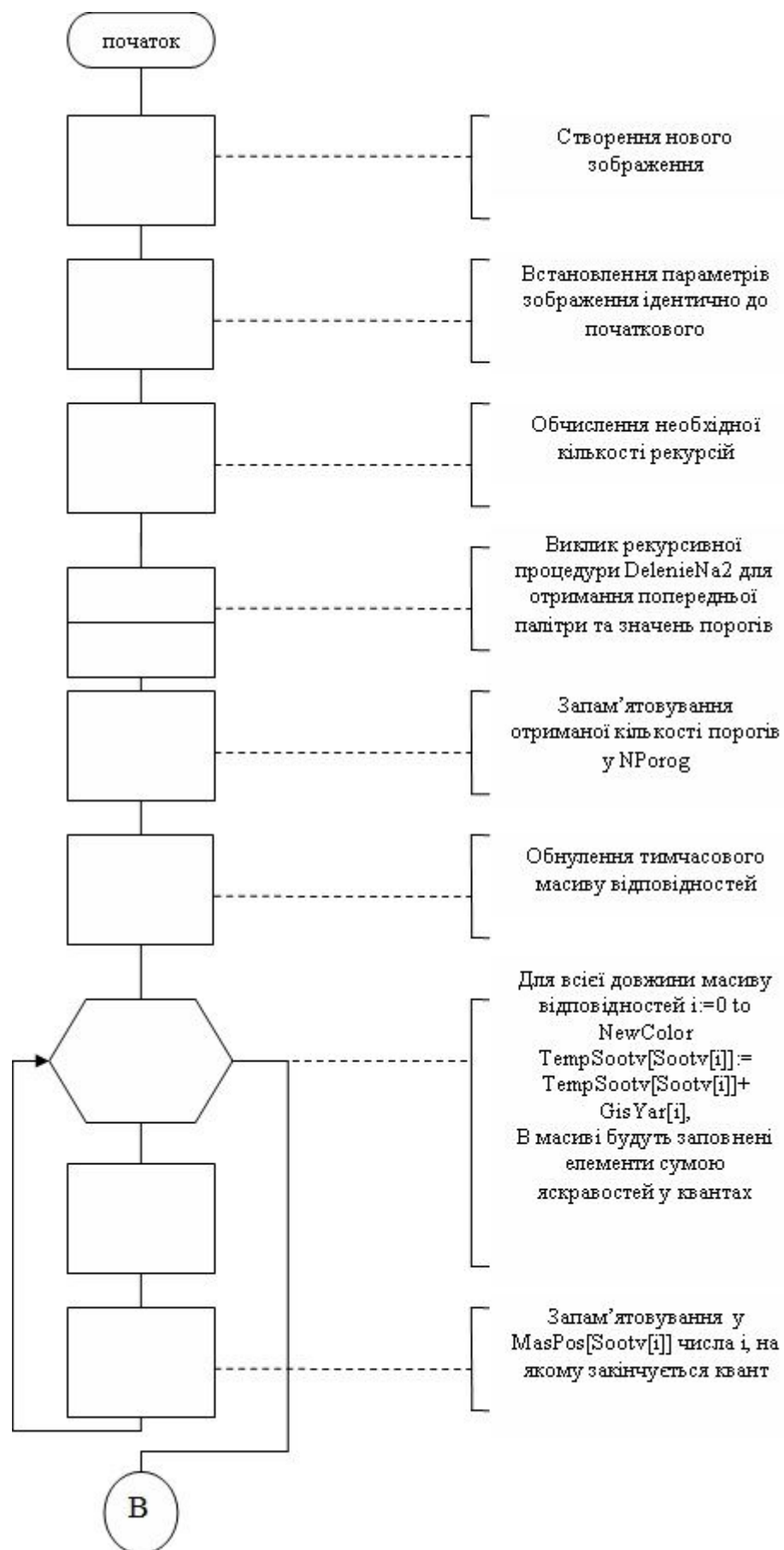


Рисунок 4.6 – Алгоритм функції ентропійного кодування.

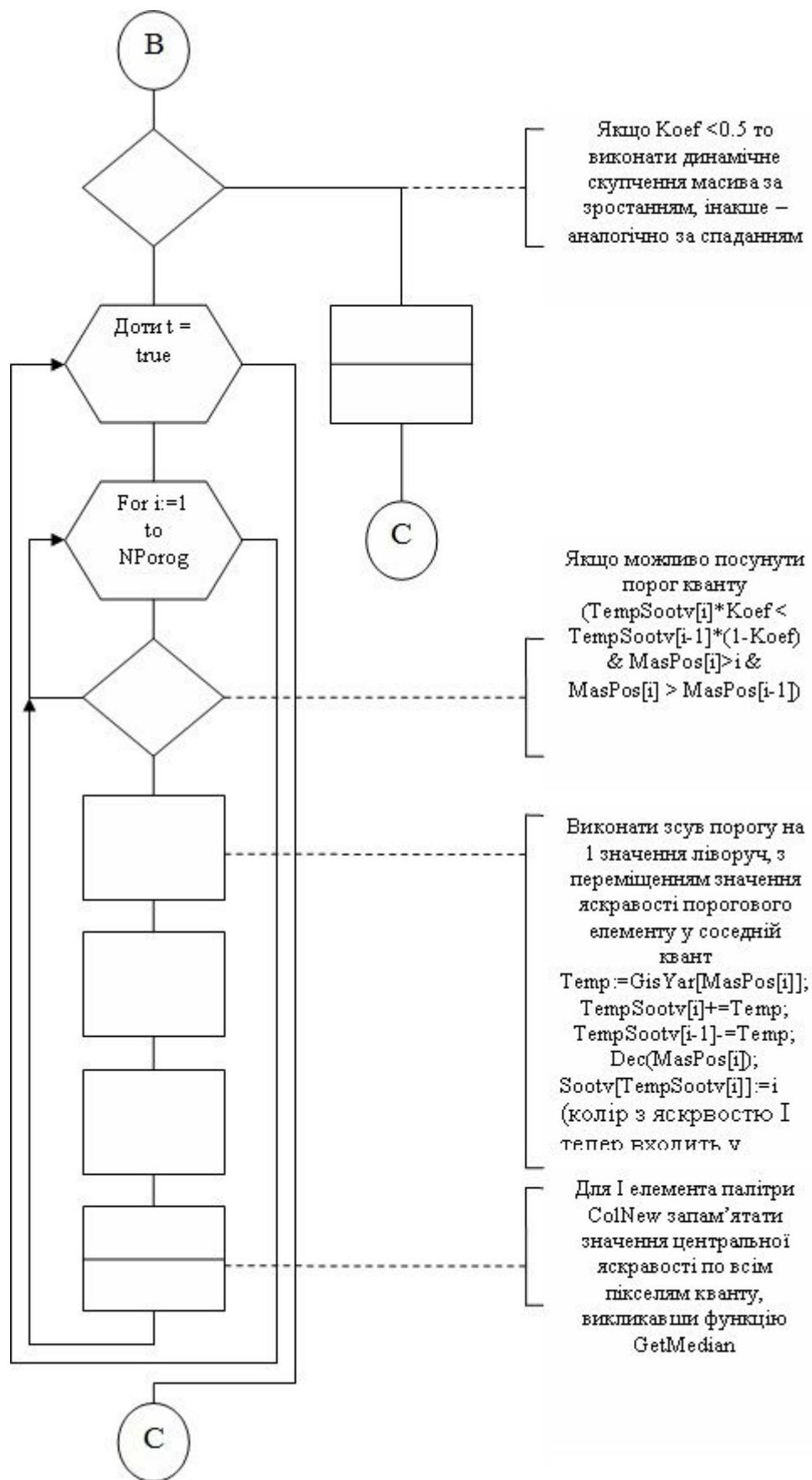


Рисунок 4.6 – продовження алгоритму функції ентропійного кодування.

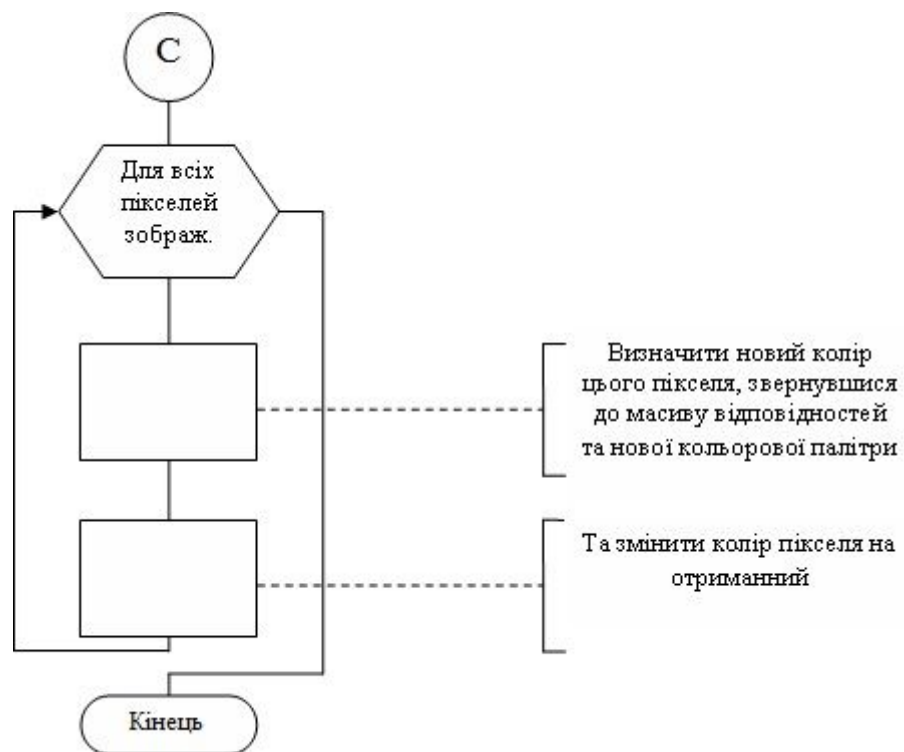


Рисунок 4.6 – Продовження алгоритму функції ентропійного кодування.

Тимчасові дані: *Med* - пороговий елементу масиву, отриманий в результаті виконання функції *GetMedian*, *InfCrit* - номер кольору нової палітри для зображення, *Sootv* - масив відповідностей (256 елементів, у який для кожного елементу буде заноситися значення номеру кванту, для якого потім ми отримаємо новий колір).

Для розробка функції ентропійного кодування зображення введемо умовні позначення функції *DecreaseColor*, результатом якої стане стисле зображення:

Вхідні дані: *bmpPicture* - початкове зображення, *OldColor* - кількість кольорів у початковому зображенні (дорівнює 255), *NewColor* - кількість кольорів у стисломому зображенні, *GisYar* - масив гістограми яскравостей зображення, *Koef* - коефіцієнт ентропійності.

Тимчасові дані: *Maxlevel* – максимальна глибина рекурсії для обраною кількості кольорів, *InfCrit* – номер кольору нової палітри для зображення, *ColNew* - масив довжиною *NewColor*, з новими значеннями яскравостей для

отримання стислого зображення, TempMas - тимчасовий масив яскравостей, Sootv – масив відповідностей, TempSootv - тимчасовий масив відповідностей. MasPos - масив позиції порогових значень.

4.2 Опис використаних компонентів

Компонент TForm

Хоча компонент TForm відсутній у палітрі компонентів, але все-таки має властиві для нього властивості, події.

Отже, компонент TForm це вікно у віконному додатку. Воно може бути мінімум одне, максимальне обмеження на кількість не накладається. Перше створене вікно в додатку автоматично стає головним. При закритті головного вікна додаток завершує свою роботу й займана їм пам'ять звільняється.

Якщо необхідно, щоб головне вікно форми автоматично не з'являлося на екрані, то в події OnCreate для цього вікна, або в програмі DPR до команди створення головного вікна необхідно вказати наступний рядок:

```
// Код DelphiApplication. ShowMainForm: =false;
```

У цій команді вказується, що в цьому додатку заборонене відображення головної форми (вікна). У цьому випадку програмістові потрібно самостійно викликати команду

```
// Код DelphiForm1. Show;
```

для відображення цього вікна. Тут Form1 - головне вікно додатка.

Даний метод може виявитися корисним для висновку запиту пароля на запуск програми. Саме це діалогове вікно потрібно показати користувачеві перед показом головного вікна, але не роблячи його головним вікном, тобто

при закритті вікна уведення пароля, додаток не завершує свою роботу. І при уведенні невірною пароля можна без проблем завершити роботу програми командою

```
// Код DelphiForm1. Close;
```

Або дати команду завершення роботи додатка.

Як уже говорилося в попередніх уроках, кожний компонент у програмі, як і сама форма, має унікальне ім'я, зазначене у властивості Name. Заголовок вікна втримується у властивості Caption. По заголовку вікна користувач довідається про функціональне призначення програми або поточного вікна.

Крім заголовка у верхній частині вікна перебуває іконка й кнопки керування станом. За замовчуванням іконка така ж сама, як і іконка в проекті. Щоб перемінити іконку в поточному вікні необхідно в інспекторі об'єктів вибрати властивість Icon у якому вибрати відповідний файл-малюнок. Файл повинен бути з розширенням ICO.

Для того, щоб перемінити іконку в проекті, потрібно увійти в меню "Project", далі "Options... ", на вкладці Application ви бачите поточний малюнок файлу проекту. Перемінити його можна кнопкою "Load Icon... ". Іконка проекту зберігається у файлі ресурсів з розширенням RES.

Властивість *BorderStyle*

bsDialog - У вікна немає іконки. Відображена тільки кнопка керування "Закрити". Розмір вікна постійний. Такий тип вікон застосовується найчастіше в діалогових вікнах, наприклад вікно запиту на збереження проекту, якщо ви намагаєтеся вийти з delphi не зробивши збереження.

bsNone – У вікна немає іконки, кнопок керування, заголовка. Розмір вікна постійний. Закрити таке вікно можна тільки програмно або за допомогою комбінації клавіш Alt+F4. Цей тип вікон застосовується в заставці при запуску програми. На вікні розташований компонент TImage, що містить малюнок.

`bsSingle` – У вікні присутнє іконка, є заголовок. Кнопки керування згорнути, розгорнути (відновити), закрити. Розмір вікна постійний.

`bsSizeable` – Тип вікна за замовчуванням. Має всі елементи, зазначені в попереднім значенні властивості плюс зміна розмірів вікна.

`bsSizeToolWin` – Цей тип вікна встановлений у вікні інспектора об'єктів. Таке вікно не має іконки, є заголовок, кнопка керування "закрити". Можна змінювати розмір вікна.

`bsToolWindow` – Аналогічний попередньому значенню, за винятком того, що не можна змінювати розміри вікна.

Всі вищеописані типи властивості `BorderStyle` встановлюються тільки для працюючого додатка, тобто ми бачимо зміни в оформленні вікна тільки після запуску програми.

Властивість `BorderIcons`

Ця властивість є вкладеним, обто якщо ви в інспекторі об'єктів натиснете на знак +, то відкриється ще трохи підвластивостей. Властивість `BorderIcons` має сенс встановлювати тільки у відповідних значеннях властивості `BorderStyle`.

`biSystemMenu` – указує, відобразити чи ні іконку й кнопки керування вікном.

`biMinimize` – відобразити чи ні кнопку керування "згорнути".

`biMaximize` – відобразити чи ні кнопку "розгорнути" ("відновити").

`biHelp` – відобразити чи ні кнопку "допомога".

Можливі два значення для цих властивостей: `true` (істинно) і `false` (ложно).

Властивість `Position`

`poDefault` – Windows саме визначає розмір і положення вікна.

`poDefaultPosOnly` – довільне положення вікна.

`poDefaultSizeOnly` – тільки довільний розмір вікна.

`poDesigned` – такий же розмір і положення, кокою був при розробці. Значення за замовчуванням. Вам належить знати, що не на всіх комп'ютерах

установлене такий же екранний дозвіл, яке є у вас, і значить вікна в програмі будуть перебувати в зовсім іншій місці.

`poDesktopCenter` – вікно перебуває в центрі екрана.

`poScreenCenter` – вікно перебуває в центрі екрана. Працює із багатодисплейним режимом відображення.

Властивість *WindowState*

`wsMaximized` – Вікно спочатку буде розгорнуто у весь екран.

`wsMinimized` – Вікно спочатку буде згорнуто в значок.

`wsNormal` – Вікно має ті ж розміри, що й під час розробки. Значення за замовчуванням.

Компонент **TEdit**

Цей компонент не містить власного коду, у ньому тільки опубліковані властивості його предка `TCustomEdit`. Він являє собою редагує строку, що (далі – просто редактор).

Стиль обрамлення цього компонента

(Pb) property `BorderStyle: TBorderStyle;`

за замовчуванням дорівнює `bsSingle`. Якщо властивість

(Pb) property `AutoSize: Boolean;`

дорівнює `True`, компонент змінює свою висоту залежно від розміру шрифту (властивість `Font`). Для того щоб зміни мали місце, повинен ще бути встановлений стиль обрамлення `bsSingle`.

Текст, що вводиться у редактори, може автоматично піддаватися деяким перетворенням.

Перетворення, що задає властивістю *CharCase*, дозволяє автоматично перетворювати всі символи тільки у верхній або тільки в нижній регістр:

(Pb) property `CharCase: TEditCharCase;`

`TEditCharCase = (ecNormal, ecUpperCase, ecLowerCase);`

За замовчуванням установлений стиль `ecNormal` і перетворення не відбувається. Аналогічно, властивість

(Pb) property `OEMConvert: Boolean;`

визначає необхідність автоматичного перетворення символів, що вводять, з кодування OEM в ANSI і назад. Така властивість часто буває потрібно при обробці тексту в кодуванні MS-DOS.

Дві наступні властивості описують поведження виділеного тексту при переході фокуса. Перше з них

(Pb) property *AutoSelect*: Boolean;

описує реакцію елемента, що редагує, при одержанні фокуса. Якщо воно встановлено в True (за замовчуванням це так), то при одержанні фокуса уведення весь текст у ньому виділяється незалежно від стану властивості *SelText*. Якщо *AutoSelect* встановлено в False, то при одержанні фокуса виділяється лише те, що було виділено до його втрати.

Після втрати фокуса редактором виділених у ньому текст звичайно губить колірне виділення. Щоб воно залишалось, встановіть в False друга властивість:

(Pb) property *HideSelection*: Boolean;

На довжину тексту може бути накладене обмеження.

Властивість (Pb) property *MaxLength*: Integer;

визначає максимальну довжину тексту редактори в символах. Якщо значення цієї властивості дорівнює 0, то обмежень на довжину тексту немає.

Властивість *PasswordChar* призначена для уведення пароля з використанням редактора:

(Pb) property *PasswordChar*: Char;

Його значення – це символ, використовуваний для відображення замість будь-яких символів, що вводять.

Можна заборонити можливість редагування тексту. Якщо значення властивості:

(Pb) property *ReadOnly*: Boolean;

дорівнює True, текст змінити не можна.

Ви можете відстежити зміни тексту в редакторі, обробляючи вступників події:

(Pb) property OnChange: TNotifyEvent;

Компонент **TButton**

TObject –> TPersistent –> TComponent –> TControl –> TWinControl –> –
*TButtonControl –> Tbutton

Звичайна кнопка Windows. У цьому компоненті опубліковані тільки властивості Default, Cancel, Caption, ModalResult і OnClick. Подвійний щиглик на кнопці не передбачений.

Компонент **TImage**

Цей компонент служить надбудовою над класом TPicture і замикає всю ієрархію графічних об'єктів VCL. Він призначений для показу на формі зображення: бітової карти (TBitmap), метафайла (TMetafile), значка (TIcon).

Властивість (Pb) property Picture: TPicture;

служить контейнером для графічного об'єкта одного з перерахованих класів.

Як канва використовується канва об'єкта Picture. Graphic – тільки якщо поле Graphic посилається на об'єкт класу TBitmap:

(Ro) property Canvas: TCanvas;

Якщо це не так, то спроба звертання до властивості викличе виняткову ситуацію EInvalidOperation, тому що малювати на метафайлі або значку не можна. Будьте уважні із цією властивістю.

Наступні три властивості визначають, як саме TPicture розташовується в робочій області компонента:

(Pb) property *AutoSize*: Boolean;

означає, що розміри компонента настроюються по розмірах графічного об'єкта, що втримується в ньому. Установлювати його в True потрібно перед завантаженням зображення з файлу або буфера обміну;

(Pb) property *Stretch*: Boolean;

якщо ця властивість установлена в True, те зображення "натягається" на робочу область, при необхідності зменшуючи або збільшуючи свої розміри. Якщо воно встановлено в False, то відіграє роль наступна властивість;

(Pb) property *Center*: Boolean;

якщо ця властивість установлена в True, зображення центрується в межах робочої області. У протилежному випадку – розташовується в її верхньому лівому куті.

Компонент **TLabel**

являє собою статичний текст. За допомогою цього компонента на робочій поверхні форми можна відобразити інформацію, зробити пояснення й показати назви інших компонентів. Але він має й іншу важливу функцію – якщо в складі тексту TLabel є символи-акселератори, інформація про їхнє натискання може передаватися від TLabel іншому елементу керування.

Нижче наведені властивості компонента:

(Pb) property *Caption*: TCaption;

TCaption = string [255] ;

Містить рядок з виведеним текстом.

(Pb) property *Alignment*: TAlignment;

Установлює правило вирівнювання тексту – по правому, лівому краї або по центрі клієнтської області.

(Pb) property *AutoSize*: Boolean;

У випадку True відбувається автоматичне приведення розмірів компонента до розмірів тексту й висоті шрифту.

(Pb) property *ShowAccelChar*: Boolean;

Показує, чи відображається в TLabel акселератор (символ, перед яким коштує знак амперсанда – '&'). Якщо воно дорівнює False, амперсанд ніяк не інтерпретується. У випадку True акселератор виділяється в рядку підкресленням і натискання клавіш <A11>+<символ> приводить до передачі фокуса вікну FocusControl (тому що TLabel сам не є віконним компонентом).

(Pb) property *FocusControl*: TWinControl;

Визначає віконний компонент, якому посилають повідомлення про уведення акселератора.

(Pb) property *Transparent*: Boolean;

Прозорість компонента. Якщо властивість установлена в True, то при перемальовуванні елемента керування не відбувається зафарбовування клієнтської області. У протилежному випадку – відбувається зафарбовування кистю bsSolid і кольорами Color.

(Pb) property WordWrap: Boolean;

Визначає можливість розриву слів у випадку, якщо довжина виведеного тексту перевищує ширину компонента.

4.3 Тестування програми

У цьому розділі дипломного проекту ми протестуємо нашу програму на стиснення кольорового зображення "схід сонця". Для тестування було обрані прямий та зворотній методи кодування, з глибиною кольору отриманого зображення 4 та 16 кольорів.

Початковий розмір файлу у форматі. bmp – 1.44 Мб. Розглянемо якість зображення в процесі кодування (рис 4.7 – 4.10)

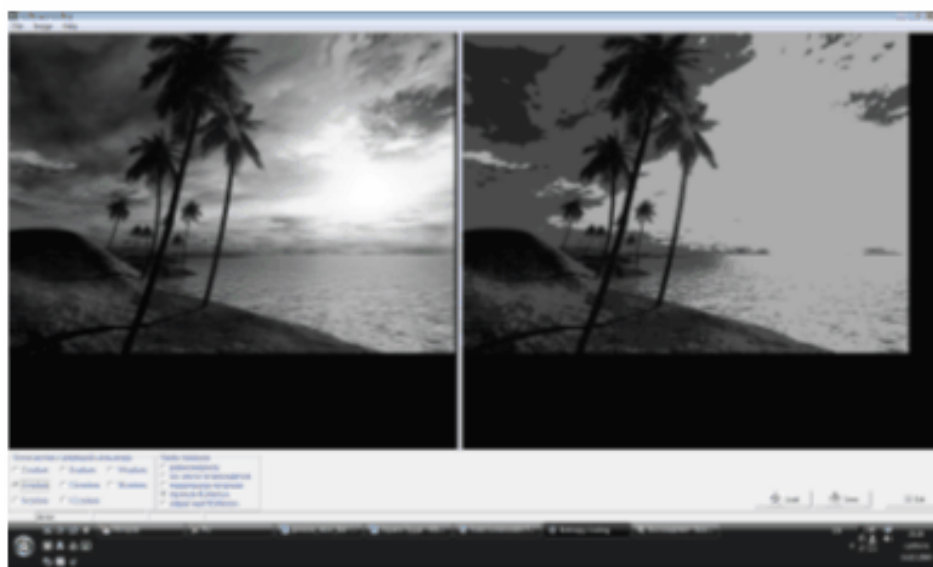


Рисунок 4.7 – Метод зворотного кодування з глибиною кольору 4

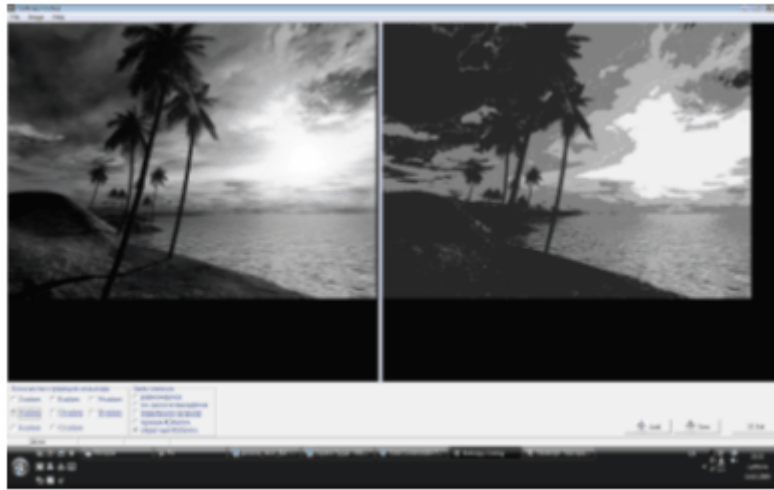


Рисунок 4.8 – Метод прямого кодирования з глибиною кольору 4

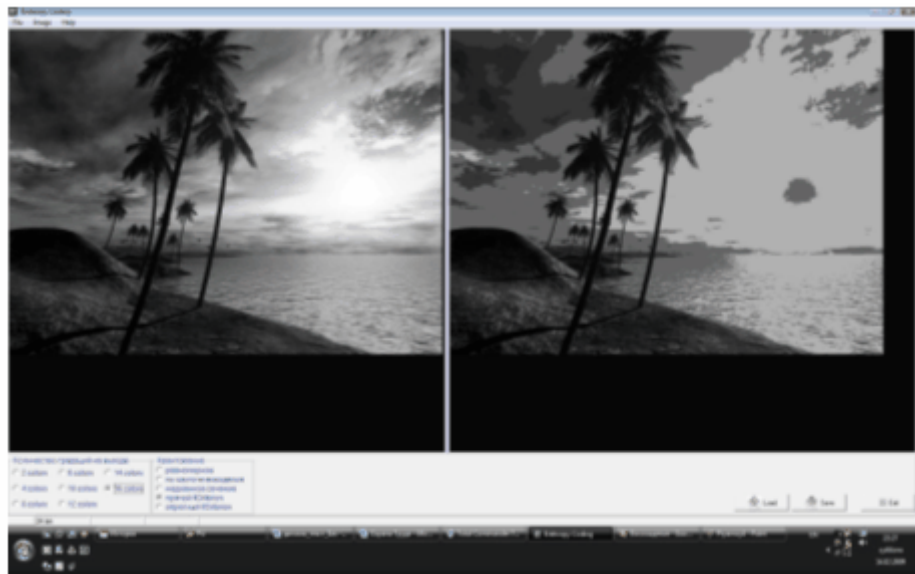


Рисунок 4.9 – Метод зворотнього кодвання з глибиною кольору 16

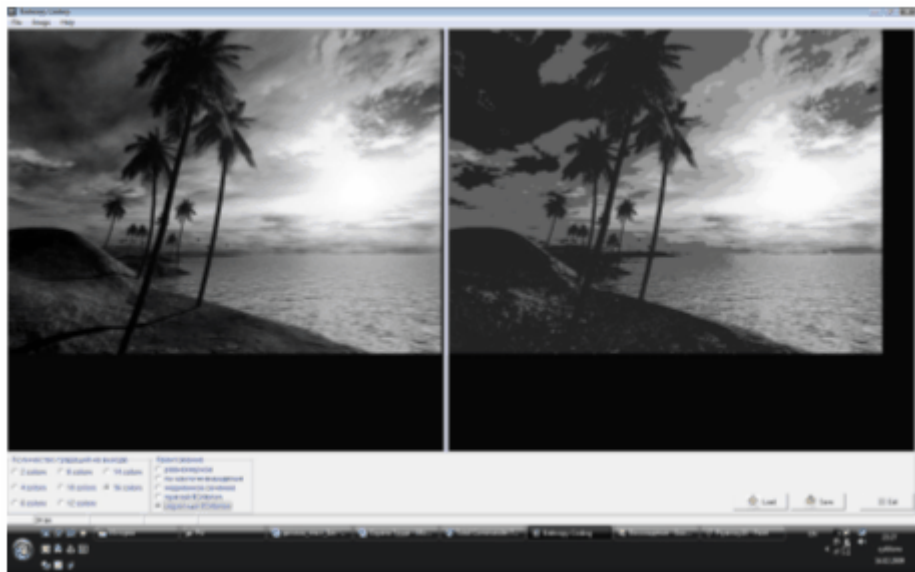


Рисунок 4.10 – Метод прямого кодвання з глибиною кольору 16

Як ми бачимо, "прямий" метод показав гарну чутливість до малих значень пікселей, "зворотній" – виділив всі яскраві ділянки вихідного зображення. При цьому всі 4 отриманих зображення мають гарну деталізацію, і чоловіче око майже не помічає різниці між зображеннями, особливо при глибині яскравостей кольорів зображення встановленого у 16.

Таким чином, при роботі з зображеннями, де розподіл світлих та темних кольорів приблизно однаковий, ми можемо помітити, що якість отриманого кодованого зображення залишається задовільною. Особливо важливим цей аспект може стати у нагоді для друкованих видавництв, або для архівів зберігання баз даних зображень (наприклад, міліція та служба безпеки).

В результаті зберігання отриманих файлів зображень, їх розмір зменшився до 240 Кб, що становить 1/6 розміру початкового файлу. При цьому, ці зображення можуть бути додатково закодовані у інший формат, оснований на довжині однакових кодів – тому що в наших отриманих зображеннях є великі одно яскраві частини, його розмір може зменшитися до 40–50 Кб, що майже у 30–35 разів менше за початкове зображення. Звісно, це дуже економить місце на жорсткому диску.

Також, ми можемо помітити, що програма ефективно переробила багатокольорове зображення у чорно-біле, повністю зберевши інформативність зображення. Тому наша програма також може використовуватися для зберігання чорно-білих зображень, особливо це може стати у нагоді фотографам, для обробки фотографій у стилі "ретро".

Результати тестування програми показали, що наша програма повністю відповідає технічному завданню, і виконує всі необхідні дії для кодування зображення на основі аналізу ентропійності.

ВИСНОВКИ

У ході виконання роботи були докладно розглянуті різні аспекти імовірнісного підходу в теорії ефективного кодування зображень. Викладено методи побудови імовірнісних моделей, методи генерації коду змінної довжини на основі імовірнісних розподілів, методи одержання імовірнісних оцінок., а також розглянуті проблеми кодування відеоінформації.

В результаті запропоновано алгоритм кодування відеоінформації із застосуванням квантування яскравостей зображення на основі ентропійних характеристик з метою зменшити його розміру із втратою інформативності.

В роботі вдосконалено алгоритм за рахунок визначення ентропійних властивостей зображень на основі використання оцінки ентропії, що базується на врахуванні кореляційних властивостей джерела інформації. Це дозволило в 2 рази збільшити ступінь стиснення інформації, хоча швидкість кодування збільшилася на 1,2 – 1,5 рази через збільшення обчислювальної складності.

Шляхом порівняння експериментальних досліджень із результатами теоретичного моделювання визначено кількісний оптимальний коефіцієнт, за яким повинні співвідноситися рівні яскравостей зображення для отримання найкращого результату.

Також розроблено програмне забезпечення, яке здійснює стиснення зображення за рахунок зменшити кількість кольорів у зображенні, але зберігається його інформативність. Цей алгоритм можна застосовувати як швидкий і зручний спосіб стиснення зображення.

В роботі приведено детальний опис реалізованого алгоритму, загальний опис інтерфейсу програмного забезпечення, пояснення окремих модулів програми. Також наведено результати роботи програмного забезпечення, що підтверджує працездатність програмного забезпечення та ефективність алгоритму в цілому.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Айвазян С.А., Мхитарян В.С. Прикладная статистика и основы эконометрики. – М.: Юнити. 1998. – 1022 с.
2. Дидэ Э. и др. Методы анализа данных / под ред Айвазяна С.А. и Бухштабера В.М. – М.: Финансы и статистика, 1985. – 357с.
3. Кричевский Р.Е. Сжатие и поиск информации. – М.: Радио и связь, 1989.
4. Куренков Н.И. Ананьев С.Н. Энтропийный подход к решению задач классификации многомерных данных. // Информационные технологии. 2006. № 8. С.50–55.
5. Левенштейн В.И. Об избыточности и замедлении делимого кодирования натуральных чисел // Проблемы кибернетики. – М., 1968. – Вып. 20. – С.173 – 179.
6. Рябко Б.Я., Фионов А.Н. Эффективный метод адаптивного арифметического кодирования для источников с большими алфавитами // Проблемы передачи информации. – 1999. – Т.35, Вып.4. – С.95 – 108.
7. Семенюк В.В. Применение вероятностного моделирования в методах экономного кодирования видеoinформации // Труды XI Всероссийской научно-методической конференции Телематика'2004. – Санкт-Петербург, Россия, 7–10 июня, 2004. – С.186 – 187.
8. Сакоян С.А. Об оптимальных разбиениях на градации в задачах классификации // Прикладная статистика – М.: Наука, 1983. – С.179–188.
9. Семенюк В.В. Экономное кодирование дискретной информации.
10. СПб: СПб ГИТМО (ТУ), 2001. – 115 с, – ISBN 5–7577–0076–9.
11. Хаффмен Д.А. Метод построения кодов с минимальной избыточностью: Пер. с англ. // Кибернетический сборник. – М.: ИЛ, 1961. – Вып.3. – С.79 – 87.
12. Шеннон К. Математическая теория связи: Пер. с англ. // Работы по теории информации и кибернетике. – М.: ИЛ, 1963. – С.243 – 332.

13. Howard P. G., Vitter J. S. Practical Implementations of Arithmetic Coding // Storer A. Image and Text Compression. – Kluwer Academic Publishers, Massachusetts, USA, 1992. – P.85 – 112
14. INTUIT.ru. Алгоритмы квантования для полутоновых и цветных изображений. - 2004. -С.1-4.
15. INTUIT.ru. Алгоритмические основы растровой графики. - 2004. -С.1-12.
16. Kraft L. A Device for Quantizing, Grouping and Coding Amplitude Modulated Pulses: MS Thesis. - Dept. of Electrical Engineering, MIT, Cambridge, Massachusetts, USA, 1949.
17. Rui Xu, D., Wunsch II. Survey of clustering algorithms // IEEE Transactions on Neural Networks. - 2005. - v.16, № 3. - p.645.
18. Хартлі Р. Передача інформації / Теорія інформації і її застосування / Під ред. А. А. Харкевича – М.: Фізматгиз, 1959. – С. 5–36.
19. Колмогоров А. Н. Теорія ймовірності і математична статистика. [Сб.] / Отв. ред. Ю. В. Прохоров; [АН СРСР, Отд-ние математики]. – М.: Наука, 1986. – 534 с.
20. Шенон К. Роботи по теорії інформації і кібернетикі. – М.: Іноземна література, 1963, – 830с.
21. Longo G. Quantitative-qualitative measure of information. Internat. Centre of mechan. Sciences (Sommerkurs in Udien). Springer-Verlag, 1972.
22. Shults G. Informationstheorie mit Bewertung. Wiss/ Zeitschrift d. Humboldt Univer. Berlin XX – 1971. S. 175-183.
23. Оливер Б. Ефективне кодування / Теорія інформації і її застосування / Під ред. А. А. Харкевича. – М.: Фізматгиз, 1959. – С. 159-190.
24. Миддлтон Д. Введення в статистичну теорію зв'язі. – М: Радіо, 1961. – 768 с.
25. Таллер В. Теоретичні обмеження швидкості передачі інформації. Теорія інформації і її застосування / Під ред. А. А. Харкевича – М.: Фізматгиз, 1959. – С. 58–81.

26. Боюн В. П., Козлов Л. Г. Принципы построения проблемно-ориентированных процессоров для решения научно-технических задач. – К.: Знание УССР, 1978. – 22 с.

27. Боюн В. П. Теоретико-информационные основы преобразования и обработки информации в системах реального времени // Проектирование и применение средств микропроцессорной техники: Сб. научн. Трудов.– Киев: ИК им. Глушкова АН УССР, 1986. – С. 9–17.4

28. Лапа В. Г. Математические основы кибернетики. – К.: И-во “Вища школа”, 1974. – 452 с.

29. Макс Ж. Методы и техника обработки сигналов при физических измерениях: пер. с франц. – М.: Мир, 1983. – Т. 1. – 311 с.; Т. 2. – 256 с.

30. Краус М., Кучбах Э., Вошни О.-Г. Сбор данных в управляющих вычислительных системах: Пер. с нем. – М.: Мир, 1987. – 294 с.

31. Голдман С. Теория информации. – М.: Изд-во иностранной литературы, 1975. – 382 с.

32. Кузьмин И. В., Кедрус В. А. Основы теории информации и кодирования. – К.: «Вища школа», 1986, – 238 с.

33. Николайчук Я.М. Розробка теорії і комплексу технічних засобів формування, передачі і обробки цифрових повідомлень в низових обчислювальних мережах автоматизованих систем. Дисертація на здобуття наукового ступеня доктора технічних наук – Івано-Франківськ, ІФІНГ, 1991. – 573 с.

34. Николайчук Я.М. Теорія джерел інформації: монографія / Я.М. Николайчук. – Тернопіль: ТНЕУ, Економічна думка, 2008. – 396 с.