

Лабораторна робота № 1

Тема: Використання згорткових нейронних мереж для класифікації зображень.

Мета: Навчитись розробляти архітектури згорткових нейронних мереж для класифікації зображень

Питання для обговорення:

1. Класифікація нейронних мереж
2. Шар згортки
3. Субдискретизуючий шар

1. Теоретичні відомості

Згорткові нейронні мережі (Convolutional neural networks - CNNs) подібні до звичайних штучних нейронних мереж. Вони також складаються з нейронів, які мають певні ваги і зміщення. Проте вхідними даними для них є зображення. Тобто вхідні дані можна подати у трьох вимірах – висота, ширина та кількість каналів. В той час як звичайні нейронні мережі приймають на вхід одномірний вектор. Саме через це CNN-мережі стали такими популярними. Оскільки такий підхід значно зменшує кількість параметрів в самій мережі і, відповідно, зменшує час навчання.

Типова згорткова нейронна мережа складається із таких шарів: згорткові, субдискретизуючі та повнозв'язні шари. Приклад такої мережі зображено на рисунку 1.1.

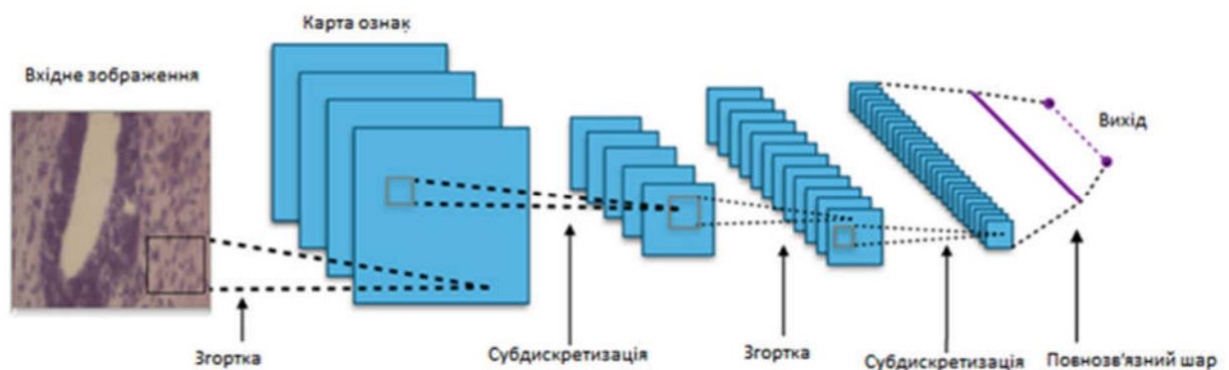


Рисунок 1.1 – Загальна структура згорткової нейронної мережі

Згортковий шар. Основними гіперпараметрами даного шару є розмір ядра (k - kernel size), кількість вхідних та вихідних фільтрів (f - filters), крок (s - stride) та доповнення нулями (p - padding). Принцип роботи полягає в наступному: пікселі ядра поелементно перемножуються із вхідними пікселями зображення та сумуються. Ядро переміщується із кроком, що дорівнює параметру stride, поки не «пройде» все вхідне зображення. Вихідний об'єм обчислюється за формулою $O = (W-K+2P)/S+1$.

Приклад. Нехай маємо вхідне зображення розміром $32 \times 32 \times 3$ (W - ширина, H - висота, C - кількість каналів). Застосуємо згортковий шар $k=3$, $s=1$, $p=0$, $f=96$. Тоді отримаємо вихідне зображення розміром $30 \times 30 \times 96$.

1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0

4		

Рисунок 1.2 – Згортка зображення

Субдискретизація. Зазвичай цей шар використовується для зменшення розмірності зображення. В більшості випадків використовують максимальний пулінг, який вибирає максимальні значення із вхідного об'єму. Гіперпараметрами є розмір ядра та крок. Вихідний об'єм обчислюється за формулою $O = (W-K)/S+1$.

Приклад. Зображення $30 \times 30 \times 96$ пікселів після застосування пулінгу буде мати об'єм $14 \times 14 \times 96$ пікселів.

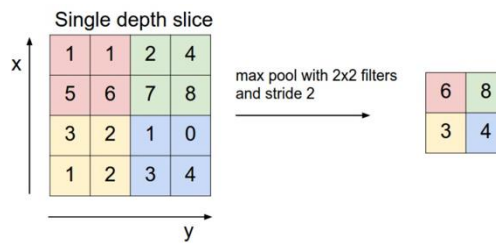


Рисунок 1.3 – Максимальний пулінг

Повнозв'язний шар. Цей шар застосовується в якості вихідного шару в задачах класифікації.

Приклад програмного коду для роботи згорткової нейронної мережі:

```
model = models.Sequential()
model.add(layers.Conv2D(32, (3, 3), activation='relu', input_shape=(32, 32, 3)))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
```

Model: "sequential"

Layer (type)	Output Shape	Param #
=====		
conv2d (Conv2D)	(None, 30, 30, 32)	896

max_pooling2d (MaxPooling2D)	(None, 15, 15, 32)	0

conv2d_1 (Conv2D)	(None, 13, 13, 64)	18496

max_pooling2d_1 (MaxPooling2D)	(None, 6, 6, 64)	0

conv2d_2 (Conv2D)	(None, 4, 4, 64)	36928
=====		
Total params: 56,320		
Trainable params: 56,320		
Non-trainable params: 0		

Рисунок 1.4 – Структура шарів нейронної мережі

Приклад датасету наведено на рисунку 1.5.

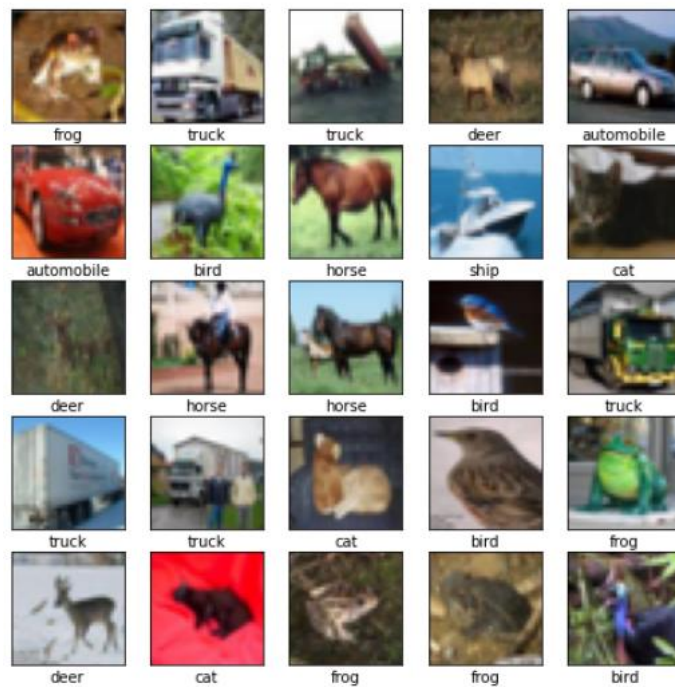


Рисунок 1.5 - Приклад датасету

Приклади результатів класифікації наведено на рисунку 1.6.

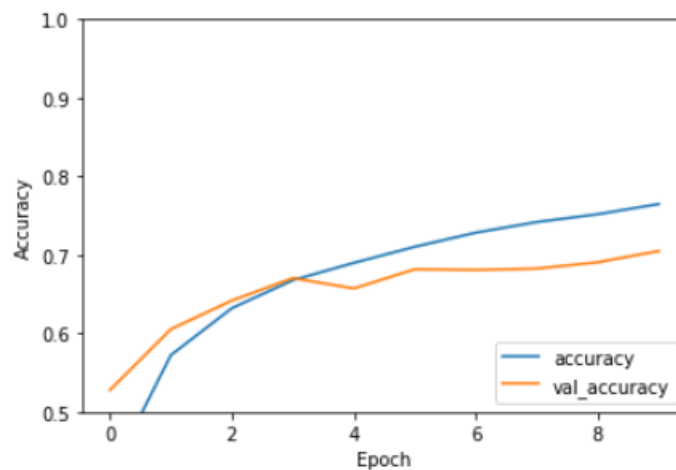


Рисунок 1.6 - Приклади результатів класифікації

Детальнішу інформацію можна переглянути за посиланням:
<https://cs231n.github.io/convolutional-networks>

2. Хід роботи.

- Розробити власну архітектуру згорткової нейронної мережі у середовищі Kaggle. Приклад реалізації згорткової нейронної мережі <https://www.kaggle.com/code/anandhuh/image-classification-using-cnn-for-beginners> . Власна розробка полягає у створенні або редагуванні згорткових, субдискретизуючих та повнозв'язних шарів нейронної мережі.
- <https://www.kaggle.com/code/parnianmalekian/simple-cnn-example>
- Скопіювати Colab Notebook за посиланням: <https://colab.research.google.com/drive/1cx95kKLN8-k60NYhPORBfvGopNYBQLxQ?usp=sharing>
- Експериментальним шляхом дослідити як змінюється точність класифікації при додаванні/видаленні шарів, коригуванні гіперпараметрів. Порівняти час навчання на CPU та GPU.

3. Структура звіту лабораторної роботи.

- Титульна сторінка.
- Тема та мета роботи.
- Код програми розв'язку індивідуального завдання.
- Копії екранів роботи програми.
- Висновки.

4. Контрольні запитання

1. Що таке CNN?
2. Які типи шарів використовуються в CNN та як вони працюють?
3. Відомі архітектури згорткових мереж.
4. Від чого залежить точність класифікації?
5. Що таке overfitting, underfitting?

