МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Київський національний університет будівництва і архітектури

Конспект лекцій
з дисципліни «Інформаційні технології представлення, обробки та розпізнавання зображень»

для здобувачів першого (бакалаврського) рівня вищої освіти
за спеціальністю 126 «Інформаційні системи та технології»
освітньої програми «Інформаційні системи та технології»

д.т.н., проф. Терейковська Л.О.

Київ 2025

Змістовний модуль 1. Представлення та обробка зображень

Лекція 1. Вступ до комп'ютерної обробки та розпізнавання зображень

Питання лекції. Предмет та завдання комп'ютерної обробки та розпізнавання зображень. Поняття сигналу та зображення. Поняття цифрової обробки сигналу. Кольорові простори. Дискретизація та квантування зображень. Похибки дискретного представлення зображень.

Мета дисципліни «Інформаційні технології представлення, обробки та розпізнавання зображень» полягає у придбанні студентами теоретичних знань, навичок та досвіду вирішення практичних задач побудови та експлуатації засобів обробки та розпізнавання зображень.

Комп’ютерна обробка зображень - це сукупність методів і технологій, які дозволяють здійснювати аналіз, перетворення, покращення, зберігання та інтерпретацію зображень за допомогою комп’ютерних засобів.

Основні завдання:

- Покращення візуальної якості зображення (фільтрація шуму, контрастування).
- Стиснення зображень для зберігання та передавання.
- Виділення контурів, об'єктів, текстур.
- Сегментація зображення (розбиття на регіони).
- Автоматичне розпізнавання об’єктів (людей, знаків, дефектів тощо).
- Побудова систем машинного зору (у робототехніці, охороні, медичній діагностиці тощо).

В загальному випадку під поняттям цифровий сигнал розуміють форму передачі інформації, що подається у вигляді дискретних значень [2, 3, 10]. У контексті цифрового сигналу, цифрове зображення – це сигнал, що описує візуальну інформацію та може бути переданий, оброблений та збережений у цифровому форматі. По способу представлення графічних об'єктів цифрові

зображення можливо класифікувати на векторні та растрові. Векторне зображення – це графічне представлення, яке створюється та зберігається у вигляді математичних формул, що описують геометричні примітиви, такі як лінії, криві, багатокутники та точки. На відміну від векторного, растрове зображення представляє собою множину окремих елементів – пікселів. Надалі в даному посібнику увага зосереджена на растрових зображеннях.

До основних термінів, що описують растрове зображення відносять:

- Піксель – мінімально можливий елемент зображення.

- Роздільна здатність – характеризує кількість пікселів по горизонталі та вертикалі на одиницю довжини зображення. Роздільна здатність вимірюється в кількості пікселів на одиницю площі зображення. Чим вища роздільна здатність, тим більш детальніше відображається зображення.

- Кольорова модель – визначає, як відображаються кольори на цифровому зображенні. Одна із найбільш поширених кольорових моделей це модель RGB, що передбачає формування кольору кожного із пікселів шляхом комбінації червоного, зеленого та синього кольорів. При використанні монохромної моделі окремий піксель може мати або чорний або білий колір. При використанні напівтонового зображення колір пікселя відображається у відтінках сірого кольору.

- Глибина кольору визначається кількістю бітів, які використовуються для представлення кольору одного пікселя. Наприклад, для напівтонового зображення при глибіні кольору 8-біт колір пікселю може відображати 256 різних відтінків сірого. При використанні кольорової моделі RGB при глибіні кольору 24-біти, піксель може відображати більше 16 мільйонів кольорів.

- Формат файлу зображення – спосіб зберігання даних цифрового зображення в комп'ютерній системі, який визначає, як інформація про пікселі, кольори та інші характеристики зображення кодується, стискається і зберігається у файл. Кожен формат має свої особливості, що використовуються в певній прикладній області.

До найбільш поширених форматів зображення відносять JPEG, PNG, GIF, BMP, TIFF. Розглянемо особливості цих форматів. JPEG - використовується для стиснення зображень з частковою втратою якості. Використання цього формату дозволяє зменшити розмір файлу зображення за рахунок видалення малопомітних деталей. PNG - підтримує стиснення зображення без втрат та забезпечує прозорість фону. GIF - підтримує анімацію та обмежує кількість кольорів до 256. BMP - нестиснений формат зображення, при якому зберігаються параметри всіх пікселів. TIFF - використовується для зображень високої якості та підтримує як стиснення без втрат, так і без стиснення. Цифрова обробка сигналів (ЦОС) - це процес математичної обробки дискретних сигналів із застосуванням алгоритмів, що реалізуються на комп'ютерах або спеціалізованих пристроях. У випадку зображень це:

- фільтрація (пригнічення шуму);
- згладжування або підсилення контрасту;
- перетворення Фур'є для аналізу частотного складу;
- побудова гістограм, морфологічні перетворення.

Ціль ЦОС - покращити якість, підготувати дані до розпізнавання, зменшити обсяг інформації при збереженні її суті.

Кольоровий простір - це математична модель, яка описує представлення кольору у вигляді векторів з кількома компонентами.

Основні кольорові простори:

Простір	Компоненти	Призначення
RGB	Червоний, зелений, синій	Базовий для дисплеїв, камер
HSV	Відтінок (Hue), насиченість (Saturation), яскравість (Value)	Зручний для фільтрації за кольором
YCbCr	Яскравість + 2 кольорові компоненти	Використовується в відео/TV
Lab	Яскравість (L), колірні координати (a, b)	Близький до людського сприйняття

Застосування: перехід між просторами дозволяє виділяти об'єкти за кольором або освітленням, підвищувати точність класифікації, стискати зображення.

Дискретизація - це процес перетворення безперервного простору зображення у дискретну сітку. Результат: розбиття зображення на пікселі (елементи з фіксованими координатами).

Квантування - це процес наближення значень яскравості (або кольору) до обмеженого набору рівнів. Наприклад, при 8-бітному квантуванні доступно 256 рівнів інтенсивності (0–255). Приклад:

- Фото з роздільністю 1920×1080 (Full HD) → 2,073,600 пікселів.
- Кожен піксель може мати 3 байти (по 1 байту на R, G, B).

Чим вища роздільність і глибина квантування - тим точніше представлення зображення, але більші обсяги даних.

Дискретизація і квантування неминуче призводять до втрат точності.

Основні похибки:

- Аліасінг (aliasing): спотворення при недостатній роздільній здатності (наприклад, зламані лінії, “сходинки” на діагоналях).

- Квантова похибка: похибка через округлення інтенсивності до найближчого рівня (проявляється як "сходинки" або “шум” при плавних переходах).

- Псевдокольори: спотворення кольору при переведенні в інший простір з меншою кількістю рівнів.

- Втрати через стиснення: при використанні JPEG або інших методів із втратами.

Способи зменшення похибок:

- підвищення роздільності;

- згладжування (антиаліасінг);

- використання безвтратного стиснення.

Лекція 2. Математичні моделі зображень

Питання лекції. Моделі неперервних зображень. Просторові спектри зображень. Спектральні інтенсивності зображень. Імовірнісні моделі зображень та функції автокореляції. Критерії якості зображень.

У реальному світі зображення є неперервними функціями двох змінних - просторових координат x та y. Вони описуються математичною моделлю:

$$f(x,y) : \mathbb{R}^2 \to \mathbb{R}$$

де *f(x,y)* - яскравість (інтенсивність) точки з координатами (x,y).

Приклади моделей

Ідеалізоване зображення: гладка функція без шумів, викривлень, з періодичною або поліноміальною структурою.

Модель “зображення + шум”:

$$f(x,y) = s(x,y) + n(x,y)$$

де *s(x,y)* - корисний сигнал (структура зображення), *n(x,y)* - шум (випадкові перешкоди).

Такі неперервні моделі - основа для аналітичних методів аналізу, фільтрації та синтезу зображень.

Просторові спектри зображень

Для аналізу зображень у частотній області використовують перетворення Фур'є. Неперервне двовимірне перетворення Фур’є:

$$F(u,v) = \iint f(x,y) e^{-j2\pi(ux+vy)} dxdy$$

де *F(u,v)* - спектральне представлення зображення,

u, v - частотні координати (горизонтальна та вертикальна частоти),

j - уявна одиниця.

Інтерпретація: спектр *F(u,v)* показує, які просторові частоти домінують у зображенні: низькі частоти - великі області однакового кольору (фони); високі

частоти - деталі, краї, текстура.

Спектральний аналіз дозволяє: застосовувати фільтрацію (low-pass, high-pass); виділяти текстурні компоненти; зменшувати шум.

Імовірнісні моделі зображень та функції автокореляції

Зображення можна розглядати як реалізацію випадкового поля, тобто кожен піксель - випадкова величина:

f(x,y) = випадкова змінна з певним розподілом

Це підхід особливо важливий у задачах: аналізу шумів, синтезу текстур, розпізнавання образів. Функція автокореляції:

$$R(\Delta x, \Delta y) = \mathbb{E}[f(x,y) \cdot f(x+\Delta x, y+\Delta y)]$$

Це математичне очікування добутку значень яскравості в парі пікселів на певній відстані. Інтерпретація:

Якщо $R(\Delta x,\Delta y)$ велике - є регулярність (повторення, текстура).

Якщо функція швидко спадає - зображення випадкове або має мало структури.

Застосування: виявлення періодичних структур (тканин, фасадів будівель); побудова моделей шуму; адаптивна фільтрація.

Критерії якості зображень. Оцінка якості зображення - важлива частина обробки, особливо після стиснення, передавання каналом зв'язку, фільтрації або реконструкції. Об'єктивні критерії (математичні):

Критерій	Формула / принцип	Призначення
MSE (Mean Squared Error)	$\frac{1}{N}\sum(f-\hat{f})^2$	Середня квадратична похибка
PSNR (Peak Signal-to-Noise Ratio)	$\mathrm{PSNR} = 10\log_{10}\left(\frac{MAX^2}{MSE}\right)$	Децибели відношення сигнал/шум
SSIM (Structural Similarity Index)	Комплексна метрика (структура, контраст, яскравість)	Більш наближено до людського сприйняття
Entropy	Рівень невизначеності (в інформаційному сенсі)	Відображає складність або деталізацію

Лекція 3. Цифрова обробка зображень шляхом поелементних перетворень

Питання лекції. Модифікація яскравості зображень. Лінійне контрастування зображень. Соляризація зображення. Зональне контрастування зображення. Порогова обробка напівтонових зображень. Поелементна обробка кольорових зображень.

Яскравість зображення – це характеристика, що показує, наскільки світлим чи темним виглядає зображення загалом чи його окремі елементи. Для фізичних об'єктів яскравість співвідноситься з кількістю світла, що випромінює або відбиває об'єкт, і сприймається людським оком. Яскравість відіграє важливу роль у сприйнятті зображення людиною, а тому це один із ключових параметрів, що використовується для оцінки якості та при обробці зображення.

В якості основної одиниці виміру яскравості реального об'єкта використовують кандела на квадратний метр (cd/m²). Зазначимо, що кандела - це одиниця виміру сили світла. Одна кандела відповідає силі світла, що випромінюється однією восковою свічкою. В контексті цифрового зображення яскравість пікселя представляється як числове значення, що визначає рівень інтенсивності його кольору.

Для монохромних зображень яскравість пікселя може приймати два значення (0 або 1), що відповідає чорному та білому кольору.

Для одноканального 8-бітного зображення у відтінках сірого яскравість пікселя варіюється від 0 (чорний колір) до 255 (білий колір).

Для трьохканальних зображень у форматі RGB яскравість пікселя визначається за допомогою виразу виду [6, 7]:

$$I = 0{,}299 \cdot R + 0{,}587 \cdot G + 0{,}114 \cdot B,$$

де R, G, B – значення, що відповідає червоному, зеленому та блакитному кольору пікселя зображення.

Для зображення в цілому визначають максимальне, мінімальне та середнє значення яскравості. Для розрахунку вказаних параметрів

використовують вирази виду:

$$I_{max} = max(I_1, I_2, \dots, I_N),$$

$$I_{min} = min(I_1, I_2, \dots, I_N),$$

$$I_{avg} = \frac{1}{N}\sum_{n=1}^{N} I_n,$$

де I_{max}, I_{min}, I_{avg} – максимальне, мінімальне та середнє значення яскравості; N – кількість пікселів зображення; I_n – яскравість n-го пікселю.

Крім I_{max}, I_{min}, I_{avg}, при вирішенні практичних задач також розраховують максимальну, мінімальну та середню яскравість певної наперед заданої області зображення Ω, що позначаються як $I_{max}(\Omega)$, $I_{min}(\Omega)$, $I_{avg}(\Omega)$. Для розрахунку цих показників використовуються вирази, аналогічні до вказаних виразів. Також при вирішенні практичних задач використовуються показники, що відповідають інтенсивності кольору кожного із кольорових каналів зображення в цілому або заданої області даного зображення:

$$D_{X,max} = max\left(D_{X,1}, D_{X,2}, \dots, D_{X,N}\right),$$

$$D_{X,min} = min\left(D_{X,1}, D_{X,2}, \dots, D_{X,N}\right),$$

$$D_{X,avg} = \frac{1}{N}\sum_{n=1}^{N} D_{X,n},$$

де $D_{x,max}$, $D_{x,min}$, $D_{x,avg}$ – максимальне, мінімальне та середнє значення інтенсивності кольору зображення в каналі X; N – кількість пікселів зображення; $D_{X,n}$ – інтенсивність кольору в каналі X для n-го пікселю.

Одним із основних засобів, що використовується для інтегральної оцінки яскравості зображення, являється гістограма інтенсивності, що забезпечує графічне представлення розподілу яскравості пікселів зображення. Іншими словами, гістограма яскравості відображає скільки пікселів в

зображенні мають кожне із можливих значень яскравості. На гістограмі по горизонтальній осі X відкладаються значення яскравості пікселів. Можливий діапазон значень по осі X визначається в залежності від умов поставленої практичної задачі. Для напівтонових зображень з глибиною кольору 8 біт, як правило, вказаний діапазон має межі від 0 до 255. Чорному кольору відповідає 0, білому кольору - 255. На вертикальній осі Y відкладається кількість пікселів з відповідним значенням яскравості. В деяких випадках відкладається не абсолютна кількість пікселів, а відносна. Чим вищий стовпчик на гістограмі, тим більше пікселів у зображенні з відповідним значенням яскравості. Приклад гістограми яскравості, побудованої за допомогою засобів MatLab для кольорового зображення, показано на рис. 1.

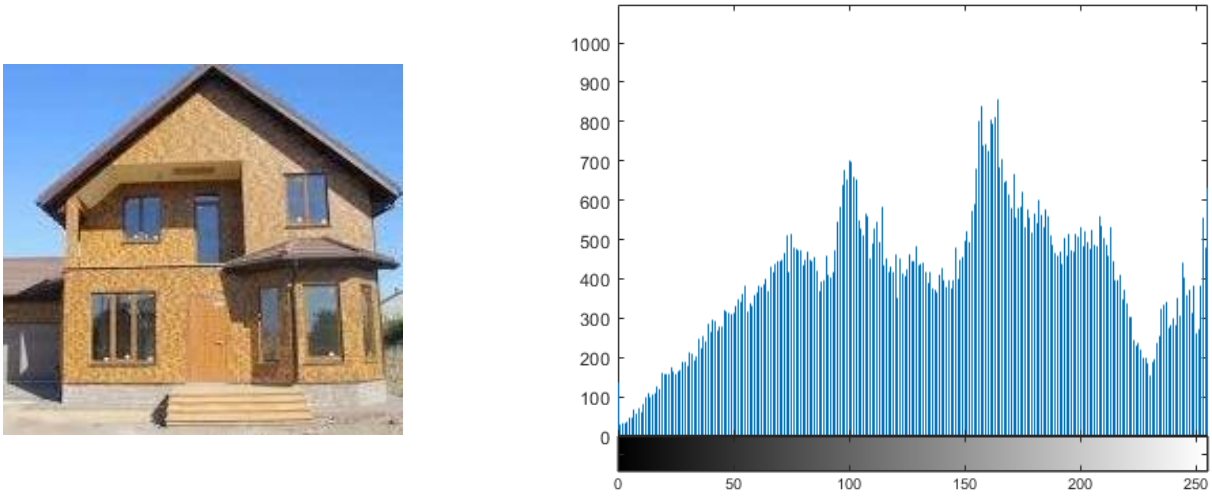

Рис. 1. Приклад гістограми для кольорового зображення

Використання засобів MatLab дозволило ілюструвати гістограму за рахунок відображення в нижній частині цієї діаграми розподілу кольорової гами, що відповідає діапазону можливих значень осі X.

Модифікація яскравості означає збільшення або зменшення інтенсивності всіх (або частини) пікселів:

$$f'(x,y) = f(x,y) + \Delta$$

де f(x,y) - початкове значення пікселя,

Δ - додатне (збільшує яскравість) або від'ємне значення.

Лінійне контрастування зображень. Контраст - це різниця між найяскравішими та найтемнішими частинами зображення. Лінійне контрастування (stretching) - масштабування інтенсивностей.

$$f'(x,y) = \alpha \cdot f(x,y) + \beta$$

$\alpha > 1$: підвищення контрасту,

$\alpha < 1$: зменшення контрасту,

β: коригування яскравості.

Автоматичне контрастування: розтягується гістограма інтенсивностей у повний діапазон (0–255):

$$f'(x,y) = \frac{f(x,y) - f_{\min}}{f_{\max} - f_{\min}} \cdot 255$$

Соляризація - це інверсія яскравостей після певного порогу, що імітує ефект плівкової фотографії. Принцип: яскраві ділянки стають темними, а темні - світлими. Типова функція:

$$f'(x,y) = \begin{cases} f(x,y), & f(x,y) < T \\ 255 - f(x,y), & f(x,y) \geq T \end{cases}$$

В результаті зображення набуває абстрактного вигляду з незвичними світлотінями.

Зональне контрастування зображення. Цей метод дозволяє посилити або послабити контраст у визначених діапазонах яскравості (зонах). Принцип: зображення розбивається на зони (наприклад, 0–64, 65–128, 129–192, 193–255). Для кожної зони застосовуємо свою лінійну трансформацію:

$$f'(x,y) = \begin{cases} a_1 f(x,y) + b_1, & f \in Z_1 \\ a_2 f(x,y) + b_2, & f \in Z_2 \\ \dots \end{cases}$$

Застосовується для покращення видимості окремих деталей (наприклад, тіней або яскравих плям).

Порогова обробка напівтонових зображень. Порогова обробка - це перетворення напівтонового зображення в бінарне, де кожен піксель стає або чорним, або білим.

$$f'(x,y) = \begin{cases} 0, & f(x,y) < T \\ 255, & f(x,y) \geq T \end{cases}$$

Методи вибору порогу:

- Фіксований поріг *T*.

- Метод Отсу - автоматичний вибір *T*, який мінімізує внутрішньокласову дисперсію.

- Адаптивна порогова обробка - індивідуальний поріг для кожної області зображення.

Застосування: виділення об’єктів; розпізнавання тексту (OCR); підготовка до морфологічних операцій.

Поелементна обробка кольорових зображень. Методи поелементної обробки базуються на постулаті про те, що значення параметрів окремих пікселів не залежать одне від іншого. Тому при використанні цих методів результат обробки окремого пікселя зображення залежить лише від значень параметрів даного пікселя. Приклади:

- Лінійна модифікація:

$$R' = a \cdot R + b, \quad G' = a \cdot G + b, \quad B' = a \cdot B + b$$

- Індивідуальна корекція каналів (наприклад, підсилення червоного):

$$R' = R + 50, \quad G' = G, \quad B' = B$$

- Перетворення в інші колірні простори (HSV, Lab) і модифікація яскравості чи насиченості.

Застосування: художні ефекти; баланс білого; корекція кольору в фото та відео.

Лекція 4. Фільтрація зображень

Питання лекції. Лінійна та нелінійна просторова фільтрація. Фільтри підвищення верхніх просторових частот зображення. Дискретне перетворення Фур'є. Згортка. Низькочастотні фільтри. Високочастотні фільтри.

Лінійна просторова фільтрація передбачає, що нове значення пікселя обчислюється як зважена сума сусідніх пікселів. Загальна формула згортки:

$$g(x,y) = \sum_{i=-k}^{k} \sum_{j=-k}^{k} w(i,j) \cdot f(x+i, y+j)$$

де *f(x,y)* - оригінальне зображення,

w(i,j) - маска фільтра (ядро),

g(x,y) - фільтроване зображення.

Приклади лінійних фільтрів: середнє згладжування (блюр) - зменшення шуму; фільтр Лапласа - підсилення контурів; фільтр Собеля - детектування градієнтів.

Математичне забезпечення виявлення перепадів яскравості зображення в точці з координатами (i,j) за Собелем складають вирази:

$$D_x = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix},$$

$$D_y = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix},$$

$$S_x = D_x * f,$$

$$S_y = D_y * f,$$

$$S = \sqrt{S_x^2 + S_y^2},$$

де f – початкове зображення; D_x, D_y – маски фільтру Собеля; S –

відфільтроване зображення з виділеними кордонами об’єктів; * – операція згортки.

Розрахунок результату згортки матриці, що відповідає початковому зображенню f, з матрицями, що відповідають маскам фільтру Собеля D_x та D_y , реалізується за допомогою виразів виду:

$$S_x(i,j) = (-1)\cdot f(i-1,j-1) + (-2)\cdot f(i-1,j) + (-1)\cdot f(i-1,j+1) + (0)\cdot f(i,j-1) + (0)\cdot f(i,j) + (0)\cdot f(i,j+1) + (1)\cdot f(i+1,j-1) + (2)\cdot f(i+1,j) + (1)\cdot f(i+1,j+1) = -f(i-1,j-1) - 2\cdot f(i-1,j) - f(i-1,j+1) + f(i+1,j-1) + 2\cdot f(i+1,j) + f(i+1,j+1)$$

$$S_y(i,j) = (-1)\cdot f(i-1,j-1) + (0)\cdot f(i-1,j) + (1)\cdot f(i-1,j+1) + (-2)\cdot f(i,j-1) + (0)\cdot f(i,j) + (2)\cdot f(i,j+1) + (-1)\cdot f(i+1,j-1) + (0)\cdot f(i+1,j) + (1)\cdot f(i+1,j+1) = -f(i-1,j-1) - 2\cdot f(i,j-1) - f(i+1,j-1) + f(i-1,j+1) + 2\cdot f(i,j+1) + f(i+1,j+1)$$

Нелінійна фільтрація. На відміну від лінійної, нелінійна фільтрація не використовує лінійну комбінацію пікселів. Прикладом може бути медіанна фільтрація, коли піксель замінюється медіаною значень в околі (ефективне для видалення імпульсного шуму ("сіль і перець")).

Фільтри підвищення верхніх просторових частот зображення. Низькі частоти - повільні зміни (однорідні області, фон). Високі частоти - різкі переходи (грані, контури, шум). Фільтри підвищення високих частот виділяють деталі, контури, швидкі зміни яскравості. Приклади:

1. Фільтр Лапласа

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

2. Фільтр підвищення різкості (unsharp masking)

$$g(x,y) = f(x,y) + \alpha\cdot(f(x,y) - \mathrm{blur}(f(x,y)))$$

Дискретне перетворення Фур'є (ДПФ). Призначення: ДПФ дозволяє представити зображення у частотній області, де кожна точка відповідає певній просторовій частоті. Представлення 2D-ДПФ:

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) \cdot e^{-2\pi i \left(\frac{ux}{M} + \frac{vy}{N}\right)}$$

f(x,y) - зображення в просторі,

F(u,v) - спектр у частотній області.

Властивості: центр спектра - низькі частоти; краї - високі частоти.

Згортка (Convolution). Згортка - це базова операція в обробці зображень, що поєднує ядро (фільтр) із зображенням для створення нового образу. Використовується в згладжуванні, виділенні контурів, детектуванні шумів. У частотній області згортка в просторі відповідає множенню спектрів.

Низькочастотні фільтри - згладжують зображення, зменшуючи високочастотні компоненти. Приклади: середнє згладжування; Гаусів фільтр; гладке згладжування без різких переходів; ідеальний низькочастотний фільтр: пропускає всі частоти нижче порогового значення D_0.

Високочастотні фільтри - виділяють контури, краї та інші деталі. Приклади: фільтр Лапласа (високочастотний у просторі); ідеальний високочастотний фільтр: пропускає частоти вище порогового значення D_0; Butterworth та Gaussian високочастотні фільтри - використовуються для м'якого виділення високих частот без різких артефактів.

Лекція 5. Використання двохвимірних вейвлетів для цифрової обробки зображень

Питання лекції. Особливості вейвлет-перетворень зображень. Необхідність застосування двохвимірних вейвлетів. Алгоритм створення двохвимірного вейвлету. Методика дискретного вейвлет-перетворення зображення.

Вейвлет-перетворення (Wavelet Transform, WT) - це математичний інструмент, який дозволяє аналізувати сигнал (зображення) на різних масштабах (масштабне декомпонування), зберігаючи інформацію про локалізацію в часі (або в просторі для зображень). На відміну від перетворення Фур'є, яке працює лише у частотній області, вейвлет-перетворення забезпечує одночасну локалізацію як у просторі, так і у частоті. Переваги вейвлет-перетворень: здатність до стиснення зображень (JPEG 2000); ефективне виявлення особливостей на різних масштабах; якісна обробка країв, кутів; можливість адаптивної фільтрації шумів. Оскільки зображення - це двовимірні сигнали, для їх ефективного аналізу потрібно використовувати двовимірні вейвлет-перетворення, які обробляють просторову інформацію по горизонталі та вертикалі.

Розглянемо метод застосування вейвлет-перетворень для фільтрації цифрового зображення, що по суті є дискретним двовимірним сигналом. Зазначимо, що вказані цифрові напівтонові або кольорові зображення, які пристосовані до відображення за допомогою сучасних розповсюджених комп'ютерних засобів описуються за допомогою двовимірної прямокутної або квадратної матриці. Тому для аналізу таких сигналів прийнято використовувати двовимірне дискретне вейвлет-перетворення, яке є композицією одновимірних дискретних вейвлет-перетворень. На кожному етапі деталізації спочатку одновимірні дискретні вейвлет-перетворення застосовуються для кожного рядка двовимірної матриці вхідних даних. Після цього виконуються одновимірні вейвлет-перетворення для кожного стовпця отриманої матриці. Математичний апарат для розрахунку матриці вейвлет-

коефіцієнтів одного рядка пікселів кольорового зображення визначається виразами виду:

$$\begin{cases} \boldsymbol{W}_{m,k} = \dfrac{1}{\sqrt{a_0^m}} \displaystyle\sum_{n=0}^{N-1} \left(q(x_n)\varphi^*(a_0^m x_n - k)\right), \\ 1 \le m, k \le N \end{cases}$$

$$q = 0{,}299R + 0{,}587G + 0{,}114B,$$

де a_0 – початковий масштаб; k – параметр зсуву; m – параметр масштабу; x_n – координата n-ої точки зображення; * – операція комплексного спряження; N – розмір зображення; $q(x_n)$ – величина закодованого кольору зображення в точці x_n; φ – базисний вейвлет; R,G, B – компоненти кожного із каналів кольору в форматі RGB для окремого пікселя.

Зазначимо, що ці вирази використовуються у випадку, коли при обробці зображення його кольором можливо знехтувати. У випадку, коли це реалізувати недоцільно, вейвлет-коефіцієнти розраховуються окремо для кожного із кольорових каналів. Відповідний математичний апарат можливо записати так:

$$\begin{cases} \boldsymbol{W}_{m,k}(i) = \dfrac{1}{\sqrt{a_0^m}} \displaystyle\sum_{n=0}^{N-1} \left(q(x_n, i)\varphi^*(a_0^m x_n - k)\right), \\ 1 \le m, k \le N \end{cases}$$

де $q(x_n, i)$ – інтенсивність і-го каналу кольору в точці x_n.

При використанні RGB-моделі канали співвідносяться з червоним, зеленим та блакитним кольором.

Власне процедура фільтрації полягає у вилученні визначених компонент матриці $\boldsymbol{W}_{m,k}$. В більшості випадків вилучаються ті компоненти матриці $\boldsymbol{W}_{m,k}$, для яких параметр зсуву та/або параметр масштабу виходять за межі наперед визначеного діапазону. Після цього компоненти відфільтрованої матриці $\boldsymbol{W}_{m,k}$ використовуються в процедурі оберненого вейвлет-перетворення для відновлення очищеного зображення. Для напівтонового зображення використовується вираз (1), для багатокольорового зображення – вираз (2).

$$q(x_n) = \frac{\pi}{ln(a_0)} \sum_{n=0}^{N-1} \sum_{k=0}^{N-1} (\varphi^*(x_n) \boldsymbol{W}_{m.k}), \quad (1)$$

$$q(x_n, i) = \frac{\pi}{ln(a_0)} \sum_{n=0}^{N-1} \sum_{k=0}^{N-1} \left(\varphi^*(x_n) \boldsymbol{W}_{m.k}(i)\right), \quad (2)$$

де i – номер кольорового каналу.

Базуючись на загальноприйнятих вимогах, які забезпечують ефективність вейвлет-перетворень, визначено доцільність використання діадного дискретного вейвлет-перетворення, що за рахунок зменшення кількості копій базисного вейвлету дозволяє зменшити обчислювальну складність процедури обчислення вейвлет-коефіцієнтів. Застосування діадного дискретного вейвлет-перетворення призводить до модифікації вищенаведених виразів:

$$\begin{cases} \boldsymbol{W}_{m,k} = \dfrac{1}{\sqrt{2^m}} \displaystyle\sum_{n=0}^{N-1} \left(q(x_n) \varphi^*(2^m x_n - k)\right), \\ 1 \le m, k \le N \end{cases}$$

$$\begin{cases} \boldsymbol{W}_{m,k}(i) = \dfrac{1}{\sqrt{2^m}} \displaystyle\sum_{n=0}^{N-1} \left(q(x_n, i) \varphi^*(2^m x_n - k)\right), \\ 1 \le m, k \le N \end{cases}$$

$$q(x_n) = \frac{\pi}{ln(2)} \sum_{n=0}^{N-1} \sum_{k=0}^{N-1} (\varphi^*(x_n) \boldsymbol{W}_{m.k}),$$

$$q(x_n, i) = \frac{\pi}{ln(2)} \sum_{n=0}^{N-1} \sum_{k=0}^{N-1} \left(\varphi^*(x_n) \boldsymbol{W}_{m.k}(i)\right).$$

Труднощі створення ефективних фільтрів багато в чому пов’язані зі складністю розрахунку меж діапазонів, які визначають перелік компонентів матриці $\boldsymbol{W}_{m,k}$, котрі підлягають вилученню. Тому розробка ефективних

методів фільтрації обов'язково передбачає врахування особливостей прикладної області застосування.

Розглянемо можливість застосування вейвлет-перетворень для фільтрації типових завад, які є характерними для зображень асоційованих з біометричними параметрами людини, зареєстрованими за допомогою відеокамер систем дистанційного навчання. В системах дистанційного навчання типові завади, що виникають на зареєстрованих зображеннях, в основному викликані: наявністю предметів, що закривають частину зображення; нерівномірністю освітлення, що призводить до відблисків на зображенні. При цьому в багатьох випадках локалізація та інші параметри типових завад мають нестаціонарний динамічний характер. Для прикладу, на рис.1 показано спотворені під впливом типових завад зображення обличчя та зображення райдужної оболонки ока.

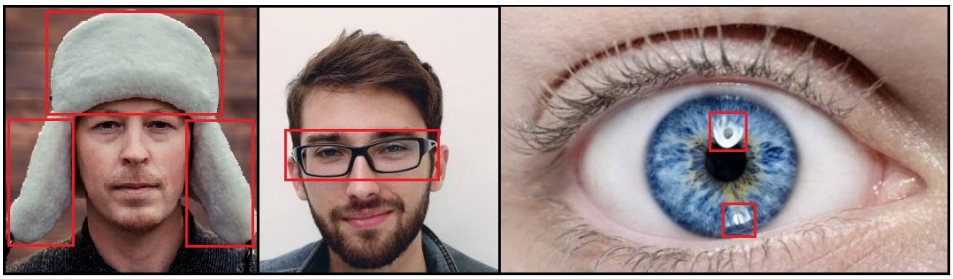

Рис. 1. Спотворені під впливом типових завад зображення обличчя та зображення райдужної оболонки ока

Динамічний характер локалізації однієї або декількох завад дозволяє стверджувати, що в кожен окремий момент часу на зареєстрованому зображенні можуть бути як викривлені, так і невикривлені частини зображення, котре асоційоване з біометричним параметром. При цьому місцеположення цих частин змінюється. Це дозволяє стверджувати, що за рахунок порівняння зображень, зареєстрованих у різні моменти часу, можливо відтворити незашумлене зображення.

Розглянемо метод фільтрації біометричних параметрів, що асоціюються з двовимірними кольоровими або напівтоновими зображеннями. В базовому випадку реалізація методу полягає у виконанні наступних етапів:

1. Провести реєстрацію відеоряду, що відповідає послідовному відображенню зображень, котрі асоціюються з певним біометричним параметром.

2. Визначити тип зображення, що підлягає фільтрації. Мається на увазі необхідність аналізу кольорового, напівтонового (з одним кольоровим каналом) або чорно-білого (монохромного) зображення.

3. Визначити перелік підконтрольних зображень відеоряду, що відповідають зображенням, у яких викривлення локалізовані у різних частинах.

4. Визначити перелік типових завад, що притаманні певному біометричному параметру, асоційованому з зображенням.

5. Реалізувати вибір найбільш ефективного базисного вейвлету.

6. Визначити необхідний рівень деталізації вейвлет-перетворень.

7. Розрахувати значення вейвлет-коефіцієнтів кожного із підконтрольних зображень.

8. Обрати принцип інтеграції вейвлет-коефіцієнтів.

9. Попарно інтегрувати розраховані значення вейвлет-коефіцієнтів між собою.

10. Використовуючи отримані значення вейвлет-коефіцієнтів провести реалізацію зворотнього вейвлет-перетворення, що призводить до отримання незашумленого зображення.

Найбільш ефективні типи базисних вейвлетів: HAAR, MHAT, WAVE, FHAT, LP, Daubechies wavelet, Morlet.

Змістовий модуль 2. Розпізнавання зображень

Лекція 6. Методи та засоби розпізнавання зображень

Питання лекції. Загальний аналіз алгоритмів розпізнавання зображень. Застосування перетворення Хафа та ознак Хаара для розпізнавання зображень. Методи контурного аналізу. Програмні бібліотеки для розпізнавання зображень.

Загальний аналіз алгоритмів розпізнавання зображень. Розпізнавання зображень - це процес автоматичної ідентифікації об'єктів або патернів на зображеннях з використанням обчислювальних методів. Типові задачі:

- Розпізнавання об'єктів (люди, автомобілі, знаки).
- Класифікація зображень.
- Виявлення та сегментація об'єктів.
- Визначення форми, положення, розміру.

Основні етапи алгоритмів розпізнавання:

- Попередня обробка: фільтрація, бінаризація, нормалізація.
- Виділення ознак: знаходження контурів, кутів, текстур, фрагментів.
- Формування ознакового простору: векторизація, зменшення розмірності.
- Класифікація: на основі машинного навчання або правил.
- Інтерпретація результатів.

Класифікація методів розпізнавання:

- Алгоритмічні методи: евристики, жорсткі правила.
- Математичні методи: шаблонне співставлення, PCA, Hough.
- Методи машинного навчання: SVM, Decision Trees, k-NN.
- Методи глибинного навчання: CNN, ResNet, YOLO тощо.

Застосування перетворення Хафа та ознак Хаара для розпізнавання зображень. Перетворення Хафа (Hough Transform) – метод для виявлення геометричних форм (прямих, кіл, еліпсів) на зображеннях, навіть якщо вони частково приховані або зашумлені. Приклади використання: виявлення дорожньої розмітки; детекція кіл; аналіз архітектурних елементів (вікна, колони).

Ознаки Хаара (Haar-like features). Це набори прямокутних шаблонів, що описують відмінності яскравості між сусідніми областями зображення.

Властивості: швидкі у розрахунку завдяки інтегральному зображенню; використовуються у реальному часі для виявлення об'єктів.

Застосування: один із найвідоміших алгоритмів - Viola-Jones Face Detector, що використовує ознаки Хаара та каскад класифікаторів.

Типи ознак:

- Два сусідні прямокутники (горизонтальні/вертикальні).
- Трипрямокутні (напр. ніс).
- Чотирипрямокутні (виявлення кутів очей).

Методи контурного аналізу. Контурний аналіз (Contour Analysis) - це виявлення та аналіз меж об'єктів.

Методи виділення контурів:

1. Оператори градієнта: Sobel, Prewitt, Roberts.
2. Оператор Канні (Canny Edge Detector):
   - Шумозаглушення (фільтр Гауса).
   - Розрахунок градієнта.
   - Нехарактерна мінімізація.
   - Подвійна порогова обробка.

Аналіз контурів:

- Обчислення геометричних характеристик: площа, довжина.
- Моменти зображень: для обчислення центра ваги, кута обертання.
- Аналіз форми: стисліть, опуклість, компактність.

Практичне застосування: розпізнавання символів (OCR); виявлення дефектів на будівельних матеріалах; визначення форми об'єктів (напр. тріщини, порожнини).

Програмні бібліотеки для розпізнавання зображень

1. OpenCV (Open Source Computer Vision Library):

- Підтримка C++, Python.
- Функції: фільтрація, виявлення контурів, Hough Transform, Haar

Cascades, CNN.

- Швидка обробка в реальному часі.

2. scikit-image (Python):
    - Частина наукового стеку Python (разом із NumPy, SciPy).
    - Обробка зображень, фільтрація, сегментація, морфологія.
3. TensorFlow / Keras / PyTorch:
    - Підтримка глибокого навчання, розпізнавання образів із використанням CNN.
    - Гнучкість, підтримка GPU.
4. Dlib:
    - Виявлення та розпізнавання облич.
    - Підтримка векторів ознак, класифікаторів, обробки облич.
5. PIL / Pillow:
    - Просте завантаження, обробка, зміна розміру зображень.
    - Використовується спільно з іншими бібліотеками.

Лекція 7. Глибокі нейронні мережі при розпізнаванні зображень

Питання лекції. Передумови використання глибоких нейронних мереж. Особливості застосування згорткових нейронних мереж. Структура нейромережевої моделі LeNet-5. Розрахунок конструктивних параметрів класичної згорткової нейронної мережі.

Передумови використання глибоких нейронних мереж:

1. Зростання складності обробки зображень

Сучасні задачі обробки та розпізнавання зображень вимагають:

- виявлення об'єктів у складному фоні,
- класифікації об’єктів із великої кількості класів,
- роботи з великими наборами даних (Big Data),
- адаптивного навчання в умовах варіативності зображень (яскравість, масштаб, поворот, спотворення тощо).

Традиційні методи обробки зображень, засновані на фіксованих фільтрах, правилах чи ознаках, часто не справляються з цими задачами.

2. Еволюція машинного навчання

До появи глибоких нейронних мереж (deep neural networks, DNN):

- використовувалися штучні нейронні мережі з одним або двома шарами;
- були популярними методи SVM, k-NN, байєсівські класифікатори;
- ознаки витягували вручну (ручна інженерія ознак).

Але ці підходи: залежали від людського втручання; не масштабувалися до великої кількості класів; не були стійкими до варіацій зображень.

3. Поява глибокого навчання (Deep Learning)

Глибоке навчання дозволяє:

- автоматично витягувати релевантні ознаки без ручного втручання,
- будувати складні нелінійні залежності,
- ефективно працювати з великими обсягами даних.

Передумови ефективного застосування:

- Наявність великих датасетів (ImageNet, CIFAR, MNIST).

- Висока обчислювальна потужність (GPU, TPU).

- Покращені алгоритми навчання (Adam, RMSprop).

- Технології регуляризації (Dropout, BatchNorm).

Особливості застосування згорткових нейронних мереж

Згорткова нейронна мережа (Convolutional Neural Network, CNN) - це тип глибокої нейронної мережі, який спеціально призначений для обробки зображень. Основна ідея - автоматично виділяти ознаки з зображення через згортку (convolution), зберігаючи просторову структуру.

Основні компоненти CNN:

1. Згорткові шари (Convolutional layers): виконують згортку з фільтрами (ядрами), які навчаються, а також виявляють локальні патерни: краї, текстури.
2. Функції активації (ReLU): вносять нелінійність, підвищують здатність до апроксимації складних функцій.
3. Шари субдискретизації (Pooling): зменшують розмірність (MaxPooling або AveragePooling) та знижують чутливість до зсувів і спотворень.
4. Повнозв’язані шари (Fully Connected): з’єднують усі ознаки в один вектор та використовуються для прийняття остаточного рішення (класифікація).

Особливості CNN:

1. Локальна сприйнятливість: кожен нейрон бачить лише частину зображення.
2. Спільне використання ваг: один фільтр застосовується до всього зображення - зменшення кількості параметрів.
3. Ієрархічність ознак: перші шари - прості ознаки (лінії), глибші - складні (об'єкти, обличчя).

Переваги CNN в обробці зображень: висока точність у задачах розпізнавання образів; автоматичне виділення ознак; ефективність при великій кількості класів; можливість перенавчання (Transfer Learning).

*Приклади застосування CNN: р*озпізнавання облич (FaceNet, VGG-Face); виявлення об'єктів (YOLO, SSD); класифікація зображень (ResNet, Inception); медична діагностика (аналіз рентгенівських знімків); дефектоскопія

в будівництві (виявлення тріщин, корозії).

Структура нейромережевої моделі LeNet-5 показана на рис. 3

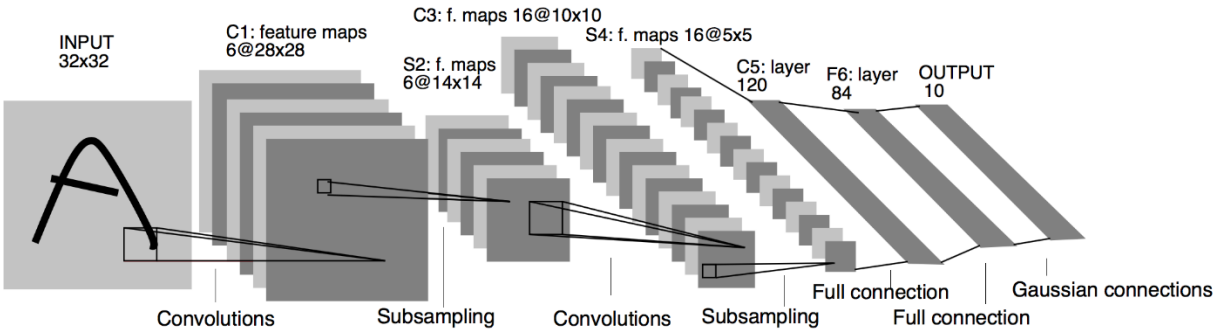

Рис. 3 Структурна схема Le-Net-5

Конструктивні параметри класичної згорткової нейронної мережі:

- Розмір вхідного поля - a_0.

- Кількість вхідних нейронів - L_{in}.

- Кількість вихідних нейронів - L_{out}.

- Кількість нейронів в повнозв'язному шарі - L_f.

- Кількість шарів згортки - K_{ls}.

- Кількість карт ознак в кожного шарі згортки - $L_{h,k}, k \in [1, K_{ls}]$.

- Кількість шарів підвибірки (субдискретизації) - K_{ld}.

- Масштабний коефіцієнт для кожного шару півибірки - $m_l, l \in [1, K_{ld}]$.

- Розмір *l*-го шару підвибірки розраховується так:

$$c_l = a_k / m_l ,$$

де a_k - розмір шару згортки, що передує l-му шару підвибірки.

- Розмір ядра згортки для кожного k-го шару згортки $(b \times b)_k, k \in [1, K_{ls}]$.

- Зміщення рецептивного поля при виконанні кожної k-ої процедури згортки $d_k, k \in [1, K_{ls}]$.

- Розмір карти ознак для кожного k-го шару згортки - $(a \times a)_k, k \in [1, K_{ls}]$.

$$a_k = (a_{k-1} - b_k + 2r_k)/d_k + 1,$$

де r_k - кількість доповнюючих нулів для k-го шару згортки.

- Структура зв’язків між сусідніми шарами згортки/підвибірки. Цю структуру можна представити у вигляді матриці:

$$Q_{i,i+1} = \begin{Vmatrix} q_{(i,1),(i+1,1)} & \cdots & q_{(i,1),(i+1,j)} \\ \cdots & \cdots & \cdots \\ q_{(i,G),(i+1,1)} & \cdots & q_{(i,G),(i+1,J)} \end{Vmatrix},$$

де G - кількість карт в і-ому шарі, J – кількість карт в (і+1)-ому шарі,

$Q_{i,i+1}$ - матриця, компоненти якої визначають наявність зв'язків між і-им та (і+1)-им схованими шарами,

$q_{(i,g),(i+1,j)}$ - компонент, що вказує на наявність/відсутність зв'язку між g-ою картою і-го шару та j-ою картою (і+1)-го шару,

$q_{(i,g),(i+1,j)} = 1$, якщо є зв'язок між g-ою картою і-го шару та j-ою картою (і+1)-го шару,

$q_{(i,g),(i+1,j)} = 0$, якщо немає зв'язку між g-ою картою і-го шару та j-ою картою (і+1)-го шару

Розрахунок конструктивних параметрів класичної ЗНМ:

Сумарний вхідний сигнал нейрона шару згортки:

$$x_k^{(i,j)} = f\left(x_{0,k} + \sum_{s=1}^{K}\sum_{t=1}^{K} w_{k,s,t}\, x^{((i-1)+s,(j+t))} \right),$$

де $x_k^{(i,j)}$ - вхідний сигнал (i,j)-го нейрона k-ої карти ознак,

$x_{0,k}$ - зміщення нейронів k-ої карти ознак,

K - розмір рецептивної області нейрона (розмір ядра згортки),

$w_{k,s,t}$ - ваговий коефіцієнт (s,t)-ого синаптичного зв'язку нейрона k-ої карти,

x - вихід нейрона попереднього шару.

Вихідний сигнал нейрона шару згортки:

$$y = \max(0, x).$$

Вихідний сигнал нейрона шару субдискретизації :

$$y^{(l)} = f\left(a^{(l)} \times subsample\left(y^{(l-1)}\right) + b^{(l)}\right) ,$$

де $y^{(l)}$- вихід l-го шару, $f()$ - функція активації, $a^{(l)}, b^{(l)}$ – коефіцієнти зміщення, $subsample()$ - операція вибірки максимальних значень.

Вихідний сигнал нейрона повнозв'язного шару:

$$y = \max(0, x)$$

Вихідний сигнал нейрона вихідного шару:

$$y_i = \frac{\exp(q_i)}{\sum_{k=1}^{L_{out}} \exp(q_k)}$$

де y_i - вихід і-го нейрона вихідного шару, q_k - сумарний вхідний сигнал для k-го нейрона вихідного шару, L_{out} - кількість вихідних нейронів.

Лекція 8. Підходи до підвищення ефективності згорткових нейронних мереж

Питання лекції. Підходи до усунення ефекту падіння градієнту. Підходи до розпізнавання об'єктів різних за розміром. Підходи до зменшення обчислювальної ресурсоємності.

Підходи до усунення ефекту падіння градієнту

Проблема падіння градієнту (Vanishing Gradient Problem)

Під час навчання глибоких нейронних мереж (особливо зі зворотним поширенням помилки) градієнти на ранніх шарах мережі можуть ставати дуже малими. Це призводить до того, що ці шари практично не навчаються, сповільнюючи або навіть унеможливлюючи навчання всієї моделі.

Основні причини:

- Використання активаційних функцій типу sigmoid або tanh, які "стискають" градієнти.
- Велика глибина мережі (багато шарів).
- Неправильна ініціалізація ваг.

Підходи до вирішення:

1. Використання ReLU та її похідних

   ReLU (Rectified Linear Unit): $f(x) = max(0,x)$

   Має градієнт 1 для $x>0$, тому зменшує ризик зникнення градієнту.

   Варіанти: Leaky ReLU, Parametric ReLU (PReLU), Exponential Linear Units (ELU).

2. Нормалізація шарів (Batch Normalization)

- Дозволяє стабілізувати розподіл вхідних даних у кожному шарі.
- Сприяє більш швидкому навчанню.
- Частково усуває проблему зниження градієнту.

3. Ініціалізація ваг

- Xavier Initialization - для tanh-активацій.
- He Initialization - для ReLU-активацій.
- Забезпечує стабільний масштаб сигналу при його проходженні через

шари.

4. Використання залишкових зв'язків (ResNet)

Додають прямі шляхи (skip-connections), що дозволяють градієнтам ефективніше передаватися назад.

$$y = F(x)+x$$

5. Використання рекурентних архітектур з контролем градієнтів.

Наприклад, LSTM і GRU - мають внутрішні механізми для збереження градієнтів у часі.

Підходи до розпізнавання об’єктів різних за розміром

У зображеннях або даних сенсорів об'єкти можуть з’являтися в різних масштабах. Стандартна CNN може "не впізнати" об’єкт, якщо його розмір відрізняється від того, який був у тренувальній вибірці. Основні підходи:

1. Використання пулінгу з різним масштабом

- Multi-scale pooling: обробка зображення через кілька гілок, кожна з яких має власний масштаб.
- Spatial Pyramid Pooling (SPP): дозволяє мережі адаптуватися до об'єктів різного розміру без необхідності однакового розміру вхідних даних.

2. Використання FPN (Feature Pyramid Network)

- Створення піраміди ознак (features) із різною роздільністю.
- Використовується у сучасних архітектурах детекції об’єктів (наприклад, RetinaNet, Mask R-CNN).

3. Віконний підхід (sliding window) з різними розмірами

Використовується для обробки фіксованих ділянок зображення. Наприклад, YOLO або SSD використовують сітку прив’язок (anchor boxes) різних розмірів.

4. Трансформери та self-attention

Attention-механізм автоматично фокусує увагу на об’єкти незалежно від їх масштабу або положення. Наприклад, у Vision Transformer (ViT) або DETR.

5. Аугментація даних.

Масштабування, кропінг, обертання зображень під час тренування дозволяє моделі краще узагальнювати.

Підходи до зменшення обчислювальної ресурсоємності

1. Прискорені архітектури

- MobileNet - використовує depthwise separable convolutions.
- EfficientNet - збалансоване масштабування ширини, глибини та роздільності.
- ShuffleNet - ефективна архітектура для мобільних пристроїв.

2. Квантування (Quantization)

- Заміна чисел з плаваючою комою (FP32) на цілочисельні (INT8).
- Зменшує обсяг пам’яті та прискорює інференс.

3. Прунінг (Pruning)

- Видалення непотрібних зв’язків або нейронів.
- Може бути структурний (видалення цілих фільтрів) або неструктурний.

4. Зменшення розмірності

PCA або Autoencoder для зменшення кількості ознак перед передачею в мережу.

5. Заміна згорток на більш дешеві операції

Наприклад, заміна звичайної згортки на точкову + глибинну згортку (MobileNet).

6. Використання TensorRT, ONNX, OpenVINO

Спеціалізовані фреймворки для оптимізації моделей на різних апаратних платформах.

Лекція 9. Згорткові нейронні мережі, призначені для виділення об’єктів

Питання лекції. Особливості задачі виділення об’єктів. Поняття кодера та декодера. Типові структури нейромережевих моделей, призначених для виділення об’єктів.

Особливості задачі виділення об’єктів. Виділення об’єктів або сегментація - це задача класифікації кожного пікселя зображення. Основна мета - розділити зображення на логічні області, що відповідають об'єктам або фону. Основні типи сегментації:

- Semantic segmentation – кожен піксель класифікується у певний клас (наприклад, "будівля", "людина", "асфальт").
- Instance segmentation – виявлення окремих об’єктів одного класу (наприклад, розділення двох людей на зображенні).
- Panoptic segmentation – поєднує semantic та instance segmentation.

Особливості задачі:

- Висока точність позиціонування: потрібна на рівні пікселя.
- Класифікація пікселя залежить від контексту навколо.
- Велика обчислювальна складність: висока роздільність зображень + складна структура мереж.

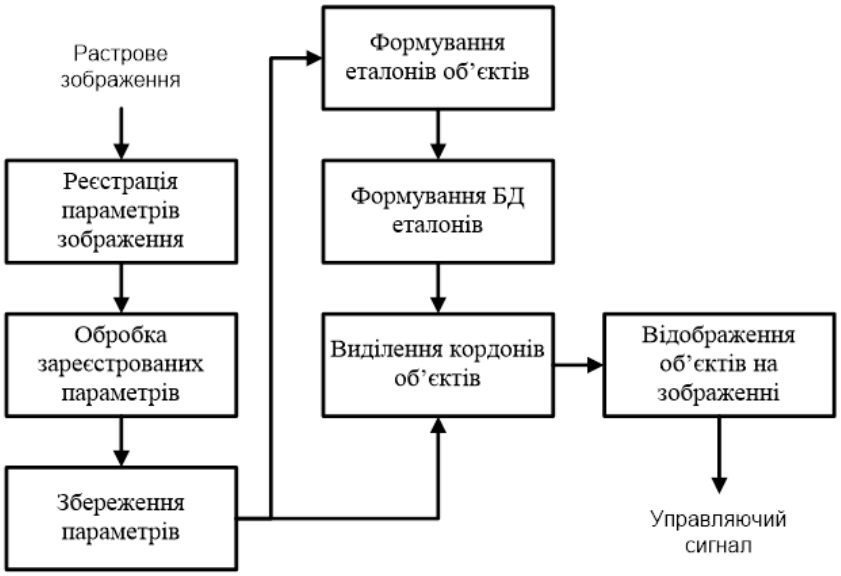

Рис. 4 Типова послідовність функціонування системи виділення об’єктів на растрових зображеннях

Поняття кодера та декодера. Архітектура "кодер-декодер" дозволяє ефективно виділяти об’єкти, зберігаючи просторову точність. Існує багато варіацій моделей: від компактних (SegNet) до потужних і дорогих (DeepLab, ViT). Вибір нейромережевої моделі залежить від задачі, доступних обчислювальних ресурсів та вимог до точності. Кодер (Encoder) - це частина нейромережі, яка зменшує розмірність зображення (downsampling), виділяючи ключові ознаки (features). Декодер (Decoder) - це частина нейромережі, яка відновлює розмірність зображення (upsampling) до вихідного розміру, щоб створити маску сегментації. Типові операції: Upsampling (інтерполяція, транспонована згортка - transposed convolution); Convolution; Skip connections - прямі зв’язки з кодера.

Основна концепція архітектури "кодер-декодер": кодуємо складне зображення в стислу форму → відновлюємо його, додаючи просторову точність. Приклад архітектур нейронних мереж: U-Net, SegNet, DeepLab.

Принципова типова схема НММ, призначеної для виділення об’єктів на зображенні, показана на рис. 5. На вхід представленої моделі подається зображення в форматі RGB, а виходом являється сегментоване зображення.

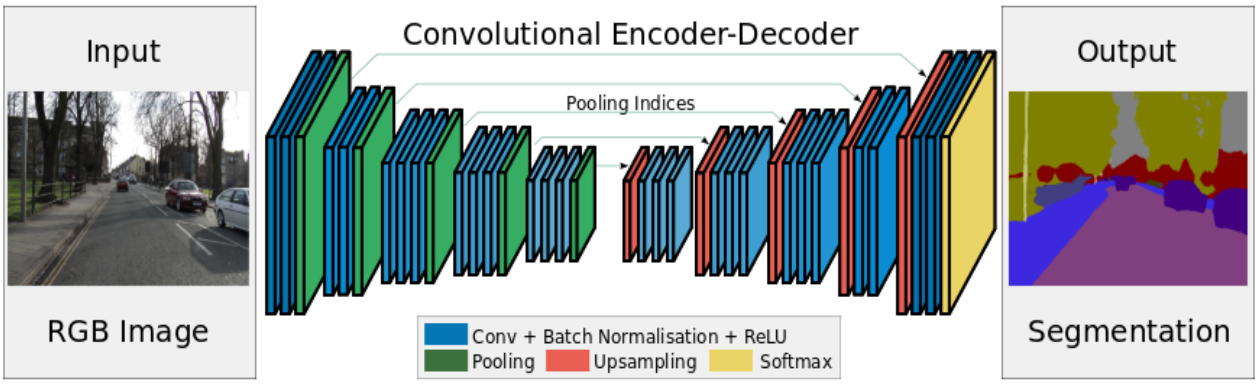

Рис. 5. Типова схема нейромережевої моделі, призначеної для виділення об’єктів на зображеннях

Структура кодера на основі ЗНМ типу VGG-16 показана на рис. 6. Передбачено, що на вхід кодера на базі VGG-16, повинно надходити растрове зображення розміром 224 на 224 пікселя, зареєстроване у форматі RGB. Для першого згорткового шару ядро згортки має розмір 1×1, що забезпечує зменшення вхідних кольорових каналів за рахунок їх лінійної трансформації з

наступною подачею в функцію активації. Далі зображення проходять через стек згорткових шарів, в яких застосовані ядра згортки розміром 3×3. Крок згортки встановлено рівним 1 пікселю. Після кожного шару згортки реалізується операція доповнення нулями, що дозволяє зберегти розмір карти згортки рівним розміру вхідного зображення. Використання кодеру на основі VGG передбачає необхідність масштабування вхідного зображення до розміру 224 на 224 пікселя. Виходом такого кодеру являється матриця ознак розміром $7 \times 7 \times 512$ елементів.

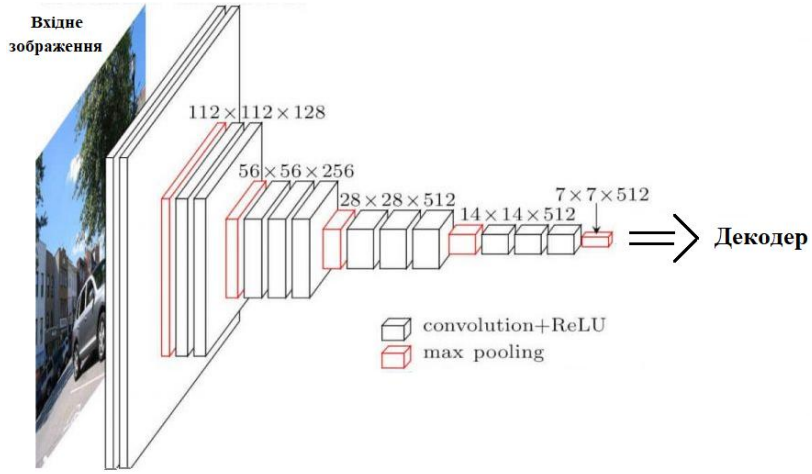

Рис. 6. Структура кодера на основі ЗНМ типу VGG-16

Типові структури нейромережевих моделей, призначених для виділення об'єктів

1. U-Net - одна з найпопулярніших архітектур для семантичної сегментації. Складові: кодер - кілька згорткових блоків + max-pooling; декодер - транспоновані згортки для upsampling; skip-зв'язки між симетричними шарами кодувальника і декодувальника - передають просторову інформацію.

2. SegNet - побудована на базі VGG. Декодер використовує індекси пулінгу з кодера для точного відновлення положення ознак. Компактна модель, зручна для мобільного використання.

3. DeepLab (v3, v3+). Використовує атразійну згортку (dilated convolution) - дозволяє розширити область огляду без зменшення роздільності.

4. Mask R-CNN - вирішує задачу instance segmentation.

5. Vision Transformer (ViT) + Segmenter - трансформери у задачах

сегментації. Дає можливість розпізнавати довгі залежності на зображенні.

Розглянемо підхід на основі багатоетапного симетричного ресамплінгу передбачає, що для розробки кодеру і декодеру використовується однакові ЗНМ з дзеркально відображеною структурою. Тобто структурні ЗНМ кодера в зворотному порядку відповідають структурі ЗНМ декодера. При цьому деякі шари згортки кодера та декодера зв'язані між собою. Для прикладу на рис. 7 показано нейромережева модель виділення об'єктів, декодер якої побудовано на основі третього підходу з використанням ЗНМ типу LeNet до складу якого входить 7 шарів згортки та 4 шари ресамплінгу (трансформації).

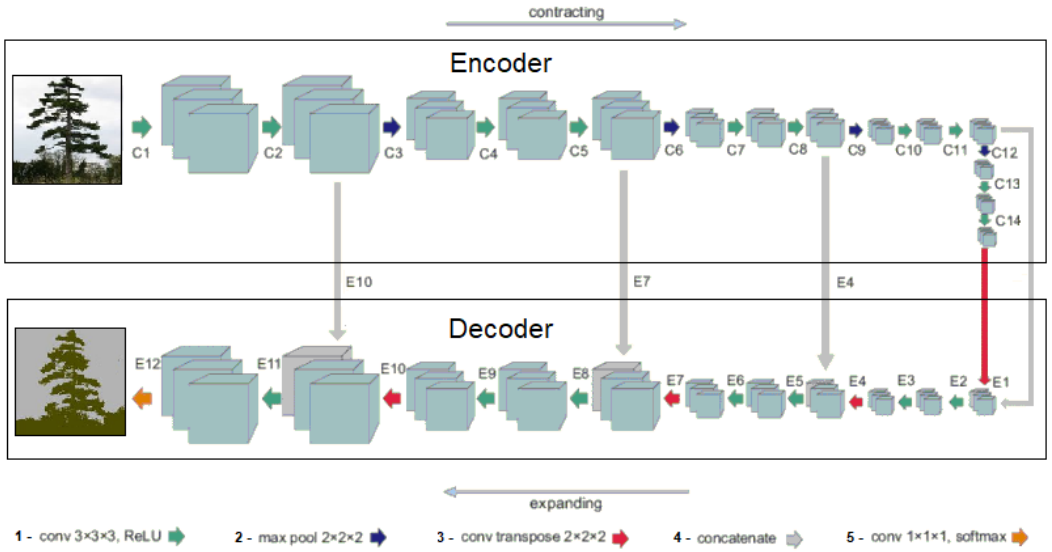

Рис. 7. Приклад застосування декодеру на основі багатоетапного симетричного ресамплінгу

Структура декодеру, що також складається із 4 стеків, має певні відмінності. До складу кожного із стеків входить шар ресемплінгу, шар згортки з ядром розміру 2×2, блок інтеграції з відповідною картою згортки кодеру та двох шарів згортки з ядром розміру 3×3. В останньому шарі використовується ядро згортки 1×1 для співставлення 64-вимірного вектора ознак з пікселями обробленого вихідного зображення. Блоки, що входять до складу показаного на рис. 7 декодеру, мають позначення типу Ex, де x – номер блоку.

Лекція 10. Перспективні типи згорткових нейронних мереж

Питання лекції. Нейромережева модель типу GoogleNet. Нейромережева модель SqueezeNet. Нейромережева модель YoLo.

Нейромережева модель типу GoogleNet. GoogleNet (Inception v1) представлена компанією Google у 2014 році. Основна ідея: використання Inception-модуля, який дозволяє моделі бути глибокою, але ефективною за кількістю параметрів.

Архітектурні особливості GoogleNet.

1. Inception-модуль

- замість того, щоб обирати фільтри одного розміру, Inception-модуль паралельно використовує кілька згорток:

1×1, 3×3, 5×5, і пулінг.

- результати об'єднуються (concatenation) вздовж каналу.

2. Застосування 1×1 згортки

- зменшення кількості каналів (компресія ознак).

- підвищення ефективності моделі.

3. Архітектура GoogleNet

- 22 шари (глибина).

- приблизно 5 млн параметрів - набагато менше, ніж у AlexNet (60 млн).

- використовує глобальний average pooling замість fully connected шарів → менше параметрів.

Переваги GoogleNet: висока точність; ефективність по параметрам і швидкодії; можливість обробляти ознаки різного масштабу в одному шарі.

Нейромережева модель SqueezeNet. Представлена у 2016 році як модель з дуже малою кількістю параметрів, але з якістю, схожою на AlexNet. Основна мета - розгортання на пристроях з обмеженими ресурсами (мобільні, вбудовані системи).

Архітектура *SqueezeNet*

Fire-модуль складається з:

- Squeeze-частина - 1×1 згортки для зменшення кількості каналів.

- Expand-частина - поєднання 1×1 та 3×3 згорток.

Архітектурні принципи:

- Використовувати максимум 1×1 згорток.
- Зменшити кількість входів у 3×3 згортки.
- Виконувати пізній downsampling - для збереження просторової інформації на глибших рівнях.

Переваги SqueezeNet: ~1,3 млн параметрів (в 50 разів менше, ніж у AlexNet); компактна модель: підходить для реального часу; придатна для IoT, мобільних пристроїв, інтелектуальних сенсорів.

Нейромережева модель YOLO (You Only Look Once).

YOLO - сімейство моделей для реального часу обробки об’єктів на зображенні (object detection). Вперше представлена в 2016 році (YOLOv1), з того часу з’явилися численні покращення: YOLOv2 → v3 → v4 → v5 → v8.

Основна ідея - замість сканування зображення кількома вікнами (як у R-CNN), YOLO розбиває зображення на сітку та для кожної клітинки одночасно:

- Визначає, чи є об'єкт.
- Передбачає координати рамки (bounding box).
- Передбачає клас об'єкта.

YOLO - вся детекція виконується за один прохід (single shot).

Архітектура (на прикладі YOLOv3)

- Базується на Darknet-53 як фреймворку feature extraction.
- Використовує residual connections.
- Детекція об’єктів відбувається на трьох масштабах - для розпізнавання об’єктів різного розміру.

Переваги YOLO: надзвичайна швидкість (реальний час); компактність - працює на CPU і GPU.