**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ**

**КАФЕДРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

**АВТОМАТИЗОВАНИХ СИСТЕМ**

**МЕТОДИЧНІ ВКАЗІВКИ**

**ДО ВИКОНАННЯ ЛАБОРАТОРНИХ РОБІТ З КУРСУ**

# «Основи інтеграції інформаційних потоків»

для спеціальності 121 - " Інженерія програмного забезпечення "

всіх форм навчання

Затверджено

на засіданні кафедри програмного забезпечення автоматизованих систем

протокол №5 від 14.02.2017 р.

Укладач \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Первунінський С.М., д.т.н., професор

**ЧЕРКАСИ – 2017**

**Вступ**

# Дані «Методичні вказівки» призначені для підтримки лабораторних робіт з курсу «Основи інтеграції інформаційних потоків» для студентів спеціальностей 121 - " Інженерія програмного забезпечення " всіх форм навчання.

Навчальною метою лабораторного заняття є практичне підтвердження окремих теоретичних положень даної навчальної дисціпліни, набуття практичних умінь та навичок роботи з обчислювальною технікою, вимірювальною апаратурою, методикою експериментальних досліджень у конкретній предметній галузі. Перелік тем лабораторних занять визначається робочою навчальною програмою дисципліни.

Лабораторні роботи проводяться з студентами, кількість яких не перевищує половини академічної групи.

Виконання лабораторної роботи поділяється на такі етапи:

1. проведення попереднього контролю підготовленості студентів до виконання конкретної лабораторної роботи;
2. виконання конкретних завдань у відповідності з запропонованою тематикою;
3. оформлення індивідуального звіту;
4. оцінювання результатів роботи студента викладачем.

Методика виконання та оформлення звіту з лабораторної роботи визначається порядком виконання роботи та індивідуальним завданням по роботі. Кожна робота розрахована на одне чотирьох часове її виконання в лабораторії.

**Лабораторні роботи**

**Лабораторна робота № 2**

**Побудова ефективного коду по методу Хаффмена**

Мета роботи

Закріпити теоретичні знання і набути навички по ефективному кодуванню інформації по методиці Хаффмена.

Підготовка до виконання роботи

При підготовці до виконання роботи необхідно:

-вивчити принципи побудови оптимальних нерівномірних (ефективних) кодів (ОНК) по методиці Хаффмена;

-ознайомитись з описом лабораторної роботи;

- підготувати бланк звіту;

Короткі відомості з теорії

Розглянемо методику побудови ОНК, запропоновану Хаффменом [1].

Задача побудови ОНК підстави ***а*** для некорельованих[[1]](#footnote-1) алфавітів m формулюється таким чином: серед всіх можливих кодів підстави *а* без розділових знаків, що володіють властивістю префікса (ніяка коротша кодова комбінація не є початком (префіксом) довшої кодової комбінації), знайти код, для якого мінімально можлива середня тривалість кодової комбінації

, (1)

де  - апріорна вірогідність передачі (появи) - го символу алфавіту m;

 - тривалість - ї кодової комбінації в коді з підставою *а*. Як показав Хаффмен, для того, щоб даний префіксний код забезпечував мінімально можливе значення величини , а отже був би ефективним, необхідно і достатньо виконання наступних трьох умов:

1. Якщо виписати символи у порядку убиваючої вірогідності:

, де , то тривалості відповідних кодових комбінацій повинні задовольняти співвідношенню .

1. В усякому разі, дві останні, але не більше ніж **,** кодові комбінації, рівні по тривалості і відрізняються значенням тільки останнього кодового знаку: де  , де  .
2. Кожна можлива послідовність кодових знаків повинна або сама бути кодовою комбінацією, або мати своїм префіксом використовувану кодову комбінацію.

Відповідно до цих трьох умов методика побудови ОНК полягає в наступному.

Символи алфавіту ***m*** записуються у порядку убиваючої вірогідності в основний стовпець. Останні по вірогідності  символів, де - таке найбільше число з інтервалу , що  - ціле число, об'єднуються в новий символ, вірогідність якого дорівнює сумарній вірогідності символів, його складових. ** символів, що залишились, та одержаний новий символ знову виписуються в перший додатковий стовпець у порядку убиваючої вірогідності. Останні **а** по вірогідності символи першого додаткового стовпця знову об'єднуються в допоміжний символ.  символів, що залишилися та знову одержаний допоміжний символ також виписуються у порядку убиваючої вірогідності і т.д. до тих пір, поки **а** останніх символів додаткового стовпця не дадуть допоміжний символ з вірогідністю, рівній одиниці.

Ключ побудови коду полягає в наступному. Число кодових знаків кодової комбінації *i* - го символу, записуваної справа наліво (від кінця), рівне числу об'єднань, в яких бере участь даний символ на шляху утворення допоміжного символу з одиничною вірогідністю.

Якщо номерам символів в об'єднаннях довільно зіставити **а** знаків [0, 1, 2,..., (а-1)], то значення *N* - го кодового знаку, відлічуваного від кінця кодової комбінації, визначається номером, займаним даним символом (або допоміжним символом, в утворенні якого він брав участь) в його *N* - у об'єднанні від початку побудови.

Приклад побудови деяких ОНК по даній методиці приведений в табл.1. Слід зазначити, що сформульовані Хаффменом три умови ОНК в явному вигляді не містять властивості префікса. Проте властивість префікса автоматично виходить із запропонованої Хаффменом методики побудови кодових комбінацій ОНК і прийнятого напряму запису кодових комбінацій справа наліво.

Необхідно відзначити, що при великому початковому алфавіті **m** безпосереднє застосування методики Хаффмена для складання ОНК приводить до громіздких побудов. У цих випадках іноді зручно скористатися відомим штучним прийомом, запропонованим Мічелем В.С. (Michel W.S.) [2]. Виписаний у порядку убиваючої вірогідності алфавіт **m** розбивається на значно менше число груп символів **m1**<**m.** При цьому бажано, щоб число об'єднуваних в групи символів складало цілий ступінь основи коду  **(****,****,…)**а вірогідність складових групи символів була б по можливості рівна між собою. Вірогідність такої групи є сумарна вірогідність складових її символів початкового алфавіту**m1.** За допомогою звичної методики Хаффмена будується допоміжний ОНК для меншого алфавіту. Кодова комбінація даного допоміжного ОНК є префіксом кодової комбінацій шуканого ОНК для символів, що становлять дану групу. Як доповнення (суфіксів) до цього префікса приписується кодова комбінація рівномірного коду на n розрядів, де **аn** - число символів в даній групі. Приклад такої побудови ОНК приведений в табл.2.

Таблиця 1



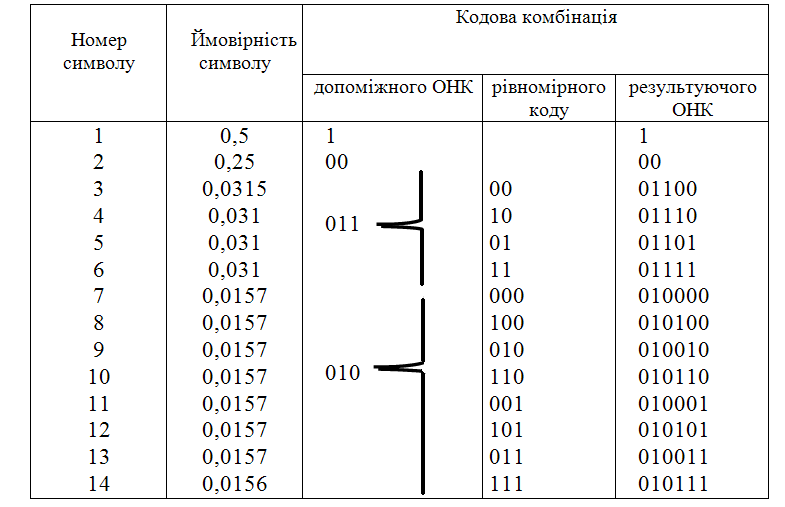
**Завдання по роботі**

1. Користуючись методикою Хаффмена побудувати ефективний ОНК для кодування букв алфавіту російської мови. Вірогідність *p(i)* букв узяти з результатів виконання лабораторної роботи № 1.

2. Побудувати кодове дерево одержаного коду. Визначити середню тривалість кодових комбінацій одержаного коду. Обчислити коефіцієнти стиснення і відносної ефективності для одержаного коду.

3. Користуючись штучним прийомом Мічеля В.С. побудувати ефективний код для кодування букв алфавіту української мови. Обчислити коефіцієнти стиснення і відносної ефективності для одержаного коду. Порівняти ефективності двох побудованих кодів.

Таблиця 2



**Порядок виконання**

1. Узяти деяку вірогідність *p(i)* букв української мови. По методиці Хаффмена побудувати ефективний код. Виконати завдання пунктів 2 і 3.

2. Оформити звіт по роботі. Звіт повинен містити завдання і порядок його виконання з аналізом одержаних результатів.

# Контрольні питання

1. Які коди відносяться до ефективних?
2. У чому полягає зміст методики Хаффмена?
3. У чому гідність методики Хаффмена в порівнянні з методикою Шеннона - Фано?

**Література**

1. Ігнатов В.А. Теорія інформації і передача сигналів. М.: Сов. Радіо, 1999. 320 с.

2. Новік Д.А. Ефективне кодування. М.: “Енергія”, -1995. -236 с.

**Лабораторна робота № 3**

# ПОРІВНЯННЯ РІЗНИХ ВЕЙВЛЕТ-ПЕРЕТВОРЕНЬ ПРИ СТИСКАННІ

**Мета роботи**

Вивчити алгоритм стиснення за допомогою вейвлет-перетворення, оцінити якість стисненого зображення і визначити найкращий базис вейвлет-функцій для заданих зображень.

**Підготовка до виконання роботи**

При підготовці до виконання роботи необхідно:  
- Вивчити основні положення вейвлет-перетворень;  
- Виконати завдання до лабораторної роботи;  
- Підготувати бланк звіту;  
- Підготувати відповіді на контрольні запитання.

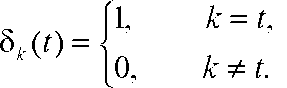
Основні положення вейвлет-перетворення

Вейвлети (wavelets) - це узагальнена назва часових функцій, що мають вигляд хвильових пакетів тієї чи іншої форми, локалізованих по осі незалежної змінної (*t* або *х*) і здатних до зміщення по ній і масштабуванню (стиску/розтягування). Вейвлети створюються за допомогою спеціальних базових функцій - прототипів, які задають їх вигляд і властивості.

Вейвлети і засновані на них інтегральні вейвлет-перетворення були запропоновані на початку 90-х років минулого століття (хоча перший найпростіший тип вейвлета, по суті був описаний Хааром ще в 1909 році) і в подальший час інтенсивно розвивалися.

Набір вейвлетів може наближати складний сигнал або зображення, причому ідеально точно або з деякою похибкою. Завдяки прекрасному поданню *локальних особливостей* сигналів, принципово відсутньому у рядів Фур'є, і безлічі видів вейвлети знайшли практичне застосування для аналізу тонких особливостей складних сигналів і зображень для їх стиснення і очищення від шуму.

Ряд Фур'є використовує в якості базисних функцій синусоїди. Вони гранично локалізовані в частотній області (перетворюючись на спектрограмі у вертикальну лінію), але дуже погано локалізовані (точніше, взагалі не локалізовані) у часовій області. Протилежний приклад - імпульсна базисна функція



Вона чітко локалізована в часовій області і тому ідеально підходить для представлення розривів сигналу. Але ця базисна функція не несе інформації про частоту сигналу і тому погано пристосована для подання сигналів на заданому відрізку часу і тим більше періодичних сигналів.

Вейвлети займають проміжне положення між розглянутими крайніми випадками (синусоїдою і імпульсною функцією). Базисними функціями вейвлетів можуть бути різні функції, в тому числі що нагадують модульовані імпульсами синусоїди, функції зі стрибками рівня і т. д. Це забезпечує легке подання сигналів з локальними скачками і розривами наборами вейвлетів того чи іншого типу. Майже всі вейвлети не мають аналітичного представлення у вигляді однієї формули і можуть даватися ітераційними виразами.

Одна з основних ідей вейвлет-представлення сигналів полягає в розбивці сигналу на дві складові - грубу (апроксимуючу) і витончену (деталізуючу) - з наступним їх подрібненням з метою зміни рівня декомпозиції сигналу.

Число використовуваних при розкладанні сигналу вейвлетів задає рівень декомпозиції сигналу. При цьому за нульовий рівень декомпозиції приймається сам сигнал, а рівні декомпозиції утворюють спадаюче вейвлет-дерево того чи іншого виду. Точність представлення сигналу в міру переходу на більш низькі рівні декомпозиції знижується, але зате з'являється можливість вейвлет-фільтрації сигналів, видалення з сигналів шумів та ефективної компресії сигналів.

*Пряме вейвлет-перетворення* означає розкладання довільного вхідного сигналу на принципово новий базис у вигляді сукупності хвильових пакетів - вейвлетів, які характеризуються чотирма принципово важливими властивостями:

* мають вигляд коротких, локалізованих у часі (або в просторі) хвильових пакетів з нульовим значенням інтеграла;
* мають можливість зсуву за часом;
* здатні до масштабування (стиску / розтягування);
* мають обмежений (або локальний) частотний спектр.

В основі безперервного вейвлет-перетворення лежить використання двох безперервних і інтегровних по всій осі *t* (або *х*) функцій:

* вейвлет-функція psi *ψ (t)* з нульовим значенням інтеграла**, що визначає деталі сигналу і породжує деталізуючі коефіцієнти:

** (1,а)

* маштабуюча функція phi φ*(t)* з одиничним значенням інтеграла**, що визначає грубе наближення (апроксимацію) сигналу і породжує коефіцієнти апроксимації:

. (1.б)

Phi-функції φ (t) притаманні далеко не всім вейвлетам, а тільки тим, які належать до ортогональних, тобто таких, у яких інтеграл від добутку будь-яких двох функцій ряду дорівнює нулю.

Властивість ортогональності помітно полегшує аналіз, дає можливість реконструкції сигналів (повного і точного відтворення) та дозволяє реалізувати алгоритми швидких вейвлет-перетворень.

Один з перших відомих ортогональних вейвлетів - *вейвлет Хаара* (haar). Функція psi у нього має вигляд прямокутних імпульсів меандру (значення 1 в інтервалі [0,0.5] і -1 в інтервалі [0.5,1]). Функція phi (див. рис. 1) має значення 1 в інтервалі [0,1] і 0 за межами цього інтервалу. Вейвлети Хаара добре локалізовані в просторі, але не дуже добре локалізовані в частотній області, оскільки меандр має широкий спектр частот (теоретично нескінченний).

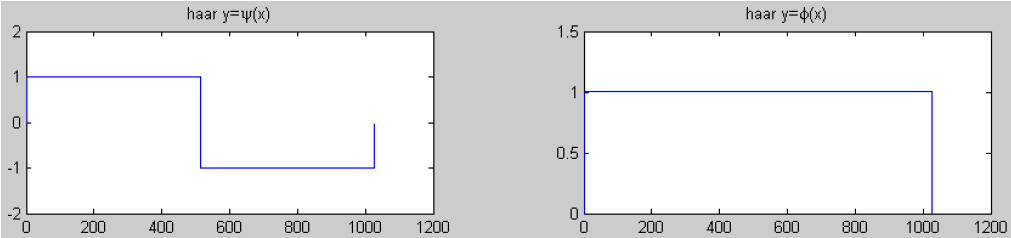


Рисунок 1- Функції psi и phi вейвлета Хаара

Перетворення Хаара в загальному вигляді для одновимірного сигналу (відліків) спрощується в порівнянні з формулами (1, *а* і *б*) і виглядає наступним чином. Нехай є одновимірний дискретний сигнал *s*. Кожній парі елементів з індексами *2j* і *2j + 1*, *j  Z*, поставимо у відповідність два значення:

, . (2)

До сигналу *aj* можна застосувати аналогічну операцію і так само отримати два сигнали, один з яких є огрублене версією *aj*, а інший містить деталізуючу інформацію, необхідну для відновлення *aj*.

Зворотне перетворення Хаара виглядає наступним чином:

, . (3)

Розглянемо конкретний приклад: нехай *si*: (220, 211, 212, 218, 217, 214, 210, 202). Ми отримаємо такі послідовності: а1 (215.5, 215, 215.5, 206) і d1 (4.5, -3, 1.5, 4). Повторимо операцію, розглядаючи а1 як si. Ми отримаємо з (215.5, 215, 215.5, 206): (215.25, 210.75) (0.25, 4.75).

На прикладі перетворення Хаара добре видно структуру вейвлет-перетворення дискретного сигналу. На кожному кроці перетворення сигнал розпадається на дві складові: наближення з більш низькою роздільною здатністю - апроксимацію та деталізуючи інформацію.

Розглянемо двовимірний сигнал - *s*-матрицю кінцевого або нескінченного розміру. Застосуємо до кожного рядка матриці один крок одновимірного вейвлет-перетворення. У результаті вийде дві матриці, рядки яких містять апроксимовану і деталізуючу складові рядків вихідної матриці. До кожного стовпця обох матриць також застосуємо крок одновимірного перетворення. У результаті виходить чотири матриці. Перша є апроксимуючою складової вихідного сигналу (огрубленою версією), інші три містять деталізуючу інформацію - вертикальну, горизонтальну і діагональну. Таким чином, крок двовимірного перетворення звівся до композиції одновимірних перетворень. Тому реалізація двовимірного перетворення не вимагає ніяких додаткових операцій.

Наприклад, для зображення 512x512 пікселів отримаємо після першого перетворення 4 матриці розміром 256x256 елементів (рис. 2).

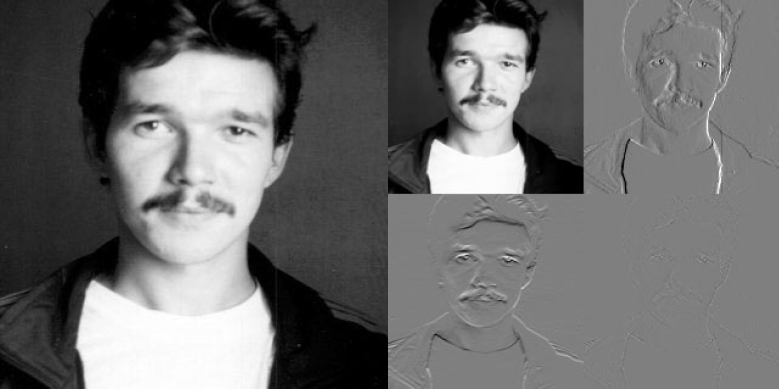


Рисунок 2 - Реалізація двовимірного вейвлет-перетворення.

У першій, як легко здогадатися, буде зберігатися зменшена копія зображення. У другій - усереднені різниці пар значень пікселів по горизонталі. У третій - усереднені різниці пар значень пікселів по вертикалі. У четвертій - усереднені різниці значень пікселів по діагоналі. За аналогією з двовимірним випадком ми можемо повторити наше перетворення і отримати замість першої матриці 4 матриці розміром 128x128. Повторивши наше перетворення втретє, ми отримаємо в результаті: 4 матриці 64x64, 3 матриці 128x128 і 3 матриці 256x256.

Наприклад, для матриці



на першому етапі при застосуванні вейвлет-перетворення до кожного рядка матриці одержуємо 2 матриці:

 і 

або

 і 

Далі застосовуємо перетворення Хаара до кожного стовпця матриць:

Для першої матриці

 і 

Для другої матриці

 і 

або

низькочастотна складова вертикальне відхилення

 і 

горизонтальне відхилення діагональне відхилення

 і 

Відновлення вихідної матриці відбувається в зворотному порядку відповідно до формул (3).

Оскільки для повної реконструкції сигналу можуть бути застосовані тільки ортогональні вейвлети, а вейвлет Хаара володіє «негладкістю», Інгрід Добеші запропонувала використовувати функції, які обчислюють ітераційним шляхом, згодом названі вейвлетами Добеші. Вони володіють наступними властивостями: ортогональністю, компактним носієм (тобто середнє значення функції дорівнює нулю і функція швидко спадає на нескінченності), а також ці функції *n+2* разів перетинають вісь абсцис. При цьому *n* називають порядком вейвлета. При *n = 1* отримуємо вейвлет Хаара. На рис. 3 представлені вейвлети Добеші порядку 2, 4 і 10.

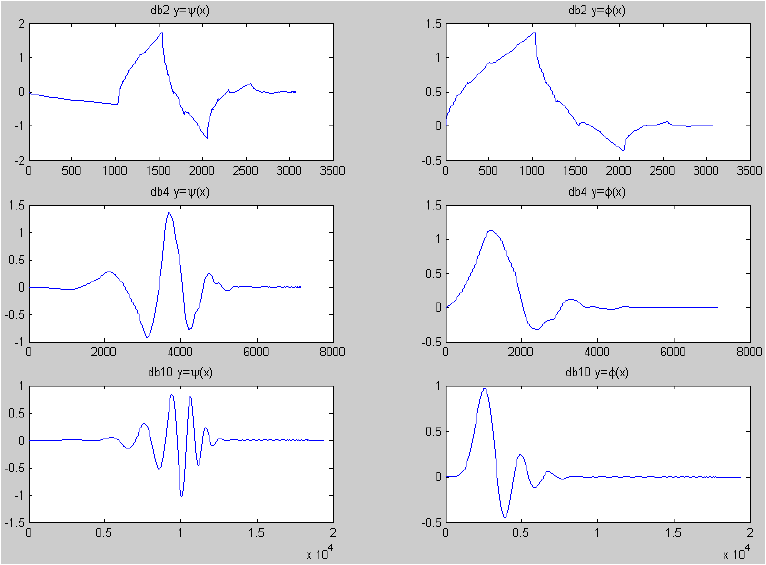


Рисунок 3 - Вейвлети Добеші порядку 2, 4 і 10

Як видно з рисунка, при збільшенні порядку вейвлета зростає «гладкість» вейвлета, що збільшує його можливості, але при цьому також збільшуються обсяги обчислень при перетворенні.

Вейвлети Добеші не можуть володіти симетричністю, що звужує їх використання. Однак можна спробувати наблизитися, наскільки можливо, до симетрії. Такі вейвлети, отримані з вейвлетів Добеші, називаються симлетами.

Питання про побудову вейвлетів, у яких нульові моменти має не тільки функція вейвлета *ψ(x)*, а й породжуючий вейвлет *φ(x)* було вперше поставлено Р. Койфманом, тому такі вейвлети називаються койфлетами. Наявність нульових моментів у породжуючих вейвлетах полегшує аналіз і вейвлет-перетворення. Койфлети несиметричні, проте вони більш симетричні, ніж вейвлети Добеші. Вид функцій симлетів і койфлетів зображений на рис. 4

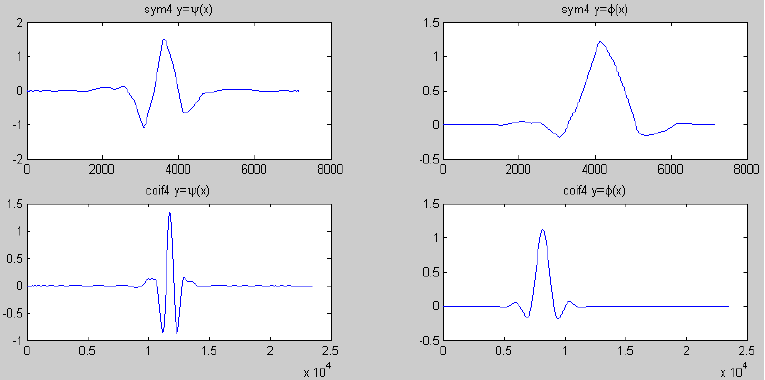


Рисунок 4 - Вигляд функцій симлетів і койфлетів 4-го порядку

***Стиснення зображень за допомогою вейвлет-перетворення   
(JPEG-2000)***

Алгоритм стискання за допомогою вейвлет-перетворення ідентичний алгоритму JPEG. Різниця полягає в застосуванні замість ДКП вейвлет-перетворення, що має такі переваги:

* при великих коефіцієнтах стискання за допомогою вейвлет-перетворення зображення стає нечітким (розмитим), що сприймається оком людини набагато краще, ніж блокова структура JPEG;
* можливість використання різних функцій як базисних, а також створення нових вейвлетів для різних типів сигналів для більш точного наближення до них;
* можливість поступового перегляду зображення в процесі завантаження зображення по мережі.

Втрати якості в даному алгоритмі відбуваються за рахунок деталізуючої інформації наступним чином. Всі значення коефіцієнтів, менші граничного, обнуляються, а решта округляються до цілого. Чим більше порогове значення, тим більші втрати деталізуючої інформацію, і, отже, тим більш розмито зображення.

Завдання до практичного заняття

1. Згідно теоретичних відомостей вивчити принципи стискання за допомогою вейвлет-перетворення.

2. Провести розрахунок двовимірного перетворення Хаара для матриці вихідного зображення:

|  |  |  |  |
| --- | --- | --- | --- |
| N | N+1 | N+2 | N-1 |
| N+3 | N-1 | N-2 | N |
| N-3 | N+2 | N+1 | N+2 |
| N+1 | N+2 | N+3 | N+1 |

де N - останні 2 цифри студентського квитка.

Округлити коефіцієнти в отриманій матриці до цілих чисел.

Відновити вихідну матрицю з округлених коефіцієнтів і розрахувати середньоквадратичне відхилення.

Завдання до лабораторної роботи

Метою лабораторної роботи є вивчення алгоритму стиснення за допомогою вейвлет-перетворення, оцінка якості стисненого зображення і визначення найкращого базису вейвлет-функцій для заданих зображень. У процесі виконання лабораторної роботи необхідно:

1. Накреслити в протоколі лабораторної роботи структурну схему алгоритму стиснення за допомогою вейвлетів (у керівництві вона не представлена).

2. Запустити з робочого столу файл Wavelet\_Compression. Для початку роботи натиснути кнопку «Пуск», вибрати малюнок, що відповідає номеру бригади, і вейвлет зі списку. По виду вейвлета визначити його порядок. Отримані на екрані дані звести в таблицю виду табл. 1:

Таблиця 1 -Підсумки виконання лабораторної роботи

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Вейвлет | Розмір файла, байт | | Коефіцієнт стискання | Оцінка якості | |
| вихідного | стисненого | СКВ | Суб‘єктивна |
|  |  |  |  |  |  |

Стиснення необхідно провести для усіх представлених у списку видів вейвлетів. Коефіцієнт стиснення розраховується самостійно. Суб’єктивна оцінка стискання виконується шляхом порівняння початкового та стиснутого зображення.

Ключові питання

* + - 1. У чому полягає ідея стискання за допомогою вейвлетів?
      2. Що таке вейвлети, породжуючі вейвлети?
      3. Якими властивостями володіють ортогональні вейвлети з компактним носієм?
      4. У чому полягає одномірне перетворення Хаара?
      5. Яким чином зберігається інформація про зображення при двовимірному перетворенні Хаара?
      6. Чим відрізняються вейвлети Хаара, Добеші, симлети і койфлети між собою?
      7. Назвіть переваги вейвлет-перетворення перед ДКП.

ЛАБОРАТОРНА РОБОТА № 4

**АЛГОРИТМ СТИСКАННЯ ДАНИХ JPEG**

**Мета роботи**

Закріпити теоретичні знання та набути навичок ефективного стискання кольорових і напівкольорових даних для факсимільного зв’язку за схемою JPEG.

**Підготовка до виконання роботи**

При підготовці до виконання роботи необхідно:  
- Ознайомитися з описом лабораторної роботи;

* Вивчити принципи стискання даних за схемою JPEG;
* Виконати завдання для практичного заняття;
* Виконати завдання до лабораторної роботи;
* Підготувати бланк звіту;
* Підготувати відповіді на контрольні запитання.

**Короткі відомості з теорії**

В 1986 році підгрупою ССІTT були початі дослідження методів стискання кольорових і напівтонових даних для факсимільного зв'язку. Застосовувані при цьому методи стиску кольорових даних дуже нагадували ті, які досліджувалися гpyпою JPEG. Тому було ухвалене рішення об'єднати ресурси цих груп для спільної pобoти над єдиним стандартом.

JPEG не був визначений як стандартний формат файлів зображень, однак на його основі були створені нові або модифіковані існуючі файлові формати. Оперує алгоритм областями 8x8 біт, на яких яскравість і колір міняються порівняно плавно, внаслідок цього, при розкладанні матриці такої області в подвійний ряд по косинусах значимими виявляються лише перші коефіцієнти. Таким чином, стиск у JPEG здійснюється за рахунок плавності зміни кольорів у зображенні.

У цілому алгоритм ґрунтується на дискретному косинусоїдальному перетворенні (ДКП), що є різновидом дискретного перетворення Фур'є, застосовуваному до матриці зображення для одержання деякої нової матриці коефіцієнтів. Для одержання вихідного зображення застосовується зворотне перетворення.

ДКП розкладає зображення за амплітудами деяких частот. Таким чином, при перетворенні одержується матриця, у якій багато коефіцієнтів, близьких або рівних нулю. Крім того, завдяки недосконалості людського зору, можна апроксимувати коефіцієнти більш грубо без помітної втрати якості зображення.

Для цього використається квантування коефіцієнтів. У найпростішому випадку - це арифметичний побітовий зсув вправо. При цьому перетворенні губиться частина інформації, але досягаються більші коефіцієнти стиску.

Процес стиску за схемою JPEG включає ряд етапів (рис. 1):

* Перетворення зображення в оптимальний колірний простір.
* Субдискретизація компонентів кольоровості усередненням груп пікселів.
* Застосування дискретних косинусних перетворень для зменшення надмірності даного зображення.
* Квантування кожного блоку коефіцієнтів ДКП із застосуванням вагових функцій, оптимізованих з урахуванням візуального сприйняття людиною.
* Кодування результуючих коефіцієнтів (даного зображення) із застосуванням алгоритму групового кодування й алгоритму Хаффмена для видалення надмірності інформації.

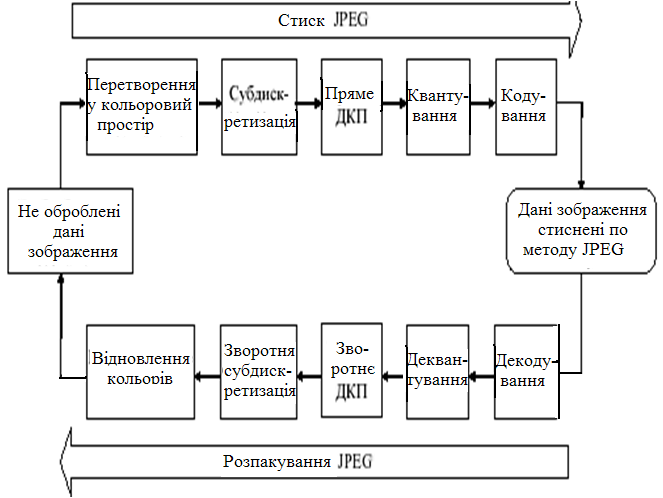


Рис. 1 - Структура JPEG-перетворень

Розглянемо коротко особливості кожного з перерахованих етапів. При цьому звернемо увагу на те, що декодування JPEG здійснюється у зворотному до описаного вище порядку.

***Перетворення зображення в оптимальний колірний простір***

У принципі, алгоритм JPEG здатний кодувати зображення, що ґрунтуються на будь-якому типі колірного простору (наприклад, розбитті кольору на три складові [червоний, зелений і синій] або [яскравість, хроматичний червоний, хроматичний синій] й ін.). JPEG кодує кожен компонент колірної моделі окремо, що забезпечує його повну незалежність від будь-якої моделі колірного простору.

У випадку застосування колірного простору яскравість/кольоровість, наприклад такого, як YUV або YCbCr, досягається кращий ступінь стиску. Компонента Y являє собою інтенсивність, a U(Cb) і V(Cr) - кольоровість (хроматичний червоний, хроматичний синій). Ця модель може бути переведена в RGB за допомогою перетворення без якої-небудь корекції насиченості. Для напівтонових зображень (у градаціях сірого) використовується тільки одна складова Y.

Спрощено переклад з колірного простору RGB у колірний простір YCrCb можна представити в такий спосіб:

Y = 0.299R + 0.587G + 0.114B

Сr = 0.5R – 0.4184G - 0.0813В+ 128

Cb = 0.1687R-0.3313G + 0.5В +128

Зворотне перетворення здійснюється так:

R=Y + 1.402Cb

G = Y- 0.34414Cr - 0.71414Cb – 128

В = Y + 1.772Сr – 128

***Субдискретизація компонентів кольоровості***

Більша частина візуальної інформації, до якої найбільш чутливі очі людини, складається з високочастотних, напівтонових компонентів яскравості (Y) колірного простору YCbCr. Дві інших складові кольоровості (Сb і Сr) містять високочастотну колірну інформацію, до якої око людини менш чутливе. Отже, певна її частина може бути відкинута й, тим самим, можна зменшити кількість врахованих пікселів для каналів кольоровості. Наприклад, у зображенні розміром 1000x1000 пікселів можна використати яскравості всіх 1000x1000 пікселів, але тільки 500x500 пікселів для кожного компонента кольоровості. При такому поданні кожен піксель кольоровості буде охоплювати ту ж область, що й блок 2x2 пікселя (для яскравості). У результаті ми збережемо для кожного блоку 2x2 усього 6 піксельних значень (4 значення яскравості і по 1 значенню для кожного із двох каналів кольоровості) замість того, щоб використати 12 значень при звичайному описі. Практика показала, що зменшення обсягу даних на 50% майже непомітно відбивається на якості більшості зображень.

Однак у випадку загальноприйнятих колірних моделей типу RGB таке подання даних неможливе, оскільки кожен колірний канал RGB несе деяку інформацію яскравості й будь-яка втрата роздільної здатності досить помітна.

Зменшення роздільної здатності каналів кольоровості шляхом субдискретизації, або усереднення груп пікселів здійснюється компресором JPEG.

***Сегментація зображення***

Сегментація зображення застосовується з метою поділу його на дві й більше частини (підзображення). Це полегшує буферизацію даних зображення в пам'яті ПЕОМ, прискорює їхню довільну вибірку з диска, і дозволяє зберігати зображення розміром понад 64x64 Кб. JPEG підтримує три типи сегментації зображень: просту, пірамідальну й комбіновану.

При *простій сегментації* зображення ділиться на два або більше сегментів фіксованого розміру. Всі прості сегменти кодуються зліва направо і згори донизу, є суміжними й не перекриваючими. Сегменти повинні мати однакову кількість вибірок й ідентифікаторів компонентів, і бути закодованими по одній схемі. Сегменти в нижній і правій частинах зображення можуть бути меншого розміру, ніж "внутрішні" сегменти, оскільки величина зображення не обов'язково повинна бути кратною розмірам сегмента.

При *пірамідальній сегментації* зображення також ділиться на сегменти, а кожний з них, у свою чергу, - на ще більш дрібні сегменти. При цьому використовуються різні рівні роздільної здатності. Моделлю такого процесу є сегментована піраміда зображення JPEG (JPEG Tіled Іmage Pyramіd, JTІР), що відбиває процедуру створення пірамідального JPEG-зображення з декількома рівнями роздільної здатності.

*Комбінована сегментація* дозволяє зберігати й відтворювати версії зображень із декількома рівнями роздільної здатності у вигляді мозаїки. Комбінована сегментація допускає наявність сегментів, що перекриваються, різних розмірів, з різними коефіцієнтами масштабування й параметрами стискання. Кожен сегмент кодується окремо й може комбінуватися з іншими сегментами без повторної дискретизації.

Наприклад, у випадку використання сегментів розміром 8х8 пікселів, для кожного блоку формується набір чисел. Перші кілька чисел представляють колір блоку в цілому, у той час як наступні числа відбивають більш тонкі деталі. Спектр деталей базується на зоровому сприйнятті людини, тому великі деталі більш помітні. На наступному етапі, залежно від обраного рівня якості, відкидається певна частина чисел, що представляють тонкі деталі.

Таким чином, чим вище рівень компресії, тим більше даних відкидається й тем нижча якість зображення. Використовуючи JPEG, можна одержати файл в 1-500 разів менший, ніж формат нестиснених зображень BMP.

***Дucкpemне косинусне перетворення.***

Ключовим компонентом роботи алгоритму є дискретне косинусне перетворення. Дискретне косинусне перетворення являє собою різновид перетворення Фур'є й, так само як і воно, має зворотне перетворення. Графічне зображення можна розглядати як сукупність просторових хвиль, причому осі X й Y збігаються із шириною й висотою картинки, а по осі Z відкладається значення кольору відповідного пікселя зображення. Дискретне косинусне перетворення дозволяє переходити від просторового подання картинки до її спектрального подання й назад. Впливаючи на спектральне подання картинки, що складається з "гармонік", тобто відкидаючи найменш значимі з них, можна балансувати між якістю відтворення й ступенем стискання. При цьому утворюється матриця, у якій коефіцієнти в лівому верхньому куті відповідають низькочастотній складовій зображення, а в правому нижньому - високочастотній.

Це перетворення можна представити так:



де

- гармоніка сигналу,

- постійна складова.

Вираз для зворотного перетворення матриці "гармонік", що застосовується при розпакуванні зображення, записується у вигляді



За визначенням дискретного косинусного перетворення, для його реалізації потрібно два вкладених цикли, і тіло циклів буде виконуватися *n\*n* разів для кожного елемента матриці дискретного косинусного перетворення. Значно більш ефективний варіант обчислення коефіцієнтів дискретного косинусного перетворення реалізований через множення матриць. У цьому випадку схему обчислення частотних коефіцієнтів матриці Y доцільно представити у вигляді множення матриць відповідно до відношення

 (\*)

де *у -* матриця вхідного зображення; *Х* - матриця постійних коефіцієнтів косинусного перетворення розміру *n\*n*, значення елементів якої обчислюються за формулою



*ХТ -* транспонована матриця *Х*.

Цей варіант реалізації ДКП більш привабливий ще й тому, що сучасні архітектури багатопроцесорних обчислювачів виконують стандартні матричні операції множення й транспонування. При перемножуванні двох матриць розміру *п\*п* для обчислення одного елемента результуючої матриці необхідно виконати *п* множень і *п* додавань.

***Квантування.***

Дискретне косинусне перетворення являє собою перетворення інформації без втрат і не здійснює ніякого стиску. Проте дискретне косинусне перетворення підготовляє інформацію для етапу стиску із втратами або округлення. Округлення являє собою процес зменшення кількості бітів, необхідних для зберігання коефіцієнтів матриці дискретного косинусного перетворення за рахунок втрати точності. Стандарт JPEG реалізує цю процедуру через матрицю квантування. Для кожного елемента матриці дискретного косинусного перетворення існує відповідний елемент матриці квантування. Результуюча матриця отримується шляхом ділення кожного елемента матриці дискретного косинусного перетворення на відповідний елемент матриці квантування й наступним округленням результату до найближчого цілого числа:



де *Е*[ ] - ціла частина від ділення; *q*[*u, v*] - матриця квантування.

Як правило, значення елементів матриці квантування ростуть у напрямку з ліва направо і згори донизу. Від вибору матриці квантування залежить баланс між ступенем стиску зображення і його якістю після відновлення. Стандарт JPEG дозволяє використати будь-яку матрицю квантування, однак ІSO розробила набір матриць округлення.

На цьому етапі більшість JPEG-компресорів керуються за допомогою установки якості. Компресор використає вбудовану таблицю, розраховану на середню якість, і нарощує або зменшує значення кожного елемента таблиці обернено пропорційно необхідній якості. Застосовувані таблиці квантування записуються до стисненого файлу, щоб декомпресор знав, як відновити коефіцієнти ДКП.

Із квантуванням пов'язані й специфічні ефекти алгоритму. При більших значеннях фактора якості втрати в низьких частотах можуть бути настільки великі, що зображення розпадеться на квадрати 8x8. Втрати у високих частотах можуть виявитися в так званому "ефекті Гіббса", коли навколо контурів з різким переходом кольору утвориться своєрідний "німб".

***Кодування***

Переводимо матрицю 8x8 в 64-елементний вектор за допомогою "зиґзаґ" –сканування (рис.2), тобто беремо елементи з індексами (0,0), (0,1), (1,0), (2,0), ...

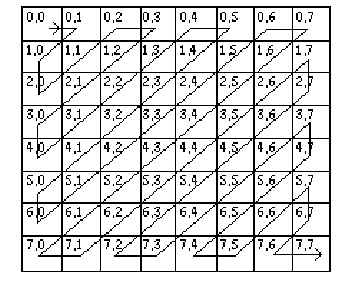


Рис. 2 - "Зиґзаґ" - сканування матриці.

Таким чином, на початку вектора ми одержуємо коефіцієнти матриці, що відповідають низьким частотам, а наприкінці - високим.

Заключна стадія роботи кодера JPEG - це власне кодування. Воно включає три дії над округленою матрицею дискретного косинусного перетворення для того, щоб підвищити ступінь стиску. Перша дія - це заміна абсолютного значення коефіцієнта, розташованого в комірці (0,0) матриці, на відносне. Оскільки сусідні блоки зображення значною мірою "схожі" один до одного, кодування чергового (0,0) елемента як різниці з попереднім дає менше значення.

Коефіцієнти матриці дискретного косинусного перетворення обходяться зиґзаґом, після чого нульові значення кодуються з використанням алгоритму кодування повторів (RLE), а потім результат обробляються за допомогою "кодування ентропії", тобто алгоритмів Хаффмана або арифметичного кодування в залежності від реалізації.

***Оцінка втрат якості***

Одна із серйозних проблем машинної графіки полягає в тому, що дотепер не знайдено адекватний критерій оцінки втрат якості зображення. А губиться воно постійно - при оцифровці, при перекладі в обмежену палітру кольорів, при перекладі в іншу систему кольорового представления для друку, і, що для нас особливо важливо, при архівації із втратами. Можна навести приклад простого критерію: середньоквадратичне відхилення (СКВ) значень пікселів:



По ньому зображення буде сильно змінене при зниженні яскравості всього на 5% (око цього не помітить - у різних моніторів настроювання яскравості варіюється набагато сильніше). У той же час зображення зі "снігом" - різкою зміною кольору окремих точок, слабкими смугами або "муаром" будуть визнані "майже незмінними". Свої неприємні сторони є й в інших критеріїв. Розглянемо, наприклад, максимальне відхилення (МВ):



Цей критерій, як можна здогадатися, украй чутливий до коливань яскравості окремих пікселів, тобто у всьому зображенні може істотно змінитися тільки значення одного пікселя (що практично непомітно для ока), однак, відповідно до цього критерію, зображення буде сильно зіпсовано.

Критерій, що зараз використовують на практиці, називається мірою відношення сигналу до шуму (ВСШ):



Даний критерій, по суті, аналогічний середньоквадратичному відхиленню,однак користуватися ним дещо зручніше за рахунок логарифмічного масштабу шкали. Йому притаманні ті ж недоліки, що й середньоквадратичному відхиленню.

Найкраще втрати якості зображень оцінюють наші очі. Відмінною вважається архівація, при якій неможливо на око розрізнити первісне й разархівоване зображення. Гарною - коли сказати, яке із зображень піддавалося архівації, можна, тільки порівнюючи дві картинки, що перебувають поруч. При подальшому збільшенні ступеня стиску, як правило, стають помітні побічні ефекти, характерні для даного алгоритму. На практиці, навіть при відмінному збереженні якості, у зображення можуть бути внесені регулярні специфічні зміни.

**Завдання до практичного заняття**

1. По вступу до даної роботи вивчити принципи алгоритму JPEG.

2. Провести розрахунок дискретного косинусного перетворення по формулі (\*) для матриці вхідного зображення:

N N+l N+2 N-l

N+3 N-l N-2 N

N-3 N+2 N+l N+2

N+1 N+2 N+3 N+1

де N - останні 2 цифри номера студентського квитка. Матриця постійних коефіцієнтів косинусного перетворення розміром 4\*4 виглядає таким чином:

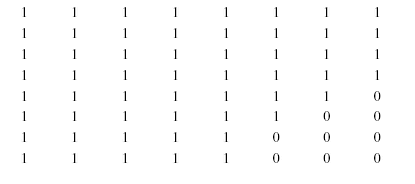
0.5 0.5 0.5 0.5

0.65328 0.2706 -0.2706 -0.65328

0.5 -0.5 -0.5 0.5

0.2706 -0.65328 0.65328 -0.2706

3. Підготувати маску квантування відповідно до фактора якості N. Маска квантування - це матриця розміром 8\*8, що складається з "1" й "0". Кількість "1" пропорційна N. Матриця заповнюється за принципом "зиґзаґ"-сканування. Наприклад, N=85. Кількість "1" розраховується в такий спосіб: [85\*64/100]=55. Маска квантування виглядає таким чином:



**Завдання до лабораторної роботи**

Метою лабораторної роботи є вивчення алгоритму стиску JPEG, оцінка якості стислого зображення й визначення найкращого фактора якості для заданих зображень. У процесі виконання лабораторної роботи необхідно:

1. Підготувати 2 маски квантування відповідно до факторів якості, заданими в таблиці 1.

Таблиця 1 - Варіанти факторів якості N

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № бригади | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |  |
| найгірший | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 |  |
| найкращий | 83 | 87 | 89 | 91 | 93 | 95 | 97 | 99 |  |

2. Запустити з робочого стола файл JPEG\_Compressіon. Для початку роботи нажати кнопку "Пуск", вибрати малюнок, що відповідає номеру бригади, а потім ввести маску квантування. Отримані на екрані дані звести в таблицю виду табл. 2:

Таблиця 2 - Підсумки виконання лабораторної роботи

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Фактор  якості | Розмір файлу, байт | | Коефіцієнт стискання | Оцінка якості | | | |
| початкового | стисненого | СКВ | МВ | ВСШ | Суб'єктивна |
|  |  |  |  |  |  |  |  |

Стискання необхідно провести для двох факторів якості, заданих у таблиці 2.1, а також індивідуально для кожного члена бригади відповідно до завдання для практичного заняття. Коефіцієнт стискання розраховується самостійно. Суб'єктивна оцінка стискання виробляється за зображенням зображенню "Відхилення".

3. Змінюючи маску квантування, знайти фактор якості, що відповідає суб'єктивній оцінці "5", при якому стискання буде максимальним. Дані додати в таблицю 2.2.

**Контрольні запитання**

1. За рахунок чого здійснюється стиск в алгоритмі JPEG?
2. Для чого необхідне перетворення колірного простору зображення?
3. Чому в алгоритмі не застосовується субдискретизація компонентів яскравості?
4. Назвіть види сегментації зображень і їхні відмінності. Яка сегментація використається в алгоритмі JPEG?
5. Для чого потрібно дискретне косинусне перетворення?
6. На якому етапі стиску відбуваються втрати якості зображення? Як вони пов'язані з фактором якості?
7. Яким чином проявляються втрати якості в алгоритмі JPEG?
8. Які алгоритми кодування використовує JPEG?
9. Приведіть приклади критеріїв втрати інформації та опишіть їхні недоліки.

1. Символи алфавіту **m** вважаються некорельованими**,** якщоміж елементами дискретних випадкових послідовностей, складених з цього алфавіту, відсутні статистичні зв’язки **(**статистичні залежності**).** [↑](#footnote-ref-1)