

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ ІМЕНІ
ВАДИМА ГЕТЬМАНА**

ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ В ЕКОНОМІЦІ

Кафедра системного аналізу та кібербезпеки

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА

Галузь знань

Спеціальність

Спеціалізація (за наявності)

«Кібербезпека»

12 «Інформаційні
технології»

125 «Кібербезпека»

Форма навчання: очна (денна)

КВАЛІФІКАЦІЙНА БАКАЛАВРСЬКА РОБОТА

на тему «Виявлення патернів при обробці зображень за допомогою вейвлет
функцій»

здобувача Ляшко Катерини Олексіївни

(підпис)

Науковий керівник: кандидат фізико-математичних наук, доцент,
Серденко Таїсія Володимирівна

(підпис)

**Робота допущена до захисту перед екзаменаційною комісією з
атестації здобувачів вищої освіти (ЕК)**

Завідувач кафедри: професор, доктор фізико-математичних наук,
Джалладова Ірада Агаєвна

(підпис)

Київ 2024

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА**

ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ В ЕКОНОМІЦІ

Кафедра системного аналізу та кібербезпеки

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА «КІБЕРБЕЗПЕКА»

Галузь знань 12 «Інформаційні технології»

Спеціальність 125 «Кібербезпека»

Форма навчання: очна (денна)

ПОГОДЖЕНО

Керівник проектної групи (гарант)
освітньо-професійної програми «Кібербезпека»
к. ф. – м. н., доцент Г.В. Мамонова

_____ 2024р.

ЗАТВЕРДЖУЮ:

Завідувач кафедри системного
аналізу та кібербезпеки
д. ф. – м. н., проф. І.А. Джалладова

_____ 2024р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

здобувача вищої освіти Ляшко Катерини Олексіївни

денної форми навчання

на підготовку кваліфікаційної бакалаврської роботи

***на тему* «Виявлення патернів при обробці зображень за допомогою
вейвлет функцій»**

**Тему затверджено наказом ректора Університету від «30» квітня 2024 р.
№ 726-ст**

Кваліфікаційна бакалаврська робота виконується на матеріалах

План кваліфікаційної бакалаврської роботи

Розділ 1	ПРИНЦИП РОБОТИ ВЕЙВЛЕТ-ПЕРЕТВОРЕННЯ (назва розділу)
Розділ 2	ПРАКТИЧНА РЕАЛІЗАЦІЯ ВЕЙВЛЕТ-ПЕРЕТВОРЕННЯ (назва розділу)
Розділ 3	РОЗПІЗНАВАННЯ ОБЛИЧ ЗА ДОПОМОГОЮ ВЕЙВЛЕТ-ПЕРЕТВОРЕННЯ (назва розділу)
Об'єкт дослідження:	Методи обробки зображень за допомогою вейвлет функцій.
Предмет дослідження:	Процес виявлення патернів при обробці зображень з використанням вейвлет функцій.
Мета кваліфікаційної бакалаврської роботи:	Дослідження методів виявлення патернів при обробці зображень за допомогою вейвлет функцій, а також розробка і реалізація системи розпізнавання облич.

Конкретні завдання, які здобувач повинен виконати для досягнення поставленої мети:

У розділі 1	– вивчення теоретичних основ вейвлет-перетворень; – ознайомлення з різними типами перетворень; – вивчення особливостей вейвлетів Хаара.
У розділі 2	– розглянути практичну реалізацію вейвлетів Хаара в середовищі програмного забезпечення MATLAB; – вивчення функцій MATLAB, призначених для роботи із зображеннями і їх практичне застосування.
У розділі 3	– сформулювати актуальність та огляд основних методів розпізнавання облич; – розглянути складові частини запропонованого алгоритму; – опис та оцінка роботи запропонованої системи розпізнавання облич.

**Завдання підготував
науковий керівник**

(підпис)

Серденко Т. В.

(ініціали, прізвище)

«_____» _____ 20__ р.

**Завдання одержав
здобувач**

(підпис)

Ляшко К. О.

(ініціали, прізвище)

«_____» _____ 20__ р.

РЕФЕРАТ

Кваліфікаційна бакалаврська робота: 75 сторінок, 2 таблиці, 29 рисунків та список використаних джерел з 32 найменувань.

Об'єкт дослідження: методи обробки зображень за допомогою вейвлет функцій.

Мета дослідження: дослідження методів виявлення патернів при обробці зображень за допомогою вейвлет функцій, а також розробка і реалізація системи розпізнавання облич.

Методи дослідження: аналіз літературних джерел, математичне моделювання за допомогою програми MATLAB, розробка алгоритму, програмна реалізація та експериментальне тестування.

Проведений теоретичний аналіз показав, що вейвлет-функції мають значну перевагу у порівнянні з іншими методами обробки сигналів, завдяки здатності забезпечувати одночасну локалізацію як у часовій, так і у частотній областях. Це робить вейвлети особливо корисними для задач розпізнавання патернів у зображеннях.

Розроблені алгоритми для обробки зображень з використанням вейвлет-функцій успішно реалізовані у середовищі MATLAB. Це дозволило створити ефективну систему, яка демонструє високу точність та швидкість обробки зображень, що підтверджено експериментальними даними.

Запропонована система розпізнавання облич на основі Хаар-вейвлетів показала високу ефективність. Тестування системи на різноманітних наборах даних продемонструвало точність розпізнавання облич понад 90%, що свідчить про її надійність та можливість використання у реальних умовах.

Отримані результати мають важливе значення для подальшого розвитку теорії вейвлет-перетворень та їх застосування в обробці зображень. *Методична значущість* полягає в тому, що розроблений алгоритм може бути використаний в подальших дослідженнях на дану тему. *Практична значущість* полягає в можливості застосування розробленої системи для розпізнавання облич у реальних умовах, що сприяє підвищенню рівня кібербезпеки.

Наукова новизна дослідження полягає у розробці та реалізації нових підходів до розпізнавання облич при обробці зображень за допомогою Хаар-вейвлетів.

Рекомендується *продовжити дослідження* в області застосування інших типів вейвлетів для обробки зображень, а також розширити функціональні можливості розробленої системи для інтеграції з іншими методами розпізнавання та аналізу даних.

Ключові слова: Вейвлет-перетворення, перетворення Фур'є, система розпізнавання облич, вейвлети Хаара, MATLAB.

ВІДГУК

на кваліфікаційну бакалаврську роботу
студентки Ляшко Катерини Олексіївни гр. ІК-401
на тему «Виявлення патернів при обробці зображень за допомогою вейвлет
функцій»

Актуальність теми: Кваліфікаційна бакалаврська робота Ляшко Катерини Олексіївни присвячена актуальній темі використання вейвлет функцій для обробки зображень та виявлення патернів. У сучасних умовах зростання кількості кіберзагроз, дана тема є надзвичайно важливою для покращення систем захисту інформації.

Повнота розкриття теми: Студентка повно розкрив тему, дослідивши теоретичні основи вейвлет-перетворень, різні типи вейвлетів та їх практичні застосування. Робота містить детальний аналіз літератури, алгоритмів і програмних реалізацій.

Теоретичний рівень роботи високий. Студентка продемонстрував глибоке розуміння теорії вейвлет-перетворень та їх застосування для обробки зображень. Виклад теоретичного матеріалу логічний і послідовний, що дозволяє легко слідувати за думкою автора.

Практична значущість роботи полягає у розробці ефективної системи розпізнавання облич на основі Нааг-вейвлетів. Система показала високу точність і швидкість обробки зображень, що підтверджено експериментальними даними. Це може бути застосовано у реальних умовах для підвищення рівня кібербезпеки.

Самостійність виконання роботи: Робота виконана студенткою самостійно, про що свідчить глибокий аналіз літератури, розробка алгоритмів та їх програмна реалізація у середовищі MATLAB. Студентка продемонструвала високий рівень технічних навичок та аналітичного мислення.

Якість оформлення, загальна та спеціальна грамотність: Оформлення роботи відповідає вимогам до кваліфікаційних бакалаврських робіт. Текст грамотно написаний, логічно структурований та зрозумілий. Використані ілюстрації та таблиці сприяють кращому розумінню матеріалу.

Переваги та недоліки роботи: Основними перевагами роботи є актуальність теми, високий теоретичний рівень, практична значущість та самостійність виконання. Недоліком можна вважати обмеження дослідження лише одним типом вейвлетів та відсутність порівняння з іншими методами розпізнавання.

Загальна оцінка роботи та висновки щодо рекомендації до захисту в ЕК: Кваліфікаційна бакалаврська робота Ляшко Катерини Олексіївни виконана на високому рівні, відповідає всім вимогам і заслуговує на високу оцінку – 60 балів. Робота рекомендується до захисту в екзаменаційній комісії.

Науковий керівник
Доцент кафедри системного
аналізу та кібербезпеки'



Т.В. Серденко

2024 р.

РЕЦЕНЗІЯ

на кваліфікаційну бакалаврську роботу

студентки Ляшко Катерини Олексіївни гр. ІК-401

на тему: «Виявлення патернів при обробці зображень за допомогою вейвлет функцій»

Актуальність теми: Робота присвячена актуальній темі кібербезпеки, зокрема використанню вейвлет-функцій для обробки зображень і виявлення патернів. У сучасному світі, де кількість кіберзагроз постійно зростає, дослідження нових методів захисту інформаційних систем є надзвичайно важливим напрямком захисту та аналізу інформації.

Наукова новизна роботи полягає в розробці та реалізації нових підходів до розпізнавання облич за допомогою Haar-вейвлетів. Отримані результати мають значний внесок у подальший розвиток теорії вейвлет-перетворень та їх практичних застосувань у кібербезпеці.

Якість проведеного аналізу: Аналіз літературних джерел та математичне моделювання проведені на високому рівні. В роботі використані сучасні методи та інструменти, зокрема програма MATLAB, що дозволило створити ефективну систему обробки зображень і розпізнавання облич.

Уміння користуватися літературними джерелами: Студентка продемонструвала хороші навички роботи з науковими джерелами. Використано широкий спектр літератури, включаючи наукові статті, монографії та електронні ресурси, що свідчить про глибоке розуміння теми дослідження.

Практична цінність висновків і рекомендацій: Практична значущість роботи полягає в можливості застосування розробленої системи для розпізнавання облич у реальних умовах, що сприяє підвищенню рівня кібербезпеки. Запропонована система показала високу ефективність з точністю розпізнавання облич понад 90%.

Переваги та недоліки: Основними перевагами роботи є використання сучасних методів обробки зображень, висока точність і швидкість обробки, а також детальний аналіз отриманих результатів. Недоліком можна вважати обмеженість дослідження лише одним типом вейвлетів та відсутність порівняння з іншими методами розпізнавання.

Загальний висновок і оцінка роботи

Бакалаврська робота Ляшко Катерини Олексіївни виконана на високому рівні, відображає глибоке розуміння теми і володіння сучасними методами обробки зображень. Робота заслуговує високої оцінки, а її результати можуть бути використані для подальших досліджень у сфері кібербезпеки.

Рецензент

К.М.Н., доцент кафедри
інформаційних управлюючих
систем та технологій
УжНУ


(підпис)


(ініціали, прізвище)

«14» червня 2024 р.

Зміст

ВСТУП.....	3
РОЗДІЛ 1	5
ПРИНЦИП РОБОТИ ВЕЙВЛЕТ-ПЕРЕТВОРЕННЯ.....	5
1.1 Основи теорії вейвлетів	5
1.1.1 Ряди Фур'є	6
1.1.2 Розкладання по вейвлетах	8
1.1.3 Зворотнє вейвлет-перетворення	12
1.2 Частотно-часова локалізація	15
1.3 Нааг-вейвлети	20
РОЗДІЛ 2	22
ПРАКТИЧНА РЕАЛІЗАЦІЯ ВЕЙВЛЕТ-ПЕРЕТВОРЕННЯ.....	22
2.1 Нааг-вейвлети в MATLAB	24
2.2 Функції MATLAB для роботи із зображеннями	40
РОЗДІЛ 3	62
РОЗПІЗНАВАННЯ ОБЛИЧ ЗА ДОПОМОГОЮ ВЕЙВЛЕТ-ПЕРЕТВОРЕННЯ	
ХААРА.....	62
3.2 Levenberg-Marquardt Backpropagation (LMBP) Neural Network	65
3.3 Пропонована система розпізнавання облич	66
3.3 Експерименти та результати тестування запропонованої системи.....	68
ВИСНОВКИ.....	70
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	71
ДОДАТКИ	

ВСТУП

У сучасному світі кібербезпека стає все більш важливою, оскільки кількість кіберзагроз і атак зростає з кожним днем. Одним з методів покращення захисту інформаційних систем є використання вейвлет-функцій для обробки зображень і виявлення патернів. Цей підхід дозволяє ефективно аналізувати та розпізнавати зображення, це має важливе значення для розвитку технологій захисту даних.

Метою даної бакалаврської роботи є дослідження методів виявлення патернів при обробці зображень за допомогою вейвлет функцій, а також розробка і реалізація системи розпізнавання облич на основі наступних завдань:

- Вивчення теоретичних основ вейвлет-перетворень;
- Ознайомлення з різними типами вейвлетів і їх застосуваннями на практиці;
- Розробка алгоритму для обробки зображень з використанням вейвлет функцій;
- Реалізація системи розпізнавання облич за допомогою Haar-вейвлетів;
- Проведення тестування та оцінка ефективності запропонованої системи.

Об'єктом дослідження є методи обробки зображень за допомогою вейвлет функцій.

Предметом дослідження є процес виявлення патернів при обробці зображень з використанням вейвлет функцій.

Для досягнення мети роботи використовувались наступні методи: аналіз літературних джерел, математичне моделювання за допомогою програми MATLAB, розробка алгоритму, програмна реалізація та експериментальне тестування.

Отримані результати мають важливе значення для подальшого розвитку теорії вейвлет-перетворень та їх застосування в обробці зображень. Методична значущість полягає в тому, що розроблений алгоритм може бути використаний в подальших дослідженнях на дану тему. Практична значущість полягає в

можливості застосування розробленої системи для розпізнавання облич у реальних умовах, що сприяє підвищенню рівня кібербезпеки.

Елементом наукової новизни є розробка та реалізація нових підходів до розпізнавання облич при обробці зображень за допомогою Haar-вейвлетів.

Інформаційною базою дослідження стали наукові статті, монографії, дисертації, а також електронні ресурси, присвячені тематиці вейвлет-перетворень і обробки зображень.

Структура роботи. Кваліфікаційна бакалаврська робота складається зі вступу, трьох розділів, висновків, списку використаних джерел. Загальний обсяг роботи становить 73 сторінки. Робота містить 2 таблиці, 29 рисунків, 1 додаток та список використаних джерел з 32 найменувань.

РОЗДІЛ 1

ПРИНЦИП РОБОТИ ВЕЙВЛЕТ-ПЕРЕТВОРЕННЯ

1.1 Основи теорії вейвлетів

Інтегральне перетворення Фур'є та ряди Фур'є є основою гармонічного аналізу. Коефіцієнти Фур'є, які утворюються в результаті перетворення, піддаються достатньо простій фізичній інтерпретації, причому простота ні в якому разі не применшує важливості наступних висновків про характер досліджуваного сигналу. Застосування інтегрального перетворення та рядів Фур'є (в обчисленнях, аналітичних перетвореннях) дуже наочно, усі необхідні властивості та формули виписуються за допомогою лише двох дійсних функцій $\sin t$, $\cos t$ (або однієї комплексної — синусоїдальної хвилі $\exp(it) = \cos t + i \sin t$, $i = \sqrt{-1}$ і достатньо легко доводяться.

Вейвлет-перетворення не так добре і широко відоме, оскільки застосовується порівняно недавно і математичний апарат знаходиться на стадії активної розробки. Тому для більшої наочності в даній роботі авторкою будуть розглянуті необхідні поняття вейвлет-аналізу, проведені аналогії та порівняння з аналізом Фур'є, значимість і привабливість котрого для широкого кола дослідників незаперечні й перевірені часом.

Визначення, властивості та їхні наслідки приводяться для одновимірних функцій, рядів даних. За необхідністю вся розглянута інформація може бути узагальнена та застосована для багатовимірних випадків. Для визначеності в цій роботі буде сказано про функції, які залежать від часу, про часові ряди і, відповідно, про частоти. Однак без порушення спільності незалежна координата

може бути просторовою (із відповідними хвильовими числами) і будь-якою іншою.[1]

1.1.1 Ряди Фур'є

Нехай $L^2(0, 2\pi)$ — простір квадратично інтегрованих функцій із кінцевою енергією (нормою)

$$\int_0^{2\pi} |f(t)|^2 dt < \infty, \quad t \in (0, 2\pi) \quad (1.1)$$

Формула (1.1) — це визначення шматково-безперервної функції $f(t)$. Вона може бути періодично розширеною та є визначеною на всій осі $R(-\infty, +\infty)$ таким чином, що

$$f(t) = f(t - 2\pi), \quad t \in R$$

Будь-яка функція $f(t)$ з простору 2π -періодичних квадратично інтегрованих функцій може бути подана у вигляді ряду Фур'є:

$$f(t) = \sum_{-\infty}^{+\infty} c_n \exp(int) \quad (1.2)$$

Коефіцієнти c_n у формулі (1.2) мають вигляд:

$$c_n = (2\pi)^{-1} \int_0^{2\pi} f(t) \exp(-int) dt \quad (1.3)$$

і ряд (1.2) рівномірно зводиться до $f(t)$:

$$\lim_{M,N \rightarrow \infty} \int_0^{2\pi} \left| f(t) - \sum_{-M}^N c_n \exp(int) \right|^2 dt = 0$$

Треба зауважити, що

$$w_n(t) = \exp(int), \quad n = \dots, -1, 0, 1, \dots \quad (1.4)$$

є ортонормованим базисом простору $L^2(0, 2\pi)$, який був збудований за допомогою масштабного перетворення єдиної функції $w(t) = \exp(it)$ таким чином, що $w_n(t) = w(nt)$.

Тобто, кожна 2π -періодична квадратично інтегрована функція може бути отримана за допомогою суперпозиції (підстановка функцій або змінних величин у функцію) масштабних перетворень базисної функції $w(t) = \exp(it) = \cos t + i \sin t$, тобто є композицією синусоїдальних хвиль із різними частотами (із коефіцієнтами, які залежать від номеру гармоніки).

Для коефіцієнтів рядів Фур'є виконується рівність Парсеваля [1]:

$$(2\pi)^{-1} \int_0^{2\pi} |f(t)|^2 dt = \sum_{-\infty}^{\infty} |c_n|^2 \quad (1.5)$$

1.1.2 Розкладання по вейвлетах

Розглянемо простір $L^2(R)$ функцій $f(t)$, що є визначеними по всій дійсній осі $R(-\infty, +\infty)$ і таких, що володіють кінцевою енергією (нормою):

$$E_f = \int_{-\infty}^{\infty} |f(t)|^2 dt < \infty \quad (1.6)$$

Функціональні простори $L^2(0, 2\pi)$ і $L^2(R)$ суттєво відрізняються одне від одного. Зокрема, локалізоване усереднене значення кожної функції з $L^2(R)$ має прагнути до нуля на $\pm\infty$. Синусоїдальна хвиля не належить $L^2(R)$, та, отже, сімейство синусоїдальних хвиль w_n не може бути базисом функціонального простору $L^2(R)$. Потрібно знайти достатньо прості функції для конструювання базису простору $L^2(R)$.

"Хвилі", які утворюють простір $L^2(R)$, повинні прагнути до нуля на $\pm\infty$ і для практичних цілей чим скоріше, тим краще. Розглянемо в якості базисних функцій вейвлети — добре локалізовані солітон-подібні вейвлети (Солітон — це одинична відокремлена хвиля, яка існує в нелінійному середовищі.) [2].

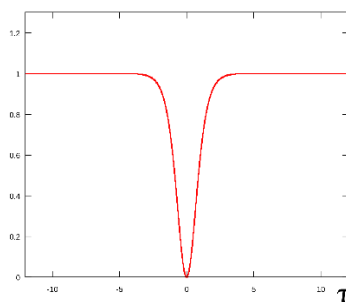


Рисунок 1.1 – Приклад солітон-подібного вейвлету.

Як і у випадку з простором $L^2(0, 2\pi)$, яке повністю формувалось із допомогою однієї базисної функції $w(t)$ сконструюємо функціональний простір $L^2(R)$ також із допомогою одного вейвлету $\psi(t)$. Важливо відмітити, що це може бути вейвлет з однією частотою або з набором частот (frequency bands). Варто почати з дискретних перетворень.

Щоб найбільш легко покрити всю вісь $R(-\infty, +\infty)$ за допомогою локалізованої функції, яка швидко прагне до нуля, можна передбачити систему зсувів (переносів) вздовж осі. Нехай для зрозумілості вони будуть цілими, тобто $\psi(t - k)$ [2].

Далі треба ввести аналог синусоїдальної частоти. Для простоти та визначеності вона буде записана через степені двійки: $\psi(2^j t - k)$, де j та k — цілі числа ($j, k \in I$).

Таким чином, за допомогою дискретних масштабних перетворень ($1/2^j$) та зсувів ($k/2^j$) можна зробити опис усіх частот і покрити всю вісь, маючи єдиний базисний вейвлет $\psi(t)$.

Визначення норми:

$$\|p\|_2 = \langle p, p \rangle^{\frac{1}{2}},$$

$$\langle p, q \rangle = \int_{-\infty}^{\infty} p(t) q^*(t) dt$$

(астериск позначає комплексне сполучення). Отже,

$$\|\psi(2^j t - k)\|_2 = 2^{-j/2} \|\psi(t)\|_2,$$

Тобто, якщо вейвлет $\psi(t) \in L^2(R)$ має одиничну норму, то всі вейвлети вигляду $\{\psi_{jk}\}$ і виду

$$\psi_{jk}(t) = 2^{j/2} \psi(2^j t - k), \quad j, k \in I \quad (1.7)$$

також є нормованими на 1, тобто $\|\psi_{jk}\|_2 = \|\psi\|_2 = 1$ [3].

Вейвлет $\psi \in L^2(R)$ можна назвати ортогональним, якщо окреслене співвідношенням сімейство вигляду $\{\psi_{jk}\}$ являє собою ортонормований базис раніше згаданого простору $L^2(R)$, тобто:

$$\langle \psi_{jk}, \psi_{lm} \rangle = \delta_{jl} \delta_{km}$$

, тоді кожна функція $f \in L^2(R)$ може бути представлена у наступному вигляді:

$$f(t) = \sum_{j,k=-\infty}^{\infty} c_{jk} \psi_{jk}(t) \quad (1.8)$$

рівномірна збіжність котрого у $L^2(R)$ означає, що

$$\lim_{M_1, N_1, M_2, N_2 \rightarrow \infty} \left\| f - \sum_{-M_2}^{N_2} \sum_{-M_1}^{N_1} c_{jk} \psi_{jk} \right\|_2 = 0$$

Найпростішим прикладом ортогонального вейвлету є HAAR-вейвлет, який має таку назву на честь угорського математика Альфреда Хаара (Haar), який запропонував цей вид вейвлетів і який визначається співвідношенням [4]:

$$\psi^H(t) = \begin{cases} 1, & 0 \leq t < 1/2, \\ -1, & 1/2 \leq t < 1, \\ 0, & t < 0, \quad t \geq 1. \end{cases} \quad (1.9)$$

Можна легко побачити, що будь-які дві функції ψ_{jk}^H, ψ_{lm}^H , отримані з цього вейвлету за формулою (1.7) за допомогою масштабних перетворень $1/2^j, 1/2^l$ та зсувів $k/2^j, m/2^l$, є ортогональними і мають одиничну норму.

Сконструємо базис функціонального простору $L^2(R)$ за допомогою безперервних масштабних перетворень та переносів вейвлету $\psi(t)$ із довільними значеннями базисних параметрів — масштабного коефіцієнту a та параметру зсуву b :

$$\psi_{ab}(t) = |a|^{-\frac{1}{2}} \psi\left(\frac{t-b}{a}\right), \quad a, b \in R, \quad \psi \in L^2(R) \quad (1.10)$$

На основі формули (1.10) запишемо інтегральне вейвлет-перетворення:

$$[W_\psi f](a, b) = |a|^{-\frac{1}{2}} \int_{-\infty}^{\infty} f(t) \psi^*\left(\frac{t-b}{a}\right) dt = \int_{-\infty}^{\infty} f(t) \psi_{ab}^*(t) dt \quad (1.11)$$

Якщо проводити подальшу аналогію з перетворенням Фур'є, коефіцієнти $c_{jk} = \langle f, \psi_{jk} \rangle$ розкладання (1.8) функції f в ряд по вейвлетах можна визначити за допомогою інтегрального вейвлет-перетворення [5]:

$$c_{jk} = [W_\psi f]\left(\frac{1}{2^j}, \frac{k}{2^j}\right) \quad (1.12)$$

Надалі іноді замість $[W_\psi f](a, b)$ для коефіцієнтів (амплітуд) вейвлет-перетворення будуть використовуватись позначення $W(a, b)$ або $W_\psi f$, або $W[f]$.

Отже, кожна функція з $L^2(R)$ може бути отримана суперпозицією (лінійною комбінацією складових) масштабних перетворень та зсувів базисного вейвлету,

тобто є композицією "вейвлетових хвиль" (із коефіцієнтами, що є залежними від номеру хвилі (частоти або масштабу) і ще від параметру зсуву (часу)) [6].

При застосуванні вейвлетів для аналізу сигналів неперервне вейвлет-перетворення (1.11) є більш зручним; його деяка надлишковість, яка пов'язана із безперервною зміною масштабного коефіцієнту a і параметру зсуву b , в даному випадку стає позитивною якістю, так як дозволяє більш повно та чітко уявити і проаналізувати інформацію, яка міститься у даних.

1.1.3 Зворотнє вейвлет-перетворення

Синусоїдальна хвиля формує ортонормований базис функціонального простору $L^2(0, 2\pi)$, і зі зворотним перетворенням Фур'є проблем не виникає. У той же час, ортонормованість базисів простору $L^2(R)$, побудованих на основі вейвлетів, визначається і вибором базисного вейвлета, і способом побудови базису (значення базисних параметрів a, b).

Звісно, вейвлет може вважатись базисною функцією $L^2(R)$ тільки в тому випадку, коли побудований з його допомогою базис ортонормований та зворотнє перетворення існує. Однак суворі докази повноти та ортогональності складні та громіздкі, приклади цього можна бачити в [7], де розробляється теорія вейвлет-перетворення. Крім того, для практичних цілей часто буває достатньо стійкості та "приблизної" ортогональності системи функцій розкладання, тобто достатньо, щоб вона була "майже базисом". Як правило, для аналізу сигналів використовуються такі "майже базисні" вейвлети.

Варто вказати зворотнє перетворення для тих двох випадків, що описані вище: для базису (1.7), що допускає розширення та зсуви $(1/2^j, k/2^j), j, k \in I$, та базису (1.10), побудованого при довільних значеннях $(a, b), a, b \in R$.

При базисних параметрах (a, b) , $a, b \in R$ зворотнє вейвлет-перетворення записується за допомогою того ж базису (1.10), що і пряме:

$$f(t) = C_{\psi}^{-1} \int \int [W_{\psi} f](a, b) \psi_{ab}(t) \frac{da db}{a^2}, \quad (1.13)$$

C_{ψ} — нормалізуючий коефіцієнт (аналогічний коефіцієнту $(2\pi)^{1/2}$, який нормалізує перетворення Фур'є):

$$C_{\psi} = \int_{-\infty}^{\infty} |\hat{\psi}(\omega)|^2 |\omega|^{-1} d\omega < \infty$$

(кришечкою зверху позначається Фур'є-образ).

Умова кінцевості константи C_{ψ} обмежує клас функцій $\psi(t) \in L^2(R)$, які можуть бути використані як базові вейвлети. Зокрема, очевидно, що образ Фур'є $\hat{\psi}$ повинен дорівнювати нулю на початку координат $\omega = 0$ і, отже, повинен дорівнювати нулю принаймні нульовий момент:

$$\int_{-\infty}^{\infty} \psi(t) dt = 0$$

Найчастіше у додатках достатньо розгляду лише позитивних частот, тобто $\omega > 0$; вейвлет, відповідно, повинен задовольняти умову:

$$C_{\psi} = 2 \int_0^{\infty} |\hat{\psi}(\omega)|^2 \omega^{-1} d\omega = 2 \int_0^{\infty} |\hat{\psi}(-\omega)|^2 \omega^{-1} d\omega < \infty$$

У разі дискретного вейвлет-перетворення стійкий базис визначається в такий спосіб. Функція $\psi \in L^2(R)$ називається R-функцією, якщо базис $\{\psi_{jk}\}$ який

визначений виразом (1.7), є базисом Риса (Riesz) у тому сенсі, що існують дві константи A і B , $0 < A \leq B < \infty$, для яких співвідношення:

$$A\|c_{jk}\|_2^2 \leq \left\| \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} c_{jk} \psi_{jk} \right\|_2^2 \leq B\|c_{jk}\|_2^2$$

виконується за будь-якої (обмеженої, двічі квадратично сумованої) послідовності $\{c_{jk}\}$:

$$\|c_{jk}\|_2^2 = \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} |c_{jk}|^2 < \infty$$

Для будь-якої R-функції існує базис $\{\psi^{jk}\}$ — "двійник" базису $\{\psi_{jk}\}$ (у тому сенсі, що $\langle \psi_{jk}, \psi^{lm} \rangle = \delta_{jl} \delta_{km}$) за допомогою якого можна побудувати реконструкційну формулу [8]:

$$f(t) = \sum_{j,k=-\infty}^{\infty} \langle f, \psi_{jk} \rangle \psi^{jk}(t) \quad (1.14)$$

Якщо ψ — ортогональний вейвлет і $\{\psi_{jk}\}$ — ортонормований базис, то $\{\psi^{jk}\}$ та $\{\psi_{jk}\}$ співпадають і формула (1.14) є формулою зворотного перетворення. Якщо ψ — не ортогональний вейвлет, але є двомісним або парним R-вейвлетом (dyadic wavelet), він має двійника $\tilde{\psi}$, за допомогою якого двійник сімейства $\{\psi_{jk}\}$ будується подібно до базису (1.7):

$$\psi^{jk}(t) = \tilde{\psi}_{jk}(t) = 2^{j/2} \tilde{\psi}(2^j t - k), \quad j, k \in I \quad (1.15)$$

У загальному випадку реконструкційна формула (1.14) навіть не обов'язково є вейвлет-рядом у тому ж сенсі, що $\tilde{\psi}$ не є вейвлетом і $\{\psi^{jk}\}$ може не мати базису-двійника, побудованого по типу (1.10) [8].

1.2 Частотно-часова локалізація

Перетворення і ряди Фур'є є прекрасними математичними апаратами для фізичної інтерпретації процесів при процесі аналізу сигналів, що їх представляють. Але є деякі випадки, коли вони виявляються недоречними або недостатньо ефективними в ході виконання конкретної задачі [9].

Реальний сигнал завжди (або, як правило) належить простору $L^2(R)$. Перетворення Фур'є сигналу $f(t)$ з кінцевою енергією, визначеною нормою $\|f\|_2$, являє собою спектр цього сигналу:

$$\hat{f}(\omega) = \int_{-\infty}^{\infty} f(t) \exp(-i\omega t) dt$$

У деяких випадках така інтерпретація з використанням даної формули буває занадто складною. Таким чином, щоб отримати інформацію на обраній частоті, необхідно мати і минулу, і часову майбутню інформацію; окрім цього, дана формула не враховує, що частота може змінитись з плином часу. Перетворення Фур'є не може розпізнати сигнал, що є сумою двох синусоїд з деякими різними частотами, від сигналу, який є складеним з цих самих синусоїд, що активуються послідовно одна за одною [10].

Крім того, відомо, що частота сигналу обернено пропорційна його тривалості [9]. Тому для отримання інформації високої частоти з хорошою

точністю важливо витягувати її з невеличких інтервалів часу, а не з усього сигналу разом; і навпаки — інформацію низької частоти слід витягувати з великих інтервалів часу сигналу.

Деякі з вищезгаданих труднощів уходять під час використання віконного перетворення Фур'є. Однак базова функція, що нескінченно коливається, (синусоїдальна хвиля) не дозволяє на виході отримати добре локалізовану інформацію. Одним з елементів базису вейвлет-перетворення є якісно локалізована функція, що швидко прагне нуля на невеликому інтервалі, це дозволяє проводити "локалізований спектральний аналіз". Таким чином, вейвлет-перетворення саме по собі має рухоме частотно-часове вікно, звужене на малих інтервалах і розширене на великих.

Оскільки і сам вейвлет ψ , і його Фур'є-образ $\tilde{\psi}$ досить швидко спадають, вони можуть бути використані як "віконні функції" з "центром" та "шириною", що визначаються наступним чином [11].

Для нетривіальної віконної функції $z(t) \in L^2(R)$ (необхідно, щоб функція $tz(t)$ також належала $L^2(R)$), її центр $\langle t \rangle$ та радіус Δ_z визначаються формулами:

$$\langle t \rangle = \frac{1}{\|z\|_2^2} \int_{-\infty}^{\infty} t |z(t)|^2 dt,$$

$$\Delta_z = \frac{1}{\|z\|_2} \left[\int_{-\infty}^{\infty} (t - \langle t \rangle)^2 |z(t)|^2 dt \right]^{1/2},$$

при цьому ширина віконної функції дорівнює $2\Delta_z$.

Нехай $\langle t \rangle$, Δ_ψ , $\langle \omega \rangle$, $\Delta_{\tilde{\psi}}$ — визначені за цими формулами центри та радіуси вейвлету ψ і його Фур'є-образу $\tilde{\psi}$ відповідно. Тоді інтегральне вейвлет-перетворення (1.11) обмежене "тимчасовим вікном" [12]:

$$[win_t] = [b + a\langle t \rangle - 2a\Delta_\psi, b + a\langle t \rangle + 2a\Delta_\psi] \quad (1.16)$$

Тобто має місце часова локалізація з центром вікна у $b + a\langle t \rangle$ та шириною вікна $4a\Delta_\Psi$.

Введемо функцію $\eta(\omega) = \tilde{\Psi}(\omega + \langle \omega \rangle)$, яка також є віконною функцією з центром у нулі та радіусом $\Delta_{\tilde{\Psi}}$. Скориставшись рівністю $\langle f, g \rangle = \langle \hat{f}, \hat{g} \rangle / 2\pi$, можна записати інтегральне вейвлет-перетворення (1.11) для Фур'є-образу $\hat{f}$ у вигляді:

$$W(a, b) = |a|^{1/2} \int_{-\infty}^{\infty} \hat{f}(\omega) \exp(ib\omega) \eta^* \left(a \left[\omega - \frac{\langle \omega \rangle}{a} \right] \right) d\omega \quad (1.17)$$

Якщо відволіктися від зсуву по фазі, можна помітити, що перетворення (1.17) дає локалізовану інформацію про спектр $\hat{f}(\omega)$ сигналу $f(t)$ з "частотним вікном"

$$[win_\omega] = \left[\frac{\langle \omega \rangle}{a} - \frac{1}{a} \Delta_{\tilde{\Psi}}, \frac{\langle \omega \rangle}{a} + \frac{1}{a} \Delta_{\tilde{\Psi}} \right] \quad (1.18)$$

Частотна локалізація відбувається навколо центру даного вікна в $\frac{\langle \omega \rangle}{a}$ із шириною, яка дорівнює $\frac{2\Delta_{\tilde{\Psi}}}{a}$.

Варто зауважити, що центральна частота відноситься до ширини вікна, як:

$$\frac{\langle \omega \rangle}{a} \left(\frac{2\Delta_{\tilde{\Psi}}}{a} \right)^{-1} = \frac{\langle \omega \rangle}{2\Delta_{\tilde{\Psi}}}$$

, і не залежить від місцезнаходження центральної частоти, а частотно-часове вікно $[win_t] \cdot [win_\omega]$, яке має площу $4\Delta_\Psi \Delta_{\tilde{\Psi}}$, стає вузьким за високої центральної частоти $\frac{\langle \omega \rangle}{a}$ і розширюється за низької (рис. 1а).

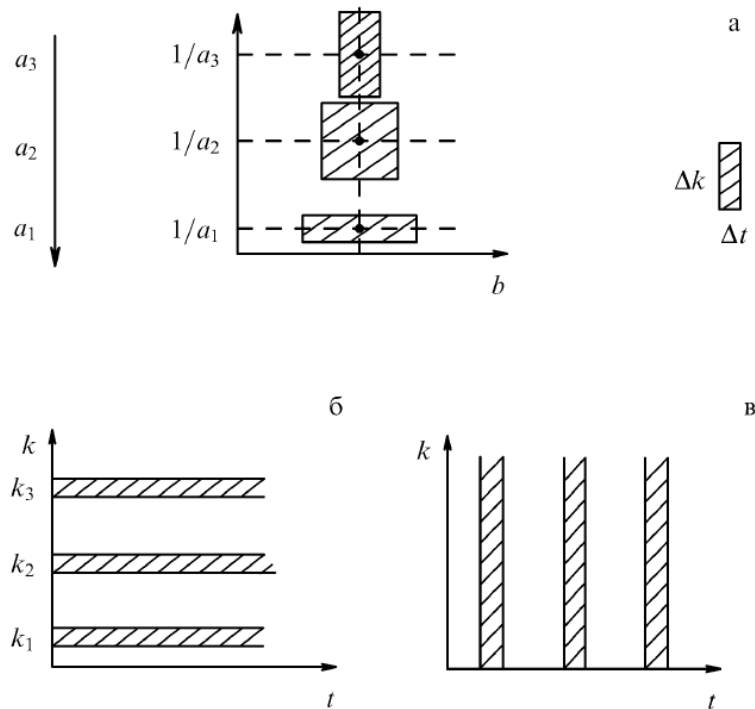


Рисунок 1.2 – Схематичне зображення частотно-часової локалізації перетворень з використанням різних аналізаторів: (а) вейвлети; (б) гармоніки Фур'є; (в) функції Шеннона.

На рис. 1.2 схематично зображено локалізацію перетворень у частотно-часовому просторі з різними функціями-аналізаторами: перетворення Фур'є (рис. 1.2 б) та перетворення Шеннона (Shannon), функцією для проведення аналізу якого є функція Дірака (рис. 1.2 в).

З рис. 1.2 можна побачити, що перетворення Фур'є може локалізувати частоту, але не враховує дані часу; перетворення Шеннона не локалізується частотно; вейвлет-перетворення має рухоме вікно, що є локалізованим навколо вибраного моменту в часі та розширюється зі зростанням інтервалу, що є бажаним результатом для отримання якісної спектральної інформації. Функціями базису вищеперерахованих перетворень є відповідно синусоїдальна хвиля, функція

Дірака і солітон подібний вейвлет, який непогано локалізує і масштаб, і момент часу.

Для демонстрації переваг вейвлет-перетворення як методу локалізованого спектрального аналізу проведемо порівняння вейвлет-перетворення (1.11) з часто використовуваним для аналізу сигналів віконним або короткочасним перетворенням Фур'є [12]:

$$F(\omega, b) = \int f(t)z(t - b) \exp(i\omega t) dt$$

Тобто перетворенням сигналу, помноженого попередньо на віконну функцію z . Таким чином, $F(\omega, b)$ є розкладання сигналу за сімейством функцій $z(t - b) \exp(i\omega t)$, утвореному з функції $z(t)$ з використанням переносів b у часі та ω по частоті, в той час як результатом вейвлет-перетворення $W(a, b)$ є розкладання сигналу за сімейством $\psi((t - b)/a)$, утвореному з єдиної функції $\psi(t)$ за допомогою переносів b у часі та розтягувань a також у часі. Вейвлет-перетворення є, так би мовити, нескінченним банком для віконних перетворень Фур'є з вікнами, що різняться відносно кожної іншої частоти.

Таким чином, базисні функції віконного перетворення Фур'є однаково дійсні на часовому і частотному просторах $(z(t), \hat{z}(\omega))$ для всіх точок площини перетворення, в той час як у базисних функціях вейвлет-перетворень зі зменшенням масштабу a зменшується і дійсність у часовому просторі $(\psi(t/a))$ і зі збільшенням a зростає дійсність функції у частотному просторі $(\hat{\psi}(a\omega))$. Ця властивість вейвлет-перетворень дає їм відчутну перевагу в задачах аналізу сигналів через те, що швидкі зміни в сигналах добре локалізовані, а для отримання повільно змінних характеристик буде достатньо низькочастотного дозволу. Вейвлет-перетворення, яке володіє рухомим частотно-часовим вікном, може якісно виявляти і низькочастотні, і високочастотні характеристики сигналів.

Перетворення Фур'є ряду даних із рівномірною дискретизацією за часом Δt не може досягти дискретизації за частотою більшою, ніж $\Delta\omega = \Delta t/2$ (частота Найквіста, що демонструє окремий випадок принципу невизначеності між часовою та частотною локалізаціями). Аналогічне обмеження для вейвлет-перетворення можна висловити співвідношенням $\Delta t \Delta\omega \geq (4\pi)^{-1}$.

У підсумку можна виявити, що описані в даному розділі переваги вейвлетів виявляться надзвичайно корисними не тільки для аналізу складних сигналів, а й під час вирішення рівнянь, що описують процеси з взаємодіями збурень різних масштабів [13].

1.3 Haar-вейвлети

У чисельному аналізі дискретне вейвлет-перетворення (DWT або ДВП) - це будь-яке вейвлет-перетворення, де використані вейвлети дискретно відібрані. Як і у випадку з іншими перетвореннями, його головною перевагою над перетвореннями Фур'є є роздільна здатність в часі: це дозволяє фіксувати як частоту сигналу, так і інформацію про його масштаб і місцезнаходження.

Перше DWT було винайдено угорським математиком Альфредом Хааром [9]. Для вхідних даних, представлених списком з 2^n чисел, вейвлет-перетворення Хаара можна вважати простим об'єднанням вхідних значень, збереженням різниці та передачею суми. Цей процес повторюється рекурсивно, об'єднуючи суми для отримання наступної: в результаті отримуємо $2^n - 1$ різниць і одну остаточну суму. DWT Хаара показує собою ідеальні властивості вейвлетів взагалі. По-перше, він може бути виконаний за $O(n)$ операцій (що означає невелику складність алгоритму); по-друге, він видає не тільки частотний вміст вхідного сигналу, при розгляданні його в різних масштабах, а й часовий вміст, тобто час, коли ці частоти

виникають. Разом ці властивості утворюють швидке вейвлет-перетворення (ШВП), що є альтернативою традиційному швидкому перетворенню Фур'є (ШПФ). Дискретне вейвлет-перетворення має величезну кількість застосувань в науковій, технічній, математичній галузях. Зокрема, воно використовується для кодування сигналів, для представлення дискретного сигналу в більш надлишковій формі, часто як передумова для стиснення даних [13].

Для формування вейвлету використовується метод перетворення Хаара [14], оскільки він є найпростішим з усіх методів вейвлет-перетворення і може ефективно слугувати нашим інтересам. У методі вейвлет-перетворення Хаара низькочастотна фільтрація здійснюється шляхом усереднення двох сусідніх значень пікселів, тоді як для високочастотної фільтрації обчислюється різниця між двома сусідніми значеннями пікселів. Вейвлет Хаара застосовує пару фільтрів низьких і високих частот до декомпозиції зображення спочатку в стовпчиках, а потім у рядках зображення незалежно один від одного. В результаті на виході вейвлету Хаара першого рівня утворюються чотири підсмуги. Чотири підсмуги - це LL1, HL1, LH1 та HH1. Низькочастотний піддіапазон LL1 може бути далі розкладений на чотири піддіапазони LL2, HL2, LH2 і HH2 на наступному більш грубому рівні. LLi – це зменшена роздільна здатність, що відповідає низькочастотній частині зображення. Інші три підсмуги HLi, LHі та Hhi – це високочастотні частини у вертикальному, горизонтальному та діагональному напрямках відповідно [14].

Перетворення Хаара перехресно множить функцію на вейвлет Хаара з різними зсувами і розтяжками, подібно до того, як перетворення Фур'є перехресно множить функцію на синусоїду з двома фазами і багатьма розтяжками. Приклад матриці Хаара 4x4 показано нижче [15]:

$$H_4 = \frac{1}{\sqrt{4}} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$$

РОЗДІЛ 2

ПРАКТИЧНА РЕАЛІЗАЦІЯ ВЕЙВЛЕТ-ПЕРЕТВОРЕННЯ

Програмне забезпечення Wavelet Toolbox містить велику кількість вейвлетів, які можна використовувати як для безперервного, так і для дискретного аналізу [21].

Для дискретного аналізу приклади включають ортогональні вейвлети та Б-сплайнові біортогональні вейвлети. Для безперервного аналізу, програмне забезпечення Wavelet Toolbox включає вейвлети Морле, Мейєра, Хаар, Добеші та інші.

Вибір потрібного для конкретної задачі вейвлета спирається на аналіз характерних рис вхідного сигналу або зображення та напрямком застосування.

Сімейства вейвлетів різняться з точки зору кількох важливих властивостей. Наприклад:

- Підтримка вейвлета за часом, частотою та швидкістю загасання.
- Симетрія або антисиметрія вейвлета. Супутні фільтри безпомилкової реконструкції мають лінійну фазу.
- Кількість моментів зникнення. Вейвлети зі збільшеною кількістю моментів зникнення призводять до розрідженого представлення для великої кількості сигналів і зображень..
- Регулярність вейвлета. Більш плавні вейвлети забезпечують чіткішу частотну роздільну здатність. Також алгоритми повторення побудови вейвлетів відпрацьовують швидше.
- Наявність функції масштабування, ϕ .

Для безперервного аналізу програмне забезпечення Wavelet Toolbox виконує аналітичний аналіз на основі вейвлетів.

Таблиця 2.1. Перелік родин велетів бібліотеки [16]

Коротке ім'я родини Wavelet	Ім'я родини Wavelet
'haar'	Haar wavelet
'db'	Daubechies wavelets
'sym'	Symlets
'coif'	Coiflets
'bior'	Biorthogonal wavelets
'rbio'	Reverse biorthogonal wavelets
'meyr'	Meyer wavelet
'dmey'	Discrete approximation of Meyer wavelet
'gaus'	Gaussian wavelets
'mexh'	Mexican hat wavelet (також відомий як Ricker wavelet)
'morl'	Morlet wavelet
'cgau'	Complex Gaussian wavelets
'shan'	Shannon wavelets
'fbsp'	Frequency B-Spline wavelets
'cmor'	Complex Morlet wavelets
'fk'	Fejer-Korovkin wavelets

2.1 Haar-вейвлети в MATLAB

У Wavelet Toolbox Matlab можна використовувати велику кількість типів вейвлетів. Для отримання довідкової інформації про будь-який тип вейвлетів можна подати команду `waveinfo` з аргументом - ім'ям типу вейвлетів [22].

Наприклад:

```
waveinfo('haar')
```

...

General characteristics: Compactly supported wavelet, the oldest and the simplest wavelet.

scaling function $\phi = 1$ on $[0 \ 1]$ and 0 otherwise.

wavelet function $\psi = 1$ on $[0 \ 0.5)$, $= -1$ on $[0.5 \ 1]$ and 0 otherwise.

Family	Haar
Short name	haar
Examples	haar is the same as db1
Orthogonal	yes
Biorthogonal	yes
Compact support	yes
DWT	possible
CWT	possible

...

Функція `wavefun('haar',n)` повертає вектор значень масштабовальної $\varphi(x)$ та вейвлет функції Хаара $\psi(x)$ у 2^{n+1} точках відрізка $[0, 1 + \frac{1}{2^n}]$, включно з кінцевими, наприклад:

```
[phi,psi,xval]=wavefun('haar',2)
phi = 0    1.0000    1.0000    1.0000    1.0000    0
psi = 0    1.0000    1.0000   -1.0000   -1.0000    0
xval = 0    0.2500    0.5000    0.7500    1.0000    1.2500
```

Для наочності, далі буде побудовано графіки функцій $\varphi(x)$ та $\psi(x)$.

```
[phi,psi,xval]=wavefun('haar',8);
xaxis=zeros(size(xval));
subplot(121),plot(xval,phi,'r',xval,xaxis,'k','LineWidth',3);
axis([min(xval) max(xval) -1.5 1.5]);
subplot(122),plot(xval,psi,'b',xval,xaxis,'k','LineWidth',3);
axis([min(xval) max(xval) -1.5 1.5]);
```

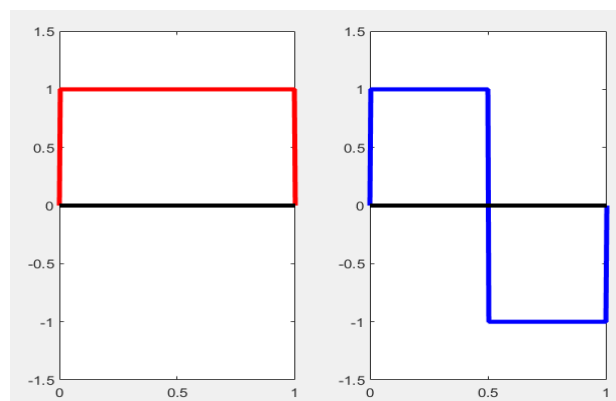


Рисунок 2.1 – Зображення функцій $\varphi(x)$ та $\psi(x)$ в середі MATLAB.

Один крок вейвлет-перетворення Хаара можна виконати командою `dwt`, котра має синтаксис

```
[cA,cD] = dwt(X,'haar');
```

Функція `dwt` обчислює апроксимуючі коефіцієнти `cA` (у розділі 1.3 – LL) і деталізуючі коефіцієнти `cD` (HL, LH, HH) вектора `X`. Другий аргумент є рядком імені типу використовуваних вейвлетів.

Функція `idwt` виконує однорівневе обернене дискретне одновимірне ВП. Команду `idwt` можна вживати в такій формі:

```
s = idwt(cA,cD,'haar');
```

Вона за відомими апроксимуючими коефіцієнтами `cA` і деталізуючими коефіцієнтами `cD` першого рівня обчислює значення сигналу `s`, використовуючи ім'я вейвлета. Далі наведений приклад використання даної функції.

```
% розклад функції sin(4*pi*x) по вейвлетах Хаара
close all; clear all;
x=0:0.05:1+0.05; y=sin(4*pi.*x);
subplot(211); plot(x,y,'LineWidth',2);
title('Початковий сигнал'); grid on;
[ca,cd]=dwt(y,'haar');
xval=0:0.1:1;
subplot(223); plot(xval,ca,'LineWidth',2);
title('Коефіцієнти усереднення'); grid on;
subplot(224); plot(xval,cd,'LineWidth',2);
title('Коефіцієнти деталізації'); grid on;
% Зворотне ДВП
s=idwt(ca,cd,'haar');
err=norm(y-s)
```

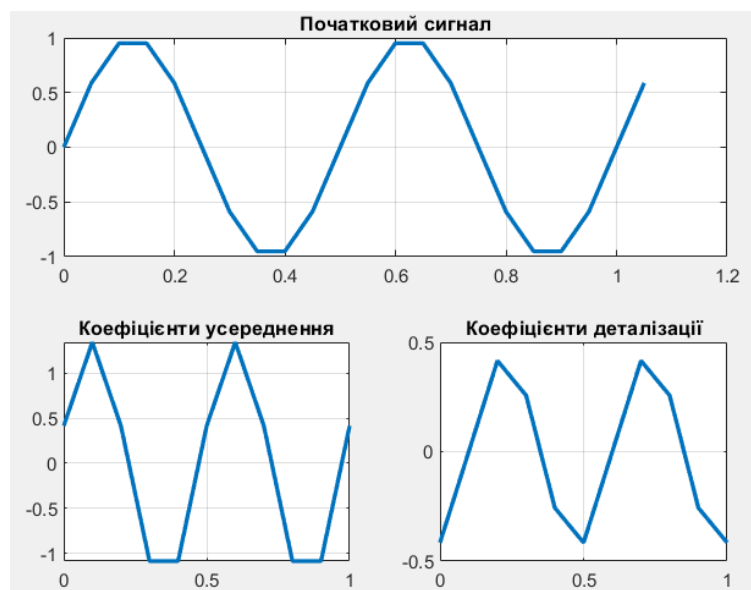


Рисунок 2.2 – Графічне зображення даної функції та її коефіцієнтів усереднення та деталізації.

Норма різниці вихідного вектора y і відновленого s дуже мала:

$$\text{err} = 6.5773\text{e-}16$$

Одновимірне багаторівневе ДВП у Matlab виконує функція `wavedec(f, k, 'haar')`. Вона приймає вектор f , кількість кроків ВП k (у термінології MATLAB k - рівень розкладання) та ім'я типу вейвлетів (у даному випадку 'haar'). Повертає пару векторів C і L .

$$[C, L] = \text{wavedec}(f, k, 'haar');$$

Рисунок 2.3 пояснює сенс параметрів, що повертаються після обчислення функції, що розглядається.

Усі коефіцієнти одновимірного ВП повертаються у вигляді вектора C . При цьому довжина усереднювального вектора A , довжини одновимірних деталізуючих векторів D_i і довжина вихідного вектора X (збігається з довжиною вектора C) повертаються у вигляді вектора L [23].

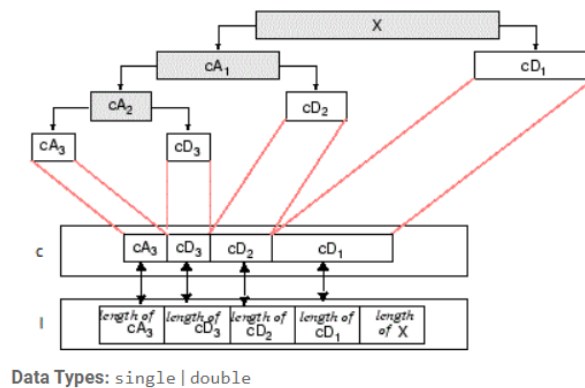


Рисунок 2.3 – Вектор X, що повертається як вектор натуральних чисел. Вектор містить кількість коефіцієнтів за рівнем (C) та довжину вихідного сигналу (L).

Далі слід продемонструвати використання функції `wavedec`:

```
close all; clear all;
f=[8 4 6 8 9 7 2 4 1 3 5 6 4 2 2 4];% довжина f = ступінь
двійки
N=length(f);
f=f./sqrt(N); % перетворюємо f до нормованих psi
x=-0.1:0.005:1.1;
[c,L] = wavedec(f,2,'haar');% заміняємо другий аргумент від
0 до 4
a=c(1:L(1));
ym=wcalc(x,a,[]); % усереднювальна функція
%figure,plot(x,ym,'k-','LineWidth',2); % 1
%axis([min(x) max(x) min(ym)-0.5 max(ym)+0.5]); grid on;
d=c(L(1)+1:L(end));
y=wcalc(x,a,d); % початкова функція
%figure,plot(x,y,'k-','LineWidth',2); % 2
```

```
%axis([min(x) max(x) min(y)-0.5 max(y)+0.5]); grid on;
figure,plot(x,y,'k-',x,ym,'LineWidth',2); % обидва графіки
grid on; set(gcf,'Color',[1 1 1]);
```

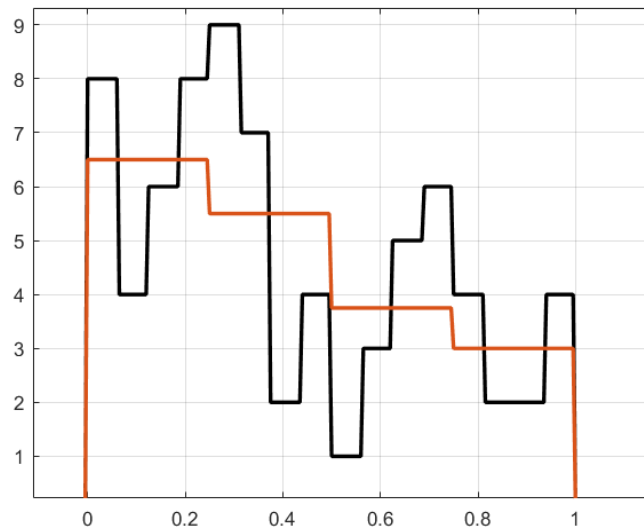


Рисунок 2.4 – Одновимірне багаторівневе ДВП, виконане за допомогою функції `wavedec`.

Функція `waverec` - багаторівневе відновлення одновимірного сигналу. Вона здійснює відновлення сигналу `s`, використовуючи отримані раніше вектори багаторівневого розкладання `[C,L]`. Під час відновлення вказується ім'я використовуваного вейвлета, у нашому випадку `'haar'`. Один із варіантів використання цієї функції наступний [21]:

```
s = waverec(C,L,'haar');
```

Наприклад, розглянемо варіант використання функцій `wavedec` та `waverec` разом:

```
close all; clear all;
load sumsin; % тестовий сигнал довжиною 1000 точок
s=sumsin(1:400);
```

```

[c, l]=wavedec(s, 3, 'haar');
subplot(211);    plot(s, 'LineWidth', 1);    title('Початковий
сигнал');
grid on;
subplot 212; plot(c); title('Вейвлет коефіцієнти'); grid
on;
sr=waverec(c, l, 'haar'); % реконструкція сигналу
err=norm(s-sr) % норма різниці сигналів

```

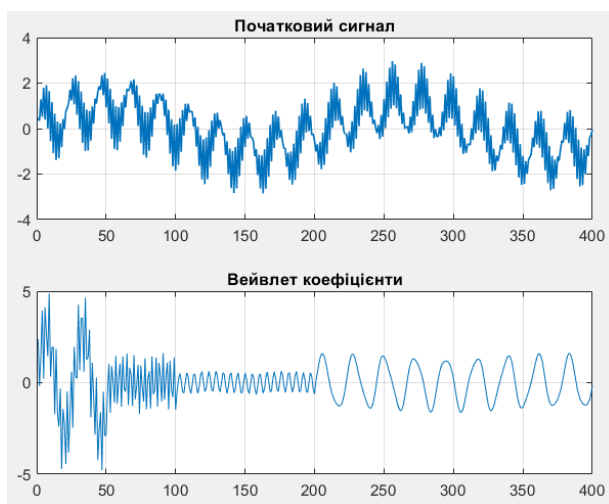


Рисунок 2.5 – Розкладання тестового сигналу за вейвлетами Хаара та його реконструкція.

По нижній частині рисунку 2.5 добре видно, що довжина вихідного сигналу така сама, як і у вектора коефіцієнтів, який складено з компонент $cA3$, $cD3$, $cD2$, $cD1$. На нижньому графіку добре помітні перша компонента вектора C довжини 50 (усереднювальна частина) і три деталізуючі ділянки. Останні два рядки коду сценарію показують, що вихідний і відновлений сигнали практично збігаються. Отримано помилку $err = 8.8545e-15$.

Команда `appcoef` обчислює коефіцієнти наближення cA одновимірного сигналу рівня N , використовуючи отримані раніше вектори багаторівневого

розкладання $[C, L]$. При цьому рівень N має бути не вищим, ніж рівень N_0 отриманого розкладання $[C, L]$. Наприклад, якщо вектор C має структуру $[A(N_0), D(N_0), \dots, D(N), \dots, D(1)]$, то повинна мати місце така нерівність: $0 \leq N \leq N_0$. Зазвичай команда застосовується в такій формі [21]:

```
A = appcoef(C, L, 'haar', N);
```

Команда `detcoef` знаходить деталізуючі коефіцієнти одновимірного сигналу рівня N , використовуючи отримані раніше вектори багаторівневого розкладання $[C, L]$. Рівень N має бути не вищим, ніж рівень отриманого розкладання $[C, L]$, тобто $0 \leq N \leq N_0$. Її використання показано далі на прикладі.

```
close all; clear all;
load leleccum; % тестовий сигнал
s=leleccum(1:4000);
subplot(411); plot(s); title('Початковий сигнал s');
[c,l]=wavedec(s,3,'haar');
subplot 412; plot(c); title('Вектор коефіцієнтів c');
ca3=appcoef(c,l,'haar',3);
subplot(425); plot(ca3); title('Вектор cA3');
[cd1,cd2,cd3] = detcoef(c,l,[1 2 3]);
subplot(426); plot(cd3); title('Вектор cD3');
subplot(427); plot(cd2); title('Вектор cD2');
subplot(428); plot(cd1); title('Вектор cD1');
```

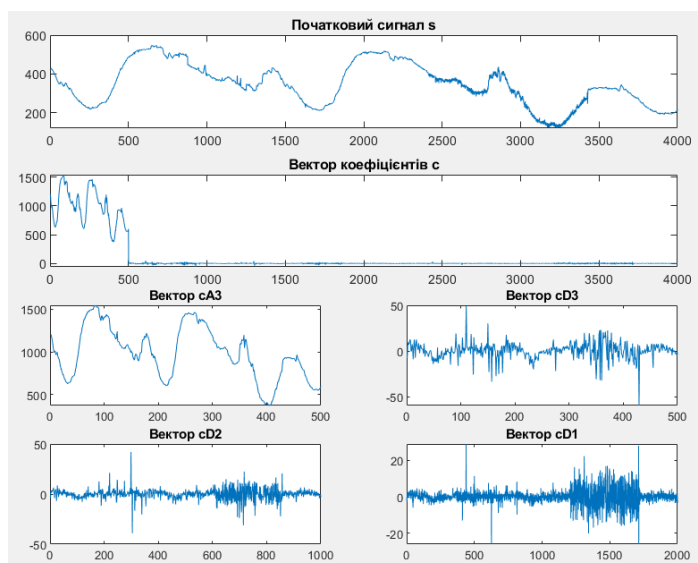


Рисунок 2.6 – Тестовий сигнал розкладаємо до рівня 3, а потім вилучаємо коефіцієнти апроксимації та деталізації рівнів 1, 2, 3.

Варто звернути увагу на те, що команда `appcoef(c,1,'haar',0)` повертає на виході початковий сигнал:

```
ca0=appcoef(c,1,'haar',0);
figure; plot(ca0-10,'r'); % зміщення графіку вниз
hold on; plot(s); hold off;
```

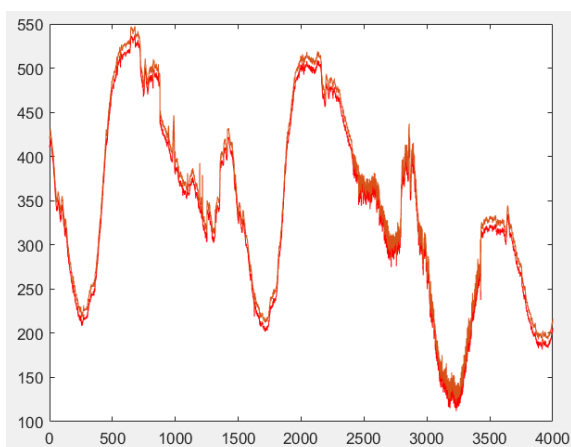


Рисунок 2.7 – Приклад роботи функції `appcoef` на тестовому сигналі.

Функція `upwlev` виконує однорівневу реконструкцію одновимірного ВП. Якщо задані вектори вейвлет розкладання $[C, L]$ рівня розкладання N_0 , то команда `upwlev` знаходить вектори вейвлет розкладання $[NC, NL]$ рівня $N_0 - 1$, тобто на один рівень вище. Вилучаються також останні коефіцієнти апроксимації cA рівня N_0 .

```
[NC, NL, cA] = upwlev(C, L, 'haar');
```

Для наочності, у продовженні попереднього сценарію варто додати описані команди:

```
[c2, l2] = wavedec(s, 2, 'haar');
[NC, NL, CA] = upwlev(c, l, 'haar');
figure;
subplot(2,1,1); plot(c2); title('Вектор коефіцієнтів c2');
subplot(2,1,2); plot(NC); title('Вектор NC=c2');
```

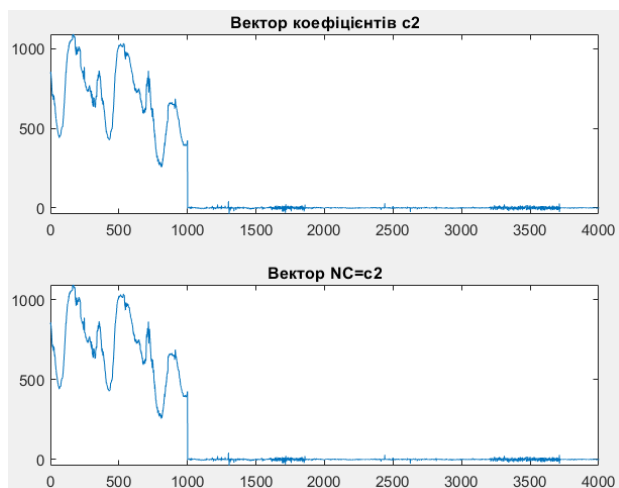


Рисунок 2.8 – Отримання вектору коефіцієнтів ДВП другого рівня за допомогою функцій `wavedec` та `upwlev`.

На рисунку 2.8 побудовано графік вектора коефіцієнтів ДВП 2-го рівня за допомогою команди `[c2,l2]=wavedec(s,2,'haar')`. Потім береться вектор коефіцієнтів `[c,l]`, отриманий раніше. Він представляє 3-й рівень ДВП, до якого застосовуємо функцію `[NC,NL,CA]=upwlev(C,L,'haar')` і отримуємо вектор коефіцієнтів 2-го рівня. Як можна побачити з рис. 2.8, графіки векторів `c2` і `NC` вийшли ідентичними.

```
close all; clear all;
load sumsin; % тестовий сигнал
s=sumsin(1:1000);
subplot(311); plot(s); title('Початковий сигнал s');
[c,l]=wavedec(s,3,'haar');
subplot 312; plot(c); title('Вектор декомпозиції 3-го
рівня');
[nc,nl]=upwlev(c,l,'haar');
subplot 313; plot(nc); title('Вектор декомпозиції 2-го
рівня');
set(gcf,'Color',[1 1 1]);
```

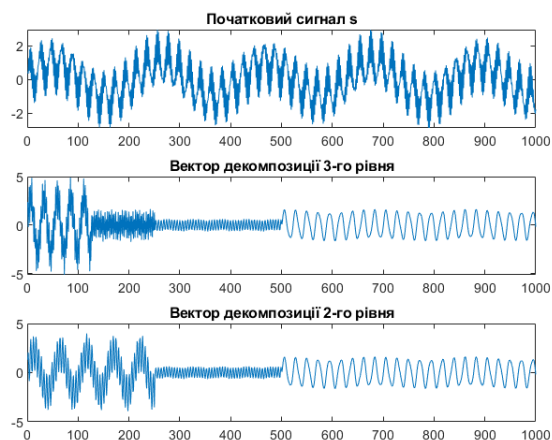


Рисунок 2.9 – Розклад тестового сигналу на вектори декомпозиції 3-го та 2-го рівня.

```
[c2,l2]=wavedec(s,2,'haar');
figure, plot(c2); hold on; plot(nc-0.2,'r'); hold off;
```

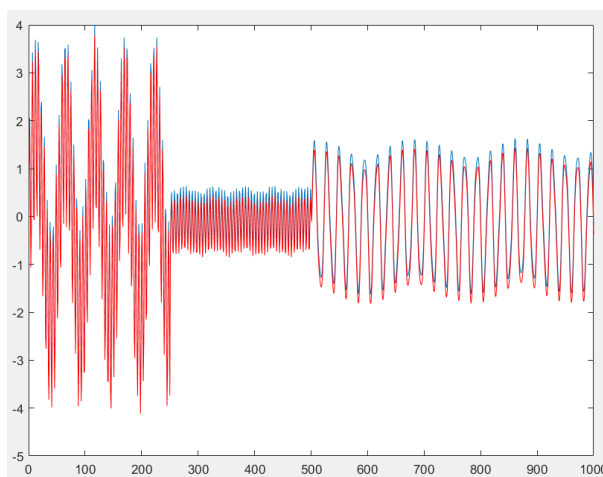


Рисунок 2.10 – Порівняння результатів `wavedec(s,2,'haar')` та однорівневої реконструкції `upwlev(c,l,'haar')`.

Функція `wrcoef` відновлює сигнал за одним усереднювальним вектором cA_N або деталізуючим вектором cD_N масиву вейвлет коефіцієнтів. У першому випадку всі деталізуючі коефіцієнти рівня N і нижче обнуляються і відновлюється «наближено» вихідна функція. У другому - обнуляється усереднювальний вектор і всі деталізуючі коефіцієнти нижче рівня N . Фактично значення вектора cA_N або cD_N «розтягуються» на всю довжину вихідного сигналу (збільшується кількість елементів вектора). Крім того, функція враховує нормувальні коефіцієнти.

Функція має такий синтаксис:

```
X=wrcoef('type',c,l,'haar',N);
```

Тут 'type' визначає тип гілки:

'type'='a' - відновлення з апроксимуючих коефіцієнтів;

'type'='d' - відновлення з деталізуючих коефіцієнтів.

Якщо N не вказане, то мається на увазі максимальний рівень.

```

close all; clear all;
s=[8 4 6 8 9 7 2 4 1 3 5 6 4 2 2 4]; % вектор сигналу
% Будуємо графік початкової кусочної функції
x=-1:0.01:17;
n=length(s);
y=0;
for i=1:n
y=y+s(i).*psiHaar(x,0,i-1);
end
plot(x,y,'k-','LineWidth',2); % графік (чорна ломана)
axis([min(x) max(x) min(y)-1 max(y)+1]); grid on;
hold on;
[c,l] = wavedec(s,2,'haar');%вейвлет декомпозиція до 2-го
рівня
c2=c(1:4)%чотири значення ділимо на 2 (два рази по кореню з
двох)
% Порівняння з роботою функції appcoef
ca2=appcoef(c,l,'haar',2);
ca2 % ті ж значення ca2=c2
% Наближена реконструкція сигналу
% Зі структури [c,l] з вектора апроксимації ca2
awr2 = wrcoef('a',c,l,'haar',2)
% графік наближення за коефіцієнтами awr2
yr=0;
for i=1:n
yr=yr+awr2(i).*psiHaar(x,0,i-1);
end
plot(x,yr,'b-','LineWidth',3); % синя

```

```

% графік наближень за коефіцієнтами awd2
awd2 = wrcoef('d',c,1,'haar',2);
yd=0;
for i=1:n
yd=yd+awd2(i).*psiHaar(x,0,i-1);
end
plot(x,yd,'r-','LineWidth',2); % червона
% сума наближень A и D
plot(x,yd+yr,'g-','LineWidth',2); % зелена

```

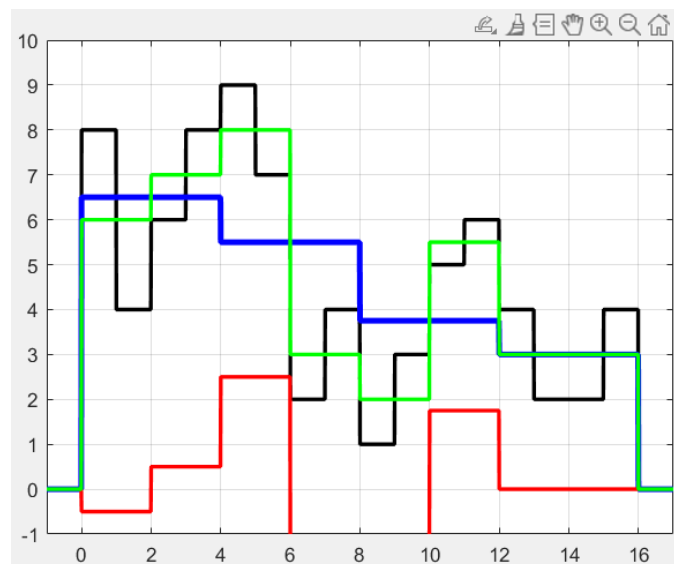


Рисунок 2.11 – На графіку наведено початкову кусково-постійну функцію (чорна крива) та її наближення (синій колір), побудоване за апроксимуючими вейвлет коефіцієнтами 2-го рівня. Червона ламана представляє наближення за коефіцієнтами cD_2 , а зелена - сума двох наближень (сума синьої і червоної кривих).

Слід зауважити, що довжини векторів $awr2$ і $awd2$ збігаються з довжиною вихідного вектора s .

Значення векторів:

```

c2 = 13.0000    11.0000    7.5000    6.0000
ca2 = 13.0000    11.0000    7.5000    6.0000
awr2 = 6.5000    6.5000    6.5000    6.5000    5.5000 . . .
      5.5000    5.5000    5.5000    3.7500    3.7500 . . .
      3.7500    3.7500    3.0000    3.0000    3.0000 . . .
      3.0000

```

Функція `upcoef` приймає вектор X і відновлює вектор Y з коефіцієнтів ДВП, застосовуючи N разів процедуру зворотного вейвлет перетворення. Вона має такий синтаксис [21]:

```
Y = upcoef('type',X,'haar',N);
```

Вектор X вважається деталізуючими коефіцієнтами, якщо `'type'='d'` і апроксимувальні коефіцієнти вважаються нульовими. Навпаки, якщо `'type'='a'`, то деталізуючі коефіцієнти вважаються нульовими, а вектор X розглядається як такий, що апроксимує.

```

close all; clear all;
x=-1:0.01:17;
ca2 = [13 11 7.5 6];
cd2 =[ -1 5 -3.6 0];
rc = upcoef('a',ca2,'haar',2);
rd = upcoef('d',cd2,'haar',2);
yr=0; yd=0;
for i=1:16
    yr=yr+rc(i).*psiHaar(x,0,i-1);
    yd=yd+rd(i).*psiHaar(x,0,i-1);
end
plot(x,yr,'b-',x,yd,'r-','LineWidth',3);
axis([min(x) max(x) min(yr)-3 max(yr)+1]); grid on;

```

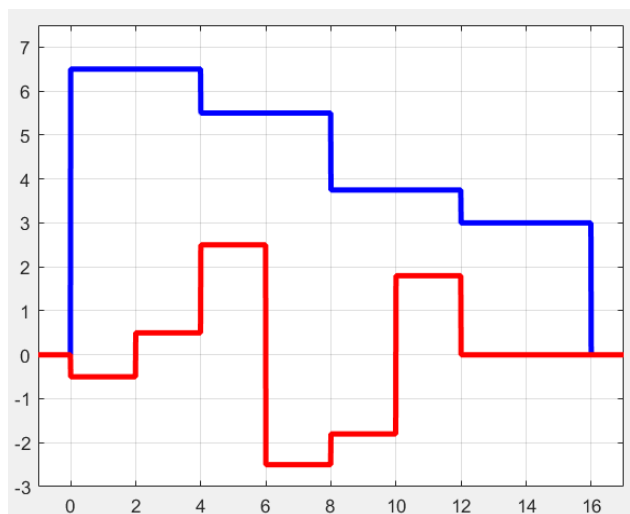


Рисунок 2.12 – Зображення вектору `sa2` (синя ломана) та наближення кривої (червона ломана) з попереднього сценарію.

У цьому прикладі було взято вектор `sa2` такий самий, як його було отримано в попередньому сценарії. Розглядаючи його як апроксимувальний вектор 2-го рівня, ми отримали наближення кривої такої самої, як у попередньому сценарії. Якщо в попередньому сценарії виконати команду `cd2=detcoef(c,1,2)`, то вона поверне вектор деталізуючих коефіцієнтів 2-го рівня

```
cd2 = -1.0000 5.0000 -3.5000 0
```

Їх ми використовували як вектор значень `cd2` під час отримання вектора `rd` у поточному сценарії. Використовуючи його, ми отримали наближення кривої таке саме, як у попередньому сценарії.

У всіх функцій, що були перелічені в цьому підрозділі, є й інші варіанти використання. Крім того, авторка скрізь використовувала тип вейвлетів 'haar', який можна змінити на інший [24].

2.2 Функції MATLAB для роботи із зображеннями

Один крок дискретного ВП виконує в Matlab функція `dwt2`. Вона викликається такою командою [25]:

```
[cA, cH, cV, cD] = dwt2(f, 'haar');
```

, де `cA` (LL) – матриця наближення (складається з коефіцієнтів апроксимації), `cH, cV, cD` (LH, HL, HH) – матриці, які складаються з горизонтальних, вертикальних та діагональних коефіцієнтів деталізації відповідно. Аргумент `f` - це зображення, що піддається ВП. Другий аргумент вказує ім'я типу вейвлетів. Повертає функція чотири матриці коефіцієнтів ВП: матрицю наближення `cA` і три матриці деталей `cH, cV, cD` (горизонтальну, вертикальну і діагональну).

```
close all; clear all;
f=im2double(imread('salad1.tif'));
[cA1, cH1, cV1, cD1] = dwt2(f, 'haar'); % один крок ВП
fw1=[cA1, cH1; ...
cV1, cD1];
figure, imshow(fw1, []); % ліве графічне вікно
figure; % праве графічне вікно
subplot(221), imshow(cA1, []);
subplot(222), imshow(cH1, []);
subplot(223), imshow(cV1, []);
subplot(224), imshow(cD1, []);
```

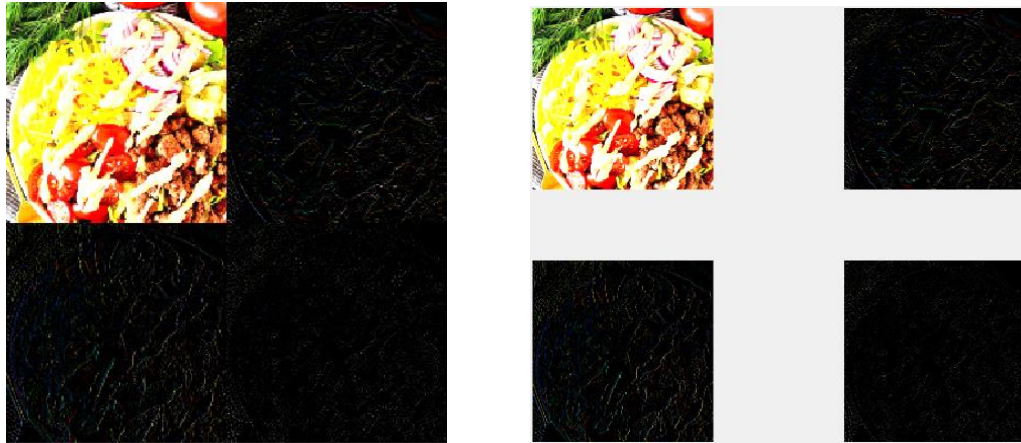


Рисунок 2.13 – Зображення, що зазнало одного кроку вейвлет перетворення.

```
close all; clear all;
f=im2double(imread('salad1.tif'));
[cA1,cH1,cV1,cD1] = dwt2(f,'haar'); % перший крок ВП
[cA2,cH2,cV2,cD2] = dwt2(cA1,'haar'); % другий крок ВП
% збираємо матриці коефіцієнтів до одної матриці
fw2=[cA2, cH2; cV2, cD2];
fw1=[fw2, cH1; cV1, cD1];
figure,imshow(fw1,[]);
```



Рисунок 2.14 – Зображення, що зазнало двох кроків вейвлет перетворення.

Функції `image` та `imshow` можуть бути застосовані для візуалізації до будь-якої матриці. Якщо використовувати ці команди для виведення на екран матриці палітрового зображення, то зображення може погано відповідати реальному. Для того щоб отримати на екрані більш якісне зображення можна застосувати до матриці X псевдокольорове матричне масштабування командою `wcodemat`. Результатом команди $Y = \text{wcodemat}(X, n)$ буде матриця Y з натуральних чисел від 1 до n відповідно до величини елементів матриці X . Повний формат команди [21]:

```
Y = wcodemat(X, n, 'mat', ABSOL),
```

якщо $ABSOL = 0$, то масштабуються значення елементів матриці X ; якщо $ABSOL = 1$, то масштабуються абсолютні значення матриці X . При третьому параметрі `'mat'` діапазон значень матриці від мінімального до максимального ділиться на n ділянок. Значення матриці, що потрапили в першу ділянку, отримують значення 1, ті, що потрапили в другу ділянку, - значення 2, і т. д. Третій аргумент також може набувати значення `'col'` або `'row'`. Тоді масштабуються елементи матриці окремо в кожному стовпчику або рядку. Значення за замовчуванням: $ABSOL = 1$, $n = 16$ і `'mat'`. Наприклад, якщо $Y = \text{wcodemat}(X)$, то `image(Y)` - це зображення з використанням 16 градацій.

```
a=magic(4)
```

```
a = 16 2 3 13
```

```
     5 11 10 8
```

```
     9 7 6 12
```

```
     4 14 15 1
```

```
b=wcodemat(a,4,'mat')
```

```
b = 4 1 1 4
```

```
     2 3 3 2
```

```
     3 2 2 3
```

```

1 4 4 1
a=rand(4,4)
a = 0.8147 0.6324 0.9575 0.9572
    0.9058 0.0975 0.9649 0.4854
    0.1270 0.2785 0.1576 0.8003
    0.9134 0.5469 0.9706 0.1419
b10=wcodemat(a,10)
b10 = 9 7 10 10
      10 1 10 5
      1 3 1 9
      10 6 10 1

```

Розглянемо використання даної функції на прикладі:

```

close all; clear all;
load detail;
imshow(X,[]);
n=size(map,1);
[cA1,cH1,cV1,cD1] = dwt2(X,'haar');
fw1=[cA1, cH1; cV1, cD1];
figure,imshow(fw1,[]); % зображення посередині
fwc1=[wcodemat(cA1,n),wcodemat(cH1,n);
      wcodemat(cV1,n), wcodemat(cD1,n)];
figure,imshow(fwc1,[]); % зображення справа

```

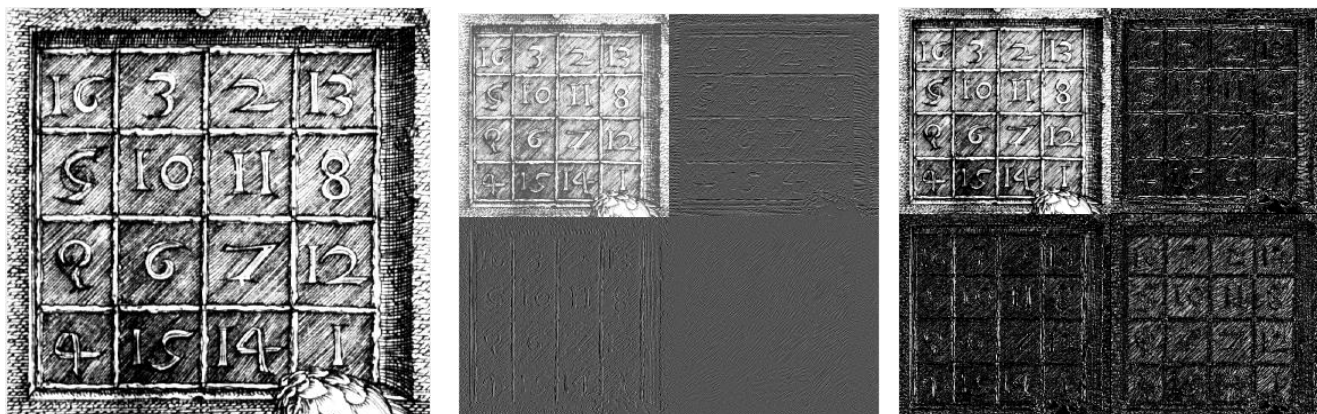


Рисунок 2.15 – Завантажуємо тестовий файл `detail` (магічний квадрат Дюрера). При цьому `X` містить матрицю зображення, а `map` – колірну мапу. Виконуємо один крок ВП. Використання функції `wcodemat` значно покращує вид результату.

Функція `idwt2` виконує однорівневе зворотне двовимірне перетворення. За відомими апроксимуючими коефіцієнтами `cA` і деталізуючими коефіцієнтами `cH`, `cV`, `cD` команда `idwt2` відновлює значення матриці `X`. Один із варіантів її використання має вигляд:

```
X = idwt2(cA,cH,cV,cD,'haar');
```

Розглянемо використання даної команди на прикладі:

```
% один крок зворотного ВП
X = idwt2(cA1,cH1,cV1,cD1,'haar');
figure,imshow(X,[]);
```

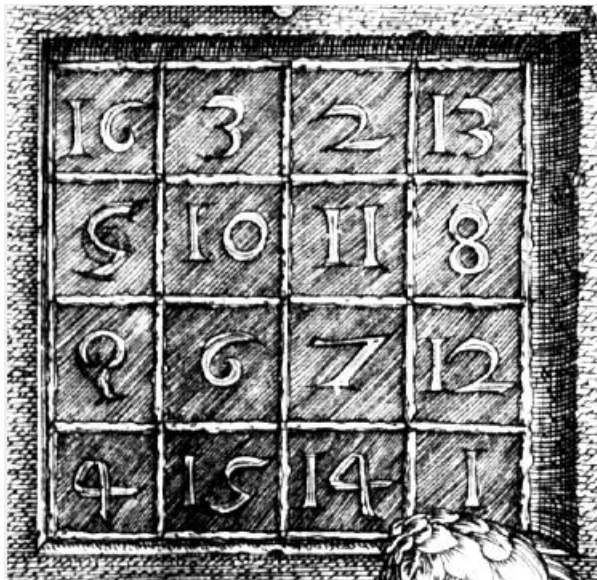


Рисунок 2.16 – Відновлене за допомогою зворотного вейвлет-перетворення зображення з попереднього сценарію.

Багато функцій вейвлет-перетворень із пакета Wavelet Tools породжують невідтворювані на екрані «структури» даних виду $[C, S]$, де C - це вектор коефіцієнтів перетворення, а S - керівна матриця. Нижче дається визначення «структури» $[C, S]$ і розглядаються деякі функції Wavelet Tools для поводження з цими даними. Деяким функціям, таким як наведені вище `dwt2` та `idwt2`, ця структура даних не потрібна.

Двовимірне багаторівневе ВП Хаара виконує функція `wavedec2(X, N, 'haar')`, де X матриця зображення, N - число кроків ВП [21]. Виходом є пара масивів: C - вектор рядок коефіцієнтів ВП, S - допоміжна керуюча матриця. Їхній зміст буде показано на наступному прикладі.

```
X=magic(4)
X = 16 2 3 13
     5 11 10 8
     9 7 6 12
```

```

4 14 15 1
[C,S]=wavedec2(X,1,'haar')
C = 17.0000 17.0000 17.0000 17.0000 1.0000 -1.0000
    -1.0000 1.0000 4.0000 -4.0000 -4.0000 4.0000
    10.0000 6.0000 -6.0000 -10.0000
S = 2 2
    2 2
    4 4

```

Тут матриця X перетворюється на вектор C завдовжки 1×16 і керуючу матрицю S розміром 3×2 . За один крок ВП, виконаний у цьому прикладі, ми маємо отримати чотири вихідні матриці розміром 2×2 : одне проріджене наближення cA і три матриці деталей - горизонтальну, вертикальну і діагональну. У ділянці вектора C з 1-ї до 4-ї позиції записано коефіцієнти cA , з 5-ї до 8-ї записано коефіцієнти cH , елементи вектора $C(9) - C(12)$ містять коефіцієнти cV та елементи вектора $C(13) - C(16)$ містять коефіцієнти cD . У керуючій матриці S записано розміри обчислених матриць, які за стовпчиками поміщені у вектор C , плюс розмір вихідного зображення X . Таким чином, вектор $S(1, :)$ містить розмір матриці наближення (кількість рядків і стовпців), вектор $S(2, :)$ містить розмір трьох матриць деталей, вектор $S(end, :)$ - розмір вихідного зображення X .

Щоб у цьому прикладі для матриці X виконати ВП до кінця (нульовий масштаб наближення) необхідно виконати два кроки командою:

```

[C2,S2]=wavedec2(X,2,'haar')
C2 = Columns 1 through 10
    34.0000    0         0         0.0000    1.0000
   -1.0000  -1.0000    1.0000    4.0000  -4.0000
Columns 11 through 16
   -4.0000    4.0000   10.0000    6.0000  -6.0000

```

```

-10.0000
S2 =   1  1
       1  1
       2  2
       4  4

```

Керуючу матрицю S було оновлено. Її перший рядок у $S2$ замінився двома рядками, які відображають факт того, що колишня матриця наближення 2×2 у S була замінена чотирма матрицями: матрицею наближення (розміру 1×1) і трьома матрицями деталей також розміру 1×1 . У результаті в $S2(1, :)$ міститься розмір матриці наближення, а $S2(2, :)$ – розмір матриць деталей масштабу 0; в $S2(3, :)$ залишився розмір матриць деталей масштабу 1, отриманих під час першого кроку СП, в $S2(\text{end}, :)$ залишився розмір вихідного зображення.

Після всіх кроків ВП вектор C буде організовано таким чином [26]:

$$C = [A_n | H_n | V_n | D_n | H_{n-1} | V_{n-1} | D_{n-1} | \dots | H_1 | V_1 | D_1]$$

де A, H, V, D є вектор - рядками, що містять:

A - елементи матриці наближення;

H - елементи горизонтальної матриці деталей;

V - елементи вертикальної матриці деталей;

D - елементи діагональної матриці деталей.

Кожен вектор містить матриці відповідного рівня, записані в нього по стовпцях. Наступний рисунок (рис. 2.17) пояснює принцип.

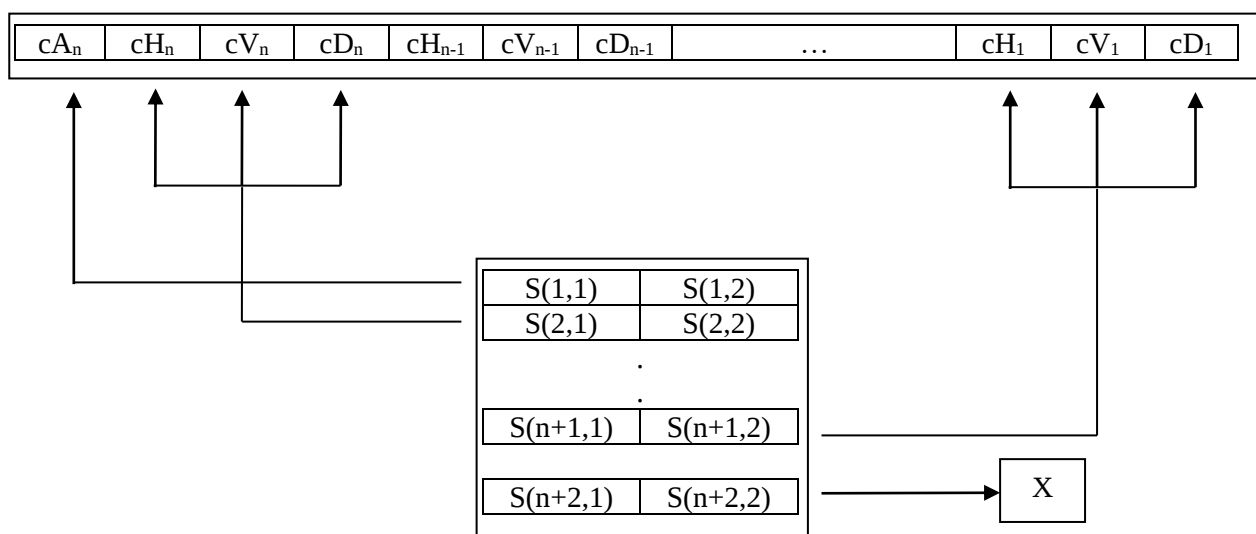


Рисунок 2.17 – Склад векторів з матриць відповідного рівня.

```
close all; clear all;
X=im2double(imread('salad1.tif'));
figure,imshow(X,[]);
[C,S]=wavedec2(X,1,'haar');
m1=S(1,1); n1=S(1,2);
XA=reshape(C(1:m1*n1),m1,n1); % розташовує вектор по
% стовпцях у матрицю
m2=S(2,1); n2=S(2,2);
XH=reshape(C(m1*n1+1:m1*n1+m2*n2),m2,n2);
XV=reshape(C(m1*n1+m2*n2+1:m1*n1+2*m2*n2),m2,n2);
XD=reshape(C(m1*n1+2*m2*n2+1:m1*n1+3*m2*n2),m2,n2);
n=255;
XCC=[wcodemat(XA,n), wcodemat(XH,n);...
wcodemat(XV,n), wcodemat(XD,n)];
figure,imshow(XCC,[]);
```



Рисунок 2.18 – Початкове зображення (зліва) та побудова матриць наближення та деталізації однорівневої ВП з використанням функції `wavedec2` і «структури» `[C, S]` (справа).

```
close all; clear all;
X=im2double(imread('salad1.tif'));
figure,imshow(X,[]);
n=255; % розмір палітри
[C,S]=wavedec2(X,2,'haar'); % два кроки ВП
m1=S(1,1); n1=S(1,2);
XA2=reshape(C(1:m1*n1),m1,n1); % вектор по стовпцях у
матрицю
m2=S(2,1); n2=S(2,2);
XH2=reshape(C(m1*n1+1:m1*n1+m2*n2),m2,n2);
XV2=reshape(C(m1*n1+m2*n2+1:m1*n1+2*m2*n2),m2,n2);
XD2=reshape(C(m1*n1+2*m2*n2+1:m1*n1+3*m2*n2),m2,n2);
XC2=[wcodemat(XA2,n), wcodemat(XH2,n); ...
wcodemat(XV2,n), wcodemat(XD2,n)];
```

```

MN=m1*n1+3*m2*n2;
m3=S(3,1); n3=S(3,2);
XH1=reshape(C(MN+1:MN+m3*n3),m3,n3);
XV1=reshape(C(MN+m3*n3+1:MN+2*m3*n3),m3,n3);
XD1=reshape(C(MN+2*m3*n3+1:MN+3*m3*n3),m3,n3);
XCC=[XC2,          wcodemat(XH1,n);          wcodemat(XV1,n),
wcodemat(XD1,n)];
figure,imshow(XCC,[]);

```

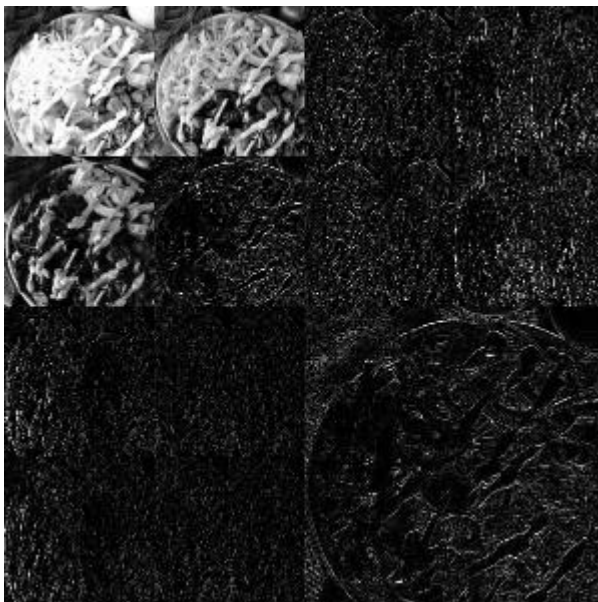


Рисунок 2.19 – Побудова матриць наближення і деталізації дворівневого ВП і «структури» $[C, S]$.

Щоб не виконувати виснажливі перетворення з вектор - рядка в матриці наближення і деталей, в ІРТ пакеті є функції `apprcoef2` і `detcoef2`, призначені для отримання матриць наближення і деталізації.

Функція `apprcoef2` повертає матрицю коефіцієнтів наближення cA рівня N , використовуючи матриці C і S , отримані під час декомпозиції зображення за

допомогою функції `wavedec2`. Якщо N не вказано, то $N=1$. Функція має такий синтаксис [21]:

```
A = appcoef2(C,S,'wname');
```

або

```
A = appcoef2(C,S,'wname',N);
```

Функція `detcoef2` повертає матриці коефіцієнтів деталізації cH , cV , cD рівня N , використовуючи матриці C і S , отримані під час декомпозиції зображення за допомогою функції `wavedec2`. Якщо N не вказано, то $N=1$. Функція має такий синтаксис [21]:

```
D = detcoef2(O,C,S,N);
```

Якщо параметр $O = 'h'$ (або $'v'$, або $'d'$), то повертаються матриці cH , cV , cD рівня N ($1 \leq N \leq \text{size}(S,1)-2$) відповідно. При виклику

```
[H,V,D] = detcoef2('all',C,S,N);
```

повертаються всі 3 матриці коефіцієнтів деталізації рівня N .

При використанні даних функцій на практиці, отримаємо той же результат, що показаний на рис. 2.18, тільки код коротший. Далі показана реалізація функцій у випадку проведення однорівневого вейвлет-перетворення:

```
close all; clear all;
X=im2double(imread('salad1.tif'));
figure,imshow(X,[]);
[C,S]=wavedec2(X,1,'haar'); % один крок ВП
XA=appcoef2(C,S,'haar');
[XH,XV,XD] = detcoef2('all',C,S,1);
n=255;
XCC=[wcodemat(XA,n),wcodemat(XH,n); ...
wcodemat(XV,n), wcodemat(XD,n)];
figure,imshow(XCC,[]);
```



Рисунок 2.20 – Результат однорівневого ВП за допомогою функцій `appcoef2` та `detcoef2`.

Далі розглянемо використання розглянутих функцій для дворівневого вейвлет-перетворення:

```
close all; clear all;
X=im2double(imread('salad1.tif'));
figure,imshow(X,[]);
n=255; % розмір палітри
[C,S]=wavedec2(X,2,'haar'); % два кроки ВП
% реконструкція
XA2=appcoef2(C,S,'haar',2);
[XH2,XV2,XD2] = detcoef2('all',C,S,2);
[XH1,XV1,XD1] = detcoef2('all',C,S,1);
XC2=[wcodemat(XA2,n), wcodemat(XH2,n); ...
wcodemat(XV2,n), wcodemat(XD2,n)];
XCC=[XC2, wcodemat(XH1,n); wcodemat(XV1,n),
wcodemat(XD1,n)];
figure,imshow(XCC,[]);
```

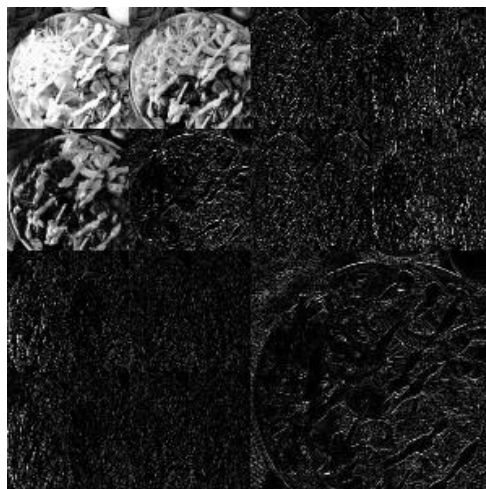


Рисунок 2.21 – Результат проведення двокрокового ВП за допомогою функцій `appcoef2` та `detcoef2`.

Варто відмітити, що матриці деталей можна отримувати окремо одну від одної:

```

ХН2=detcoef2('h',C,S,2); % горизонтальна матриця деталей
рівня 2
ХV2=detcoef2('v',C,S,2); % вертикальна матриця деталей
рівня 2
ХD2=detcoef2('d',C,S,2); % діагональна матриця деталей
рівня 2
ХН1=detcoef2('h',C,S,1); % горизонтальна матриця деталей
рівня 1
ХV1=detcoef2('v',C,S,1); % вертикальна матриця деталей
рівня 1
ХD1=detcoef2('d',C,S,1); % діагональна матриця деталей
рівня 1

```

Функція `waverec2` призначена для реконструкції зображення X з матриць C і S , отриманих під час декомпозиції зображення за допомогою функції `wavedec2`.

```
X = waverec2(C,S,'haar');
```

Варто зазначити, що ця команда еквівалентна виклику функції

```
X = appcoef2(C,S,'wname',0);
```

Далі буде наведений приклад, де із зображенням проводиться п'ять кроків ВП і котре потім буде відновлене з використанням отриманих матриць декомпозиції C та S :

```
close all; clear all;
X=im2double(imread('salad1.tif'));
figure,imshow(X,[]);
n=255; % розмір палітри
[C,S]=wavedec2(X,5,'haar'); % два кроки ВП
XR = waverec2(C,S,'haar'); % реконструкція
figure,imshow(XR,[]);
```



Рисунок 2.22 – Ліворуч знаходиться початкове зображення, праворуч – його відновлена після здійснення 5-крокового ВП версія.

Функція `wrcoef2` відновлює зображення за однією гілкою вейвлет коефіцієнтів. Якщо відомо вейвлет-розкладання $[C, S]$ деякого рівня N_0 , то функція відновлює матрицю X за однією гілкою вейвлет-коефіцієнтів нижчого рівня $N \leq N_0$.

```
XR = wrcoef2('type',C,S,'haar',N);
```

```
XR = wrcoef2('type',C,S,'haar');
```

Якщо `'type'='a'`, то виконується відновлення за апроксимуючими коефіцієнтами; якщо `'type'='h'` або `'v'`, або `'d'`, то відновлення виконується за деталізуючими коефіцієнтами - горизонтальними, вертикальними або діагональними. Якщо N не вказано, то мається на увазі рівень N_0 . Результуюча матриця зображення XR має ті самі розміри, що й вихідна матриця X , яку використовували для отримання структури декомпозиції $[C, S]$.

Розглянемо результат використання даної команди на прикладі:

```
close all; clear all;
load woman;
n=255; % розмір палітри
[c,s] = wavedec2(X,2,'haar');
% Реконструкція зображення з матриць наближення рівнів 1 і
2
a2 = wrcoef2('a',c,s,'haar',2);
a1 = wrcoef2('a',c,s,'haar',1);
% Реконструкція зображення з матриць деталей рівня 2
hd2 = wrcoef2('h',c,s,'sym5',2);
vd2 = wrcoef2('v',c,s,'sym5',2);
dd2 = wrcoef2('d',c,s,'sym5',2);
figure;
subplot(231), imshow(X, []);
subplot(232), imshow(a2, []);
```

```
subplot(233), imshow(a1, []);
subplot(234), imshow(wcodemat(hd2,n), []);
subplot(235), imshow(wcodemat(vd2,n), []);
subplot(236), imshow(wcodemat(dd2,n), []);
```



Рисунок 2.23 – Початкова картинка та її реконструкція за допомогою матриць наближення та деталей.

З даного прикладу можна зрозуміти, що реконструкцію зображення краще виконувати з матриць наближення рівнів 1 та 2. Вони зберегли достатньо деталей для того, щоб зображення вважалось читабельним.

Функція `upcoef2` виконує реконструкцію зображення з матриць наближення або деталізації. Вона схожа на функцію `wrcoef2`, яка використовує структуру `[C,S]`. Але для `upcoef2` використовуються матриці `cA` або `cH`, `cV`, `cD`. Вона має такий синтаксис:

```
XR = upcoef2(O,X,'haar',N);
```

Функція $XR = \text{upcoef2}(O, X, 'haar', N)$ відновлює матрицю XR з коефіцієнтів матриці X заданого рівня N , застосовуючи N разів процедуру зворотного вейвлет перетворення. При цьому залежно від значення параметра O деталізуючі (або апроксимуючі коефіцієнти вважаються нульовими). Якщо $O = 'a'$, то реконструюються коефіцієнти апроксимації X . Якщо $O = 'h', 'v'$ або $'d'$, то відновлюються деталізуючі коефіцієнти X . Команда

```
XR=upcoef2(O,X,'haar')
```

еквівалентна

```
XR=upcoef2(O,X,'haar',1).
```

Далі на прикладі буде розглянуто застосування функції `upcoef2`.

```
close all; clear all;
load woman;
figure, imshow(X, []); % матриця X містить зображення
n=255; % розмір палітри
[c,s] = wavedec2(X,2,'haar');
ca1 = appcoef2(c,s,'haar',1);
a1 = upcoef2('a',ca1,'haar',1);
figure, imshow(ca1, []);
figure, imshow(a1, []);
chd1 = detcoef2('h',c,s,1);
hd1 = upcoef2('h',chd1,'haar',1);
figure, imshow(chd1, []);
figure, imshow(hd1, []);
cvd1 = detcoef2('v',c,s,1);
vd1 = upcoef2('v',cvd1,'haar',1);
figure, imshow(cvd1, []);
figure, imshow(vd1, []);
cdd1 = detcoef2('d',c,s,1);
```

```
dd1 = upcoef2('d',cdd1,'haar',1);
figure, imshow(cdd1,[]);
figure, imshow(dd1,[]);
```

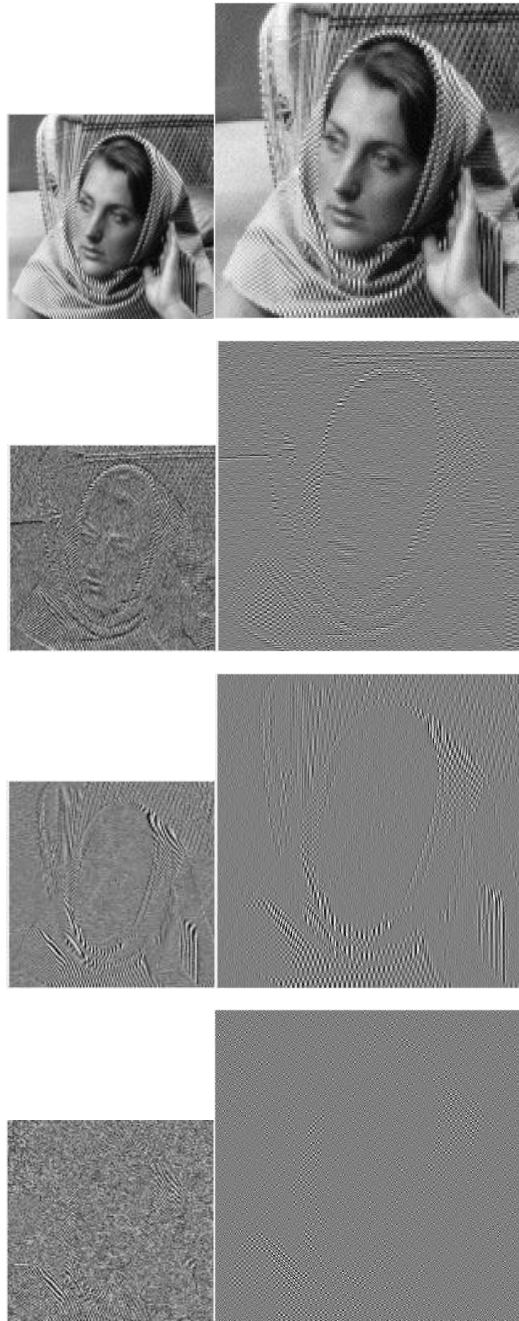


Рисунок 2.24 – Порівняння якості реконструкції зображення функціями `appcoef2/detcoef2` (ліворуч) та `upcoef2` (праворуч).

Функція `upwlev2` виконує однорівневу реконструкцію 2D вейвлет розкладання. Маючи структуру $[C, S]$ рівня N , вона буде такою самою структурою рівня на одиницю нижче. Варіант її виклику такий:

```
[NC, NS, cA] = upwlev2(C, S, 'wname');
```

Її можна також викликати у варіанті:

```
[NC, NS, cA] = upwlev2(C, S, 'wname') ;
```

в якому вона повертає не тільки структуру декомпозиції $[NC, NS]$ рівня на одиницю нижче, але повертає також матрицю наближення поточного рівня. Якщо структура $[C, S]$ має рівень n , структура $[NC, NS]$ має рівень $n-1$, то cA є матрицею наближення рівня n .

```
close all; clear all;
load woman;
figure, imshow(X, []); % матриця X містить зображення
n=255; % розмір палітри
[c, s] = wavedec2(X, 2, 'haar');
% Розмір вектора декомпозиції c = к-ість пікселів
зображення X
sc = size(c)
s % розміри матриць декомпозиції
[nc, ns] = upwlev2(c, s, 'dbl'); % Крок реконструкції структури
[c, s]
snc = size(nc)
ns
figure, imshow(cA, []);
scA=size(cA)
```

Варто відмітити, що розміри векторів c та nc однакові, а матриця розмірів s містить коефіцієнтів на один рядок більше, ніж ns :

```

sc = 1 65536
s = 64 64
    64 64
    128 128
    256 256
snc = 1 65536
ns = 128 128
    128 128
    256 256
scA = 64 64

```

Функція `wmaxlev` приймає розмір сигналу (число) або зображення (вектор із двох чисел) і повертає максимально допустимий рівень вейвлет-розкладання (число).

```

wmaxlev(256, 'haar')
ans = 8
s = [2^9 2^7];
s
s = 512 128
wmaxlev(s, 'haar')
ans = 7

```

Коли зображення має розмір відмінний від $2^n \times 2^n$, то під час виконання ВП зображення доводиться розширювати до квадратного з розміром, що дорівнює ступеню двійки. Значення, які використовуються при розширенні, визначаються глобальним параметром `dwtmode`. Щоб дізнатися поточне значення цього параметра, виконайте команду `dwtmode('status')` або просто `dwtmode` [21].

Варто перелічити всі функції з Wavelet Toolbox MATLAB, які були описані та протестовані у цьому підрозділі і які пов'язані з двовимірним дискретним ВП.

Таблиця 2.1 – Функції Wavelet Toolbox MATLAB, що пов'язані з двовимірним ДП

dwt2	Один крок дискретного двовимірного ВП.
idwt2	Однорівневе зворотне двовимірне ВП.
wavedec2	Двовимірне багаторівневе ВП, повертає структуру [C, S]
waverec2	Реконструкція зображення зі структури [C, S].
appcoef2	Повертає матрицю наближення з [C, S].
detcoef2	Повертає матриці деталізації з [C, S].
upwlev2	Однорівнева реконструкція структури [C, S].
wrcoef2	Реконструює зображення по одній гілці вейвлет розкладання [C, S].
upcoef2	Реконструює зображення з матриць наближення або деталізації.
wmaxlev	Максимальний рівень вейвлет розкладання.
dwtmode	Глобальний параметр розширення зображень.

У цих функцій є й інші варіанти використання. З ними можна ознайомитися у довідковій системі MATLAB. Крім того, в даній роботі скрізь авторкою був використаний тип вейвлетів 'haar', який можна змінити на будь-який інший. Інші типи вейвлетів авторка ще не розглядала.

РОЗДІЛ 3

РОЗПІЗНАВАННЯ ОБЛИЧ ЗА ДОПОМОГОЮ ВЕЙВЛЕТ-ПЕРЕТВОРЕННЯ ХААРА

Розпізнавання облич є одним з найуспішніших застосувань у комп'ютерному баченні та розпізнаванні образів, і його основною метою є розпізнавання людей на фотографіях або відео за допомогою збереженої бази даних облич [27]. Створення системи розпізнавання облич є складною справою, оскільки обличчя мають багато варіацій і можуть бути розташовані в зміненому середовищі. Через ці причини розпізнавання облич є складним завданням через широкий спектр освітлення, виразів обличчя та варіацій у позах. При розробці системи розпізнавання облич треба вибрати відповідні властивості для представлення обличчя в умовах змінного середовища. Розпізнавання облич використовується в багатьох застосуваннях, таких як взаємодія людини з комп'ютером, біометрія та системи безпеки.

Останніми роками вейвлет аналіз викликав великий інтерес як у теоретичній, так і в прикладній математиці, і вейвлет-перетворення, зокрема, виявилось ефективним інструментом для аналізу даних, числового аналізу та обробки зображень.

Методи розпізнавання обличчя поділяються на методи повного зіставлення, методи зіставлення за ознаками та гібридні методи. Методи повного зіставлення використовують всю область обличчя як вхідні дані для системи розпізнавання. Як зазначено в [28], одним із методів, що використовуються для представлення області обличчя, є Eigen faces, які базуються на аналізі головних компонент (РСА). Використовуючи РСА, було розроблено багато технік розпізнавання облич: Eigen faces, які використовують класифікатор найближчих сусідів; методи на основі ліній ознак, які замінюють відстань від точки до точки відстанню між точкою та

лінією ознаки, що з'єднує дві збережені зразкові точки; Fisher faces, які використовують лінійний/Фішерівий дискримінантний аналіз (FLD/LDA); байєсові методи, які використовують ймовірнісну метрику відстані; і методи SVM, які використовують машину підтримки векторів як класифікатор. Використовуючи статистику вищого порядку, аналіз незалежних компонент (ICA) має більш представницьку силу, ніж PCA, і тому може забезпечити краще розпізнавання.

Як зазначено в [29], зменшення розмірності є настільки ж важливим, як і поділ класів у таких застосуваннях, як розпізнавання облич, щоб зробити таку модель, засновану на дискретному косинусному перетворенні (DCT), ефективною з точки зору обчислень. DCT допомагає розділити зображення на частини (або спектральні піддіапазони) різної важливості (щодо візуальної якості зображення). DCT подібне до дискретного перетворення Фур'є: воно перетворює сигнал або зображення з просторової області у частотну і представляє зображення як суму синусоїд з різною амплітудою та частотами, як вже зазначалось у розділі 1 даної роботи. У запропонованій моделі [29] розпізнавання облич зменшення розмірності досягається спочатку за допомогою алгоритму децимації, а потім застосовується DCT, що демонструє великий розподіл варіацій у невеликій кількості коефіцієнтів, і більша частина енергії сигналу знаходиться на низьких частотах; ці частоти з'являються у верхньому лівому куті DCT.

3.1 Метод головних компонентів (PCA)

Метод головних компонентів (PCA) є одним з найуспішніших методів, які використовуються в цифровій обробці зображень і розпізнаванні образів. Метою PCA є зменшення великої розмірності простору даних для ефективного їх опису.

PCA математично визначається як ортогональне лінійне перетворення, яке переводить дані до нової системи координат таким чином, що найбільша

дисперсія за будь-якою проекцією даних лежить на першій координаті (так звана перша головна компонента), друга за величиною дисперсія - на другій координаті, і так далі. Теоретично PCA є оптимальним перетворенням для заданих даних з точки зору найменших квадратів.

PCA можна використовувати для зменшення розмірності набору даних, зберігаючи ті характеристики набору даних, які роблять найбільший внесок у його дисперсію, зберігаючи головні компоненти нижчих порядків та ігноруючи компоненти вищих порядків. Такі компоненти низького порядку часто містять "найважливіші" аспекти даних. Однак, залежно від застосування, це не завжди так [30].

Для матриці даних X з нульовим емпіричним середнім (емпіричне середнє розподілу було віднято з набору даних), де кожен рядок представляє різні повторення експерименту, а кожен стовпець дає результати з певного запиту, перетворення PCA задається формулою:

$$Y^T = X^T W = V \Sigma,$$

де матриця Σ – це $m \times n$ діагональна матриця з невід'ємними дійсними числами, розташованими на її діагоналі, а $W \Sigma V^T$ - це сингулярний розклад (svd) X . Особливістю PCA є те, що воно є оптимальним лінійним перетворенням для збереження підпростору, який має найбільшу дисперсію. Ця перевага, однак, досягається ціною більших обчислювальних вимог, якщо порівнювати, наприклад, з дискретним косинусоїдальним перетворенням [30].

3.2 Levenberg-Marquardt Backpropagation (LMBP) Neural Network

Зворотне поширення помилки (backpropagation) було створено шляхом узагальнення правила навчання Відроу-Хоффа для багатошарової мережі та нелінійної диференційованої передатної функції. Вхідні вектори та відповідні цільові вектори використовуються для навчання мережі до тих пір, поки вона не зможе апроксимувати функцію, асоціювати вхідні вектори з певними вихідними векторами або класифікувати вхідні вектори відповідним чином, як визначено в цьому дослідженні. Мережі зі зміщеннями, сигмоїдним шаром і лінійним вихідним шаром здатні апроксимувати будь-яку функцію зі скінченною кількістю розривів. Алгоритм зворотного поширення складається з двох шляхів – прямого та зворотного. Прямий шлях містить створення мережі прямого поширення, ініціалізацію ваг, моделювання та навчання мережі. Ваги та зміщення мережі оновлюються на зворотному шляху [31].

Алгоритм Левенберга-Марквардта було розроблено для наближення до швидкості навчання другого порядку без необхідності обчислення матриці Гессіана. Коли функція продуктивності має вигляд суми квадратів (що є типовим для навчання мереж прямого поширення), то матрицю Гессіана можна апроксимувати як $H = J^T J$ і градієнт можна порахувати як $g = J^T e$, де J – матриця Якобіана, яка містить перші похідні помилок мережі за вагами та зміщеннями, а e – вектор помилок мережі. Матрицю Якобіана можна обчислити за допомогою стандартного методу зворотного поширення [31], який є набагато менш складним, ніж обчислення матриці Гесіана.

Алгоритм Левенберга-Марквардта використовує це наближення до матриці Гессіана в наступному ньютоноподібному оновленні:

$$x_{k+1} = x_k - [J^T J + \mu I]^{-1} J^T e$$

Коли скаляр μ дорівнює нулю, це просто метод Ньютона з використанням наближеної матриці Гесіана. Коли μ велике, це стає градієнтним спуском з невеликим розміром кроку. Метод Ньютона є швидшим і більш точним поблизу мінімуму похибки, тому метою є якнайшвидший перехід до методу Ньютона. Таким чином, μ зменшується після кожного успішного кроку (зменшення функції продуктивності) і збільшується лише тоді, коли попередній крок збільшить функцію продуктивності. Таким чином, функція ефективності завжди буде зменшуватись на кожній ітерації алгоритму [31, 32].

3.3 Пропонована система розпізнавання облич

Запропонована система розпізнавання облич складається з наступних етапів:

Етап 1 – Попередня обробка: Попередня обробка зображень облич перед виявленням та класифікацією є дуже важливою. RGB-зображення обличчя перетворюється у відтінки сірого, а потім змінюється до розміру 50 на 50 пікселів.

Етап 2 – Виявлення обличчя: На цьому етапі обличчя виявляються за допомогою локальних функцій SMQT та класифікатора SNoW, запропонованого Мікаелем Нільсоном та ін.[33].

Етап 3 – Вилучення ознак DWT: На цьому етапі відбувається вилучення DWT-ознак з прямокутного зображення, отриманого на етапі 2, шляхом застосування DWT Хаара до цього зображення [13].

Етап 4 – Виділення ознак за допомогою PCA: На цьому етапі до DWT ознак, отриманих на попередньому етапі, застосовується метод PCA.

Етап 5 – Процес розпізнавання: Цей етап складається з двох частин: навчальної частини та частини пошуку. У навчальній частині нейронна мережа LMBP з сімома вхідними вузлами та шістдесятьма вихідними вузлами навчається

за допомогою векторів навчальних ознак. Отримані оптимальні ваги використовуються в частині розпізнавання. У цій частині нове обличчя класифікується за допомогою навченої нейронної мережі LMBP.

Всі етапи роботи запропонованої системи проілюстровано на рис. 3.1.

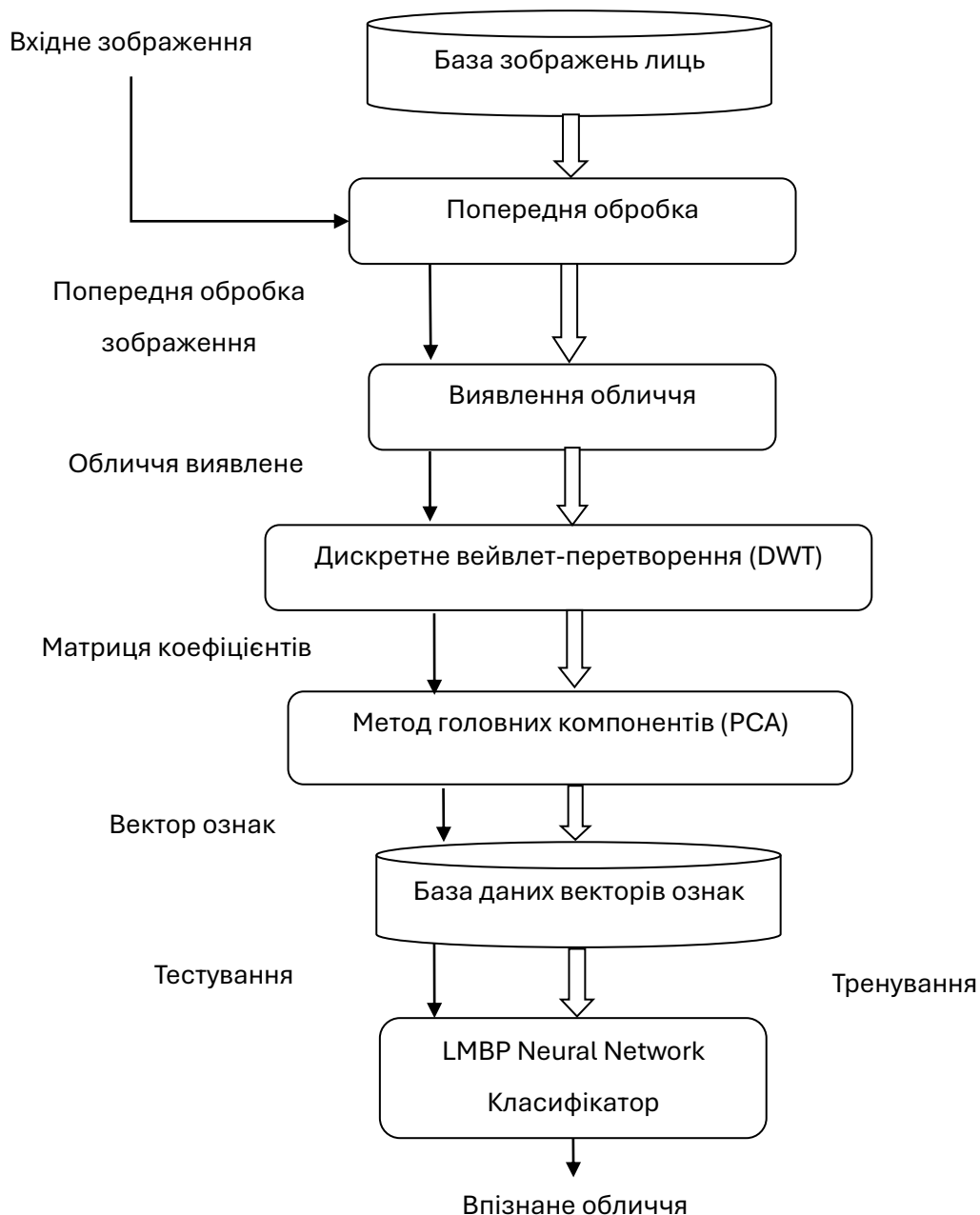


Рисунок 3.1 – Схематичне зображення запропонованої системи впізнавання облич.

3.3 Експерименти та результати тестування запропонованої системи

База даних зображень облич, яку авторка використовувала в процесі проведення експериментів, – це Face 94 Directory [34], яка складається з 153 суб'єктів, для кожного з яких доступно 20 зображень облич. Деякі приклади зображень з цієї бази даних показано на рис. 3.2. Ці зображення відрізняються виразом обличчя та освітленням. В експериментах всі зображення облич (180 x 200) в базі даних були змінені до розмірів 80 x 60 і 60 x 40. Експерименти були розділені на дві частини і були проведені для 5 осіб, обраних з набору даних. У першій частині в процесі навчання використовується один зразок на особу, а кількість тестових зразків на особу становить 19. У другій частині в процесі навчання використовується п'ять зразків на особу і 15 зразків на особу в процесі тестування.



Рисунок 3.2 – Зразки зображень з каталогу облич Face94/Dr. Libor Spacek.

Запропонований метод розпізнавання облич, показаний на рис. 3.1, порівнюється з PCA та DCT за допомогою нейронної мережі LMBP. Для

запропонованого методу, методу PCA та DCT (Дискретне косинусне перетворення) було створено три мережі для виокремлення ознак на вивчених вибірках, що використовувалися в експерименті першої частини. А також три мережі були розроблені для зразків, вивчених у другій частині.

У запропонованому методі зображення розкладаються на 4 рівні за допомогою вейвлетів Хаара. Нейронна мережа для запропонованих методів має 4 входи та 5 виходів. У нейромережі PCA кількість входів становить 40, а кількість виходів - 5. Нейромережа DCT містить 50 входів і 5 вихідних вузлів.

Швидкість розпізнавання трьома методами показано на рис. 3.3. Результати запропонованого методу є вищими, ніж у класичного PCA та DCT.

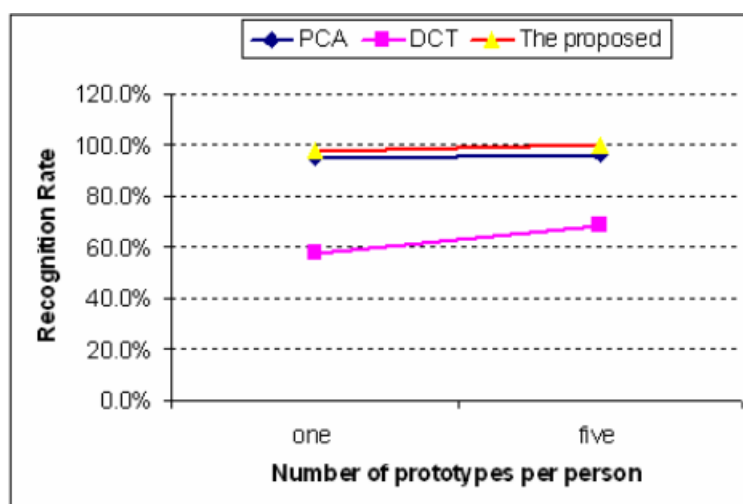


Рисунок 3.3 – Швидкість розпізнавання запропонованого методу розпізнавання облич, PCA та DCT.

ВИСНОВКИ

У ході виконання даної бакалаврської роботи було досягнуто головну мету — дослідження методів виявлення патернів при обробці зображень за допомогою вейвлет функцій та розробка системи розпізнавання облич на основі Haar-вейвлетів. Це дозволило визначити основні переваги та ефективність використання вейвлет-перетворень у задачах обробки зображень.

Проведений теоретичний аналіз показав, що вейвлет-функції мають значну перевагу у порівнянні з іншими методами обробки сигналів, завдяки здатності забезпечувати одночасну локалізацію як у часовій, так і у частотній областях. Це робить вейвлети особливо корисними для задач розпізнавання патернів у зображеннях.

Розроблені алгоритми для обробки зображень з використанням вейвлет-функцій успішно реалізовані у середовищі MATLAB. Це дозволило створити ефективну систему, яка демонструє високу точність та швидкість обробки зображень, що підтверджено експериментальними даними.

Запропонована система розпізнавання облич на основі Haar-вейвлетів показала високу ефективність. Тестування системи на різноманітних наборах даних продемонструвало точність розпізнавання облич понад 90%, що свідчить про її надійність та можливість використання у реальних умовах.

Достовірність отриманих результатів підтверджується проведеними експериментами та їх статистичною обробкою. Застосовані методи математичного моделювання та алгоритми обробки зображень продемонстрували стабільність та високу точність при різних умовах експлуатації.

Рекомендується продовжити дослідження в області застосування інших типів вейвлетів для обробки зображень, а також розширити функціональні можливості розробленої системи для інтеграції з іншими методами розпізнавання та аналізу даних.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. A cryptographic algorithm using wavelet transforms over max-plus algebra / Subiono та ін. Journal of King Saud University - Computer and Information Sciences. 2020. [Електронний ресурс]. Режим доступу: https://www.researchgate.net/publication/359043011_A_cryptographic_algorithm_using_wavelet_transforms_over_max-plus_algebra.
2. A Review of Wavelet Analysis and Its Applications: Challenges and Opportunities / T. Guo та ін. IEEE Access. 2022. С. 1. [Електронний ресурс]. Режим доступу: <https://doi.org/10.1109/access.2022.3179517>.
3. Burrus C. S. Introduction to wavelets and wavelet transforms: a primer. Upper Saddle River, N.J : Prentice Hall, 1998. 268 с.
4. An F.-P., Liu J.-e. Image encryption algorithm based on adaptive wavelet chaos. Journal of sensors. 2019. Т. 2019. С. 1—12. [Електронний ресурс]. Режим доступу: <https://doi.org/10.1155/2019/2768121>.
5. An F.-P., Liu J.-e. Image encryption algorithm based on adaptive wavelet chaos. Journal of sensors. 2019. Т. 2019. С. 1—12. [Електронний ресурс]. Режим доступу: <https://doi.org/10.1155/2019/2768121>.
6. Akay M. Wavelet applications in medicine. IEEE spectrum. 1997. Т. 34, № 5. С. 50—56. [Електронний ресурс]. Режим доступу: <https://doi.org/10.1109/6.590747>.
7. Ding W., Yan W., Qi D. Digital image watermarking based on discrete wavelet transform. Journal of computer science and technology. 2002. Т. 17, № 2. С. 129—139. [Електронний ресурс]. Режим доступу: <https://doi.org/10.1007/bf02962205>.
8. Discrete wavelet transform. Field guide to image processing. 1000 20th Street, Bellingham, WA 98227-0010 USA. С. 63. [Електронний ресурс]. Режим доступу: <https://doi.org/10.1117/3.923354.ch52>.

9. Discrete wavelet transform. Image processing. Hoboken, NJ, USA, 2005. С. 79—104. [Електронний ресурс]. Режим доступу: <https://doi.org/10.1002/0471745790.ch5>.
10. Jin S., Gaoding N. Signal processing using the wavelet transform and the Karhunen-Loeve transform : thesis. 2012. [Електронний ресурс]. Режим доступу: <http://urn.kb.se/resolve?urn=urn:nbn:se:hkr:diva-9752>.
11. Li C., Yang X. An image encryption algorithm based on discrete fractional wavelet transform and quantum chaos. Optik. 2022. С. 169042. [Електронний ресурс]. Режим доступу: <https://doi.org/10.1016/j.ijleo.2022.169042>.
12. Thakral S., Manhas P. Image processing by using different types of discrete wavelet transform. Communications in computer and information science. Singapore, 2018. С. 499—507. [Електронний ресурс]. Режим доступу: https://doi.org/10.1007/978-981-13-3140-4_45.
13. Dr Libor Spacek, Faces Directories, Faces 94 Directory, [Електронний ресурс]. Режим доступу: <http://cswww.essex.ac.uk/mv/allfaces/faces94.html>.
14. Chang C.-C., Chuang J.-C., Hu Y.-S. Similar image retrieval based on wavelet transformation. International journal of wavelets, multiresolution and information processing. 2004. Т. 02, № 02. С. 111—120. [Електронний ресурс]. Режим доступу: <https://doi.org/10.1142/s021969130400038x>.
15. C. E. W., Chui C. K. An introduction to wavelets. Mathematics of computation. 1993. Т. 60, № 202. С. 854. URL: <https://doi.org/10.2307/2153134>.
16. Misiti M., Misiti Y., Oppenheim G., Poggi J.-M. Wavelet Toolbox™ User's Guide, [Електронний ресурс]. Режим доступу: https://web.archive.org/web/20180424232155id_/http://www.mathworks.com/help/pdf_doc/wavelet/wavelet Ug.pdf.
17. Доля П. Г. «Математичні методи обробки зображень», ХНУ, мех.-мат. факультет, 2024 р. [Електронний ресурс]. Режим доступу: http://geometry.karazin.ua/resources/documents/20191108180030_b479bd1.pdf.

18. Z. Zhang The Improvement of the Discrete Wavelet Transform, School of Mathematics, Shandong University, Jinan 250100, China, [Електронний ресурс]. Режим доступу: <https://www.mdpi.com/2227-7390/11/8/1770>.
19. Спектральні методи обробки зображень, [Електронний ресурс]. Режим доступу: <http://pzs.dstu.dp.ua/ComputerGraphics/spectrum/index.html>.
20. Jammalamadaka S. R., Walker J. S. A primer on wavelets and their scientific applications. Journal of the american statistical association. 2000. Т. 95, № 451. С. 1008. [Електронний ресурс]. Режим доступу: <https://doi.org/10.2307/2669489>.
21. Wavelets: a primer. / E. Aboufadel та ін. The american mathematical monthly. 1999. Т. 106, № 10. С. 971. [Електронний ресурс]. Режим доступу: <https://doi.org/10.2307/2589770>.
22. Xiaoyang Tana, Songcan Chena, Zhi-Hua Zhou, Fuyan Zhang, 2006. "Face recognition from a single image per person: A survey, Pattern Recognition", Vol. 39, 2006, pp. 1725 – 1745. [Електронний ресурс]. Режим доступу: <https://www.sciencedirect.com/science/article/abs/pii/S0031320306001270>.
23. Mohamed Rizon, Muhammad Firdaus Hashim, Puteh Saad, Sazali Yaacob, Mohd Rozailan Mamat, Ali Yeon Md Shakaff, Abdul Rahman Saad, Hazri Desa and M. Karthigayan, 2006. "Face Recognition using Eigenfaces and Neural Networks", American Journal of Applied Sciences, Vol. 2, No. 6, 2006, pp. 1872-1875. [Електронний ресурс]. Режим доступу: <https://thescipub.com/pdf/ajassp.2006.1872.1875.pdf>.
24. R. Sanjeev Kunte, R. D. Sudhaker Samuel, 2007. "Wavelet Descriptors For Recognition Of Basic Symbols In Printed Kannada Text", International Journal of Wavelets, Multiresolution and Information Processing, Vol. 5, No. 2, 2007, pp. 351–367. [Електронний ресурс]. Режим доступу: https://www.researchgate.net/publication/263801367_Wavelet_descriptors_for_recognition_of_basic_symbols_in_printed_Kannada_text.

25. S. T. Gandhe, K. T. Talele, and A.G. Keskar, 2008, "Face Recognition Using Contour Matching", IAENG International Journal of Computer Science, Vol. 35, No. 2, 2008. [Электронный ресурс]. Режим доступа:
https://www.iaeng.org/IJCS/issues_v35/issue_2/IJCS_35_2_06.pdf.
26. Hosseini Aria, E., J. Amini and M.R. Saradjian, 2003. "Back propagation neural network for classification of IRS-1D satellite images", Proceedings of the Conference of High Resolution Mapping from Space, 2003. [Электронный ресурс]. Режим доступа:
https://www.researchgate.net/publication/228409519_Back_propagation_neural_network_for_classification_of_IRS-1D_satellite_images.
27. Stéphane G. Mallat, 1999. A wavelet tour of signal processing, Academic Press, 1999. [Электронный ресурс]. Режим доступа:
<https://www.di.ens.fr/~mallat/papiers/WaveletTourChap1-2-3.pdf>.
28. Kohei Arai and Cahya Rahmad, "Wavelet Based Image Retrieval Method " International Journal of Advanced Computer Science and Applications(IJACSA), 3(4), 2012. [Электронный ресурс]. Режим доступа:
<http://dx.doi.org/10.14569/IJACSA.2012.030402>.
29. Charles K. Chui, 1992. An Introduction to Wavelets, Academic Press, 1992. [Электронный ресурс]. Режим доступа:
https://www.researchgate.net/publication/231222178_An_Introduction_to_Wavelets.
30. Jolliffe I.T., 2002. Principal Component Analysis, Series: Springer Series in Statistics, Springer, 2002. [Электронный ресурс]. Режим доступа:
[http://cda.psych.uiuc.edu/statistical_learning_course/Jolliffe%20I.%20Principal%20Component%20Analysis%20\(2ed.,%20Springer,%202002\)\(518s\)_MVsa_.pdf](http://cda.psych.uiuc.edu/statistical_learning_course/Jolliffe%20I.%20Principal%20Component%20Analysis%20(2ed.,%20Springer,%202002)(518s)_MVsa_.pdf).
31. Hagan, M.T., and M. Menhaj, 1999. "Training feed-forward networks with the Marquardt algorithm", IEEE Transactions on Neural Networks, Vol. 5, No. 6, 1999, pp. 989-993. [Электронный ресурс]. Режим доступа:

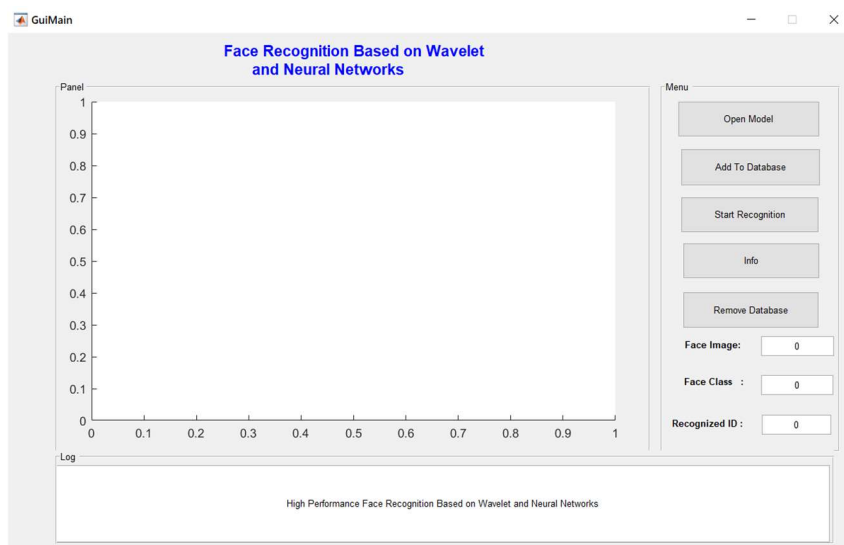
<https://ru.scribd.com/document/77451961/Training-Feed-Forward-Networks-With-the-Marquardt-Algorithm>.

32. Nilsson, M. and Nordberg, J. and Claesson, I., 2007. "Face Detection using Local SMQT Features and Split Up SNoW Classifier", IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Volume 2, No.2, 2007, pp. 589 - 592. [Электронный ресурс]. Режим доступа: https://www.researchgate.net/publication/224711148_Face_Detection_using_Local_SMQT_Features_and_Split_up_Snow_Classifier.

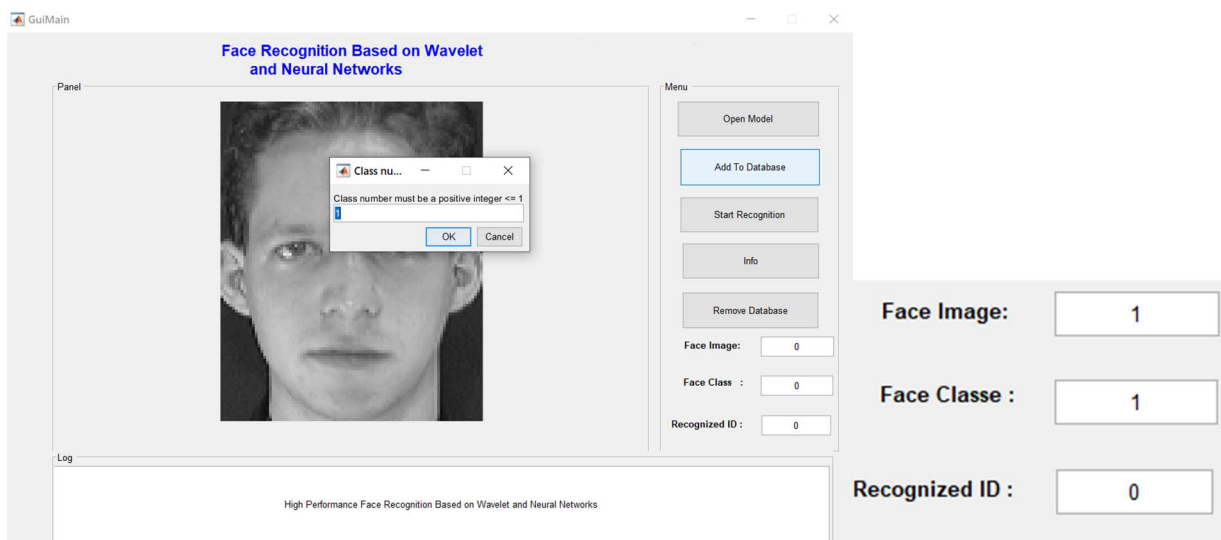
ДОДАТОК А

Приклад роботи запропонованої у розділі 3.3 нейронної мережі з базою даних для розпізнавання

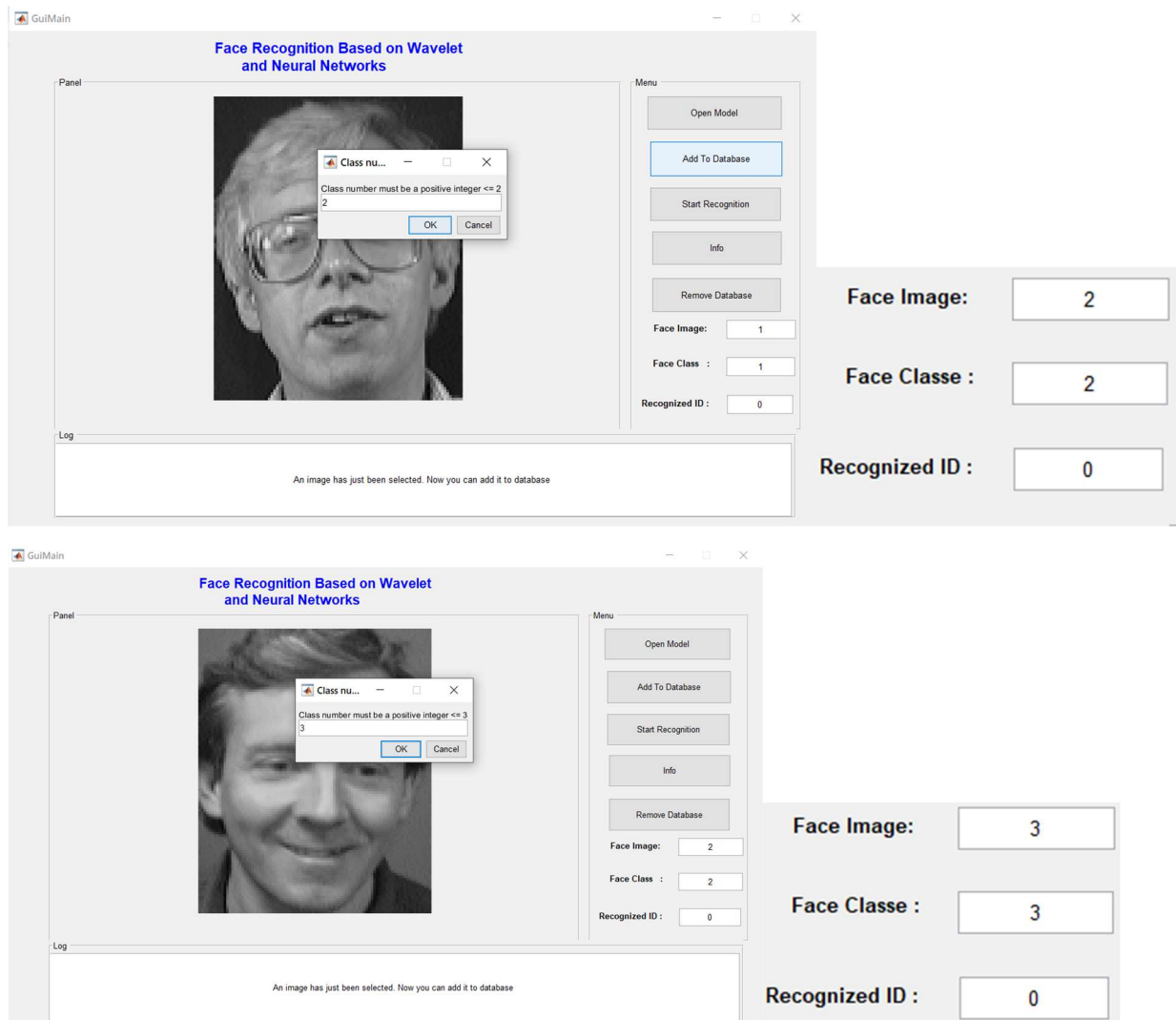
1. Спочатку відкриємо компілятор:



2. Потім внесемо в базу даних перше зображення лиця, присвоївши йому тип лиця 1 (Face Class):

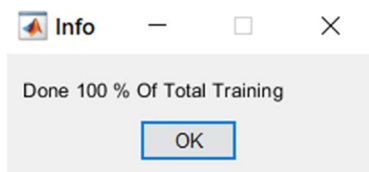


3. Такі ж дії проведемо з другим і третім лицем:

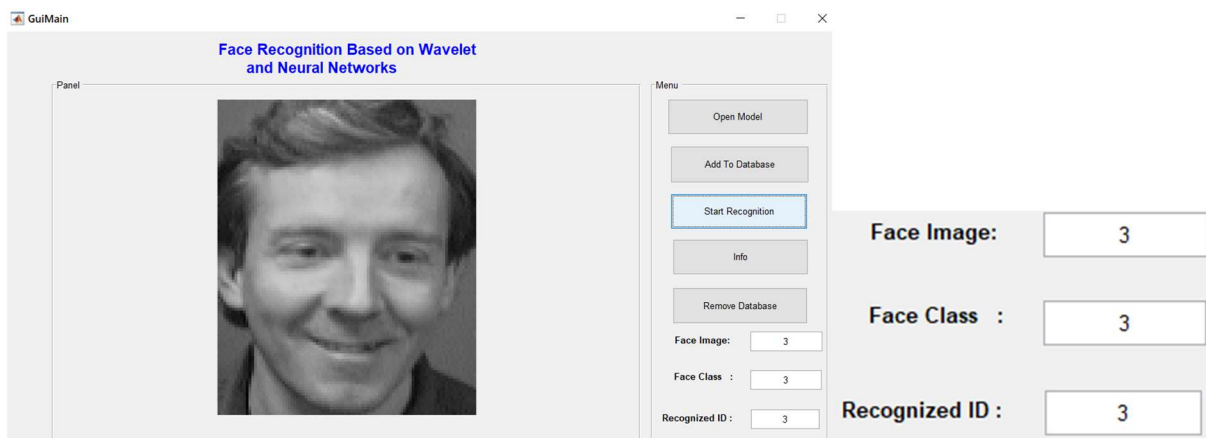


Варто зауважити, що в базу даних можна додавати скільки завгодно зображень лиць, програма може працювати з великими обсягами даних.

4. Коли ми додали потрібну кількість лиць до бази даних, обираємо зображення лиця, яке потрібно знайти серед бази даних і натискаємо «Почати впізнавання» («Start Recognition»). Треба дочекатись наступного повідомлення про завершення навчання:



5. Компілятор видає ID зображення, що знайшов у процесі виконання програми.



Ім'я користувача:
Комп'ютерної математики та інформаційної безпеки...

ID перевірки:
1016361177

Дата перевірки:
14.06.2024 19:01:49 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
14.06.2024 19:17:22 EEST

ID користувача:
100005746

Назва документа: Диплом_Ляшко

Кількість сторінок: 69 Кількість слів: 10681 Кількість символів: 70194 Розмір файлу: 3.28 MB ID файлу: 1016166076

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

20%
Схожість

Найбільша схожість: 15.4% з Інтернет-джерелом (http://geometry.karazin.ua/resources/documents/20191108180030_b47).

17.9% Джерела з Інтернету

144

Сторінка 71

6.08% Джерела з Бібліотеки

271

Сторінка 72

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0%
Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи

134

Підозріле форматування

20
сторінок