

Звіт до лабораторної роботи №4

Дисципліна: Вступ до штучного інтелекту

Тема: Нейронні мережі

Мета: Удосконалити агента-автомобіль шляхом додавання системи контролю швидкості на основі згорткової нейронної мережі та отримати навички класифікації зображень.

Постановка завдання

Метою роботи є удосконалення інтелектуального агента-автомобіля, який було реалізовано в попередній лабораторній роботі. Зокрема, потрібно додати систему контролю швидкості, що використовує згорткову нейронну мережу (CNN) для розпізнавання дорожніх знаків обмеження швидкості. Знаки представлені у вигляді зображень цифр (2–9) з набору MNIST, де кожна цифра відповідає обмеженню швидкості від 20 до 90 км/год.

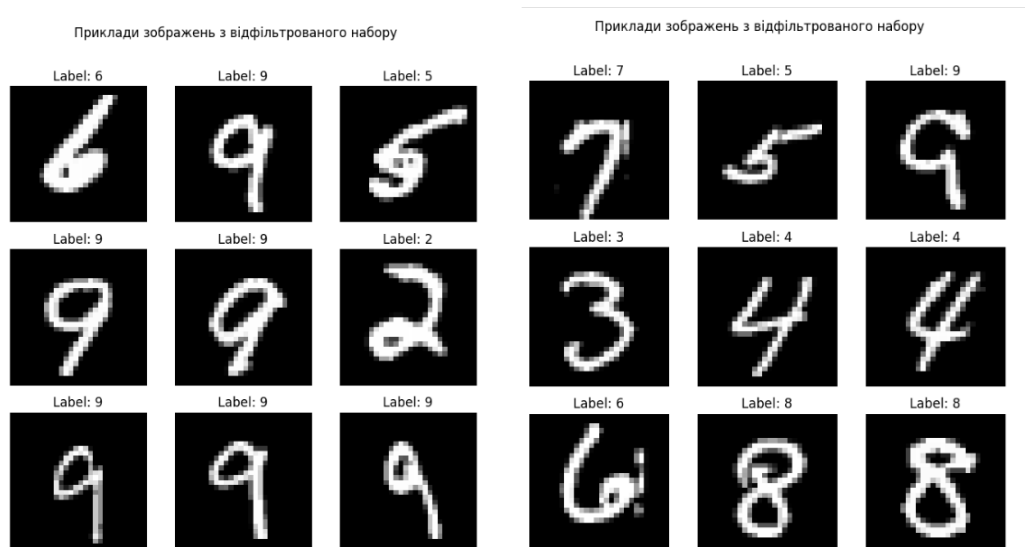
Завдання роботи:

- Отримати базові навички роботи з нейронними мережами.
- Реалізувати класифікацію зображень з набору MNIST за допомогою CNN.
- Інтегрувати модель розпізнавання в агента для визначення допустимої швидкості руху.

Програмна реалізація

1. Підготовка даних:

- Завантажено датасет MNIST та відфільтровано зображення з цифрами від 2 до 9.
- Проведено нормалізацію пікселів та додано канали для роботи з CNN.
- Виконано one-hot encoding міток для 8 класів (2–9).



2. Створення моделі CNN:

Модел ь складається з трьох згорткових шарів:

- Conv2D(32, 3x3) + MaxPooling(2x2)
- Conv2D(64, 3x3) + MaxPooling(2x2)
- Conv2D(64, 3x3)

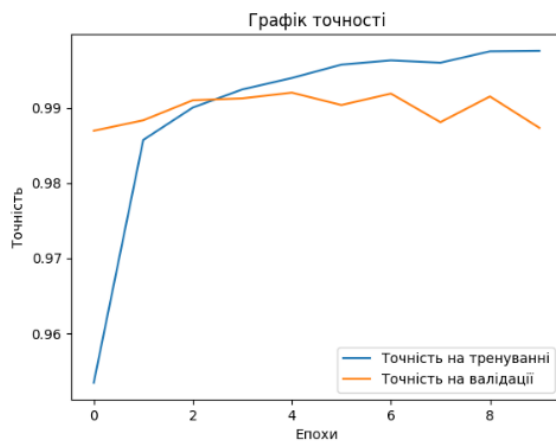
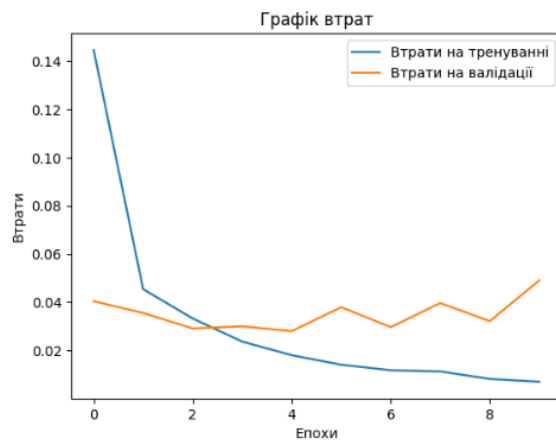
Далі йде Flatten, один прихований Dense(64), та вихідний Dense(8) з softmax-активацією.

3. Навчання моделі:

- Використано оптимізатор Adam та функцію втрат categorical_crossentropy.
- Кількість епох: 10

4. Графіки результатів:

- Побудовано графік втрат (loss) для train/val.
- Побудовано графік точності (accuracy) для train/val.



-

5. Збереження та завантаження моделі:

- Модель збережена у форматі `.keras`.

6. Інтеграція в агента:

```
# 🍷 Тест розпізнавання випадкової цифри

import numpy as np
import matplotlib.pyplot as plt

# Випадковий індекс із тестового набору
random_idx = np.random.randint(0, x_test_filtered.shape[0])

# Вибір зображення та справжньої мітки
random_image = x_test_filtered[random_idx]
true_label = np.argmax(y_test_filtered[random_idx]) + 2

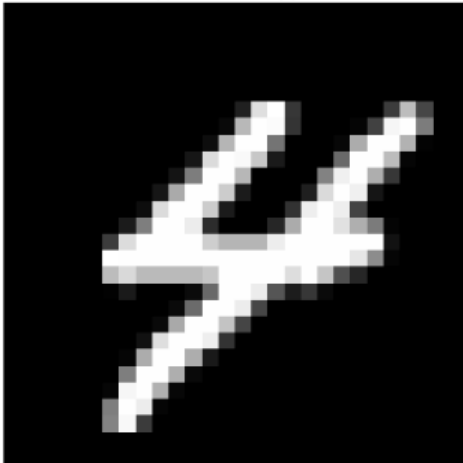
# Виведення зображення
plt.imshow(random_image.squeeze(), cmap='gray')
plt.title(f"Справжня цифра: {true_label}")
plt.axis('off')
plt.show()

# Розширення розмірності для подачі в модель (batch_size = 1)
image_for_prediction = np.expand_dims(random_image, axis=0)

# Прогноз. Результат розпізнавання
prediction = loaded_model.predict(image_for_prediction)
predicted_label = np.argmax(prediction) + 2

print(f"Розпізнано цифру: {predicted_label}")
```

Справжня цифра: 4



1/1 ————— 0s 64ms/step
Розпізнано цифру: 4

- Модель використовується для розпізнавання знаків при кожному русі агента по графу.
- Згенеровані зображення знаків відповідають випадковим обмеженням на дорогах.
- Агент отримує зображення перед переміщенням, розпізнає цифру та визначає швидкість.
- Весь шлях агента візуалізується, включаючи обрані обмеження швидкості.

Отримані результати

- Модель досягла точності близько **98.8%** на валідації.

- Графік втрат демонструє добре зниження втрат без переобучення.
- Агент успішно інтегрує модель розпізнавання в логіку свого руху.
- Розпізнавання працює коректно: цифра → швидкість → вплив на рішення агента.

додано до бачених вузлів: (1, 0)

Крок 3: Агент рухається до (1, 1)

Поточний напрямок: right

1/1 0s 56ms/step

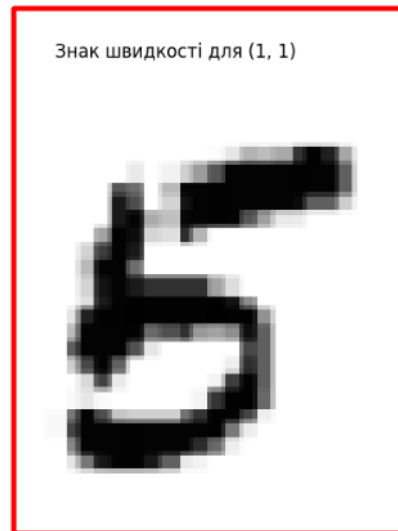
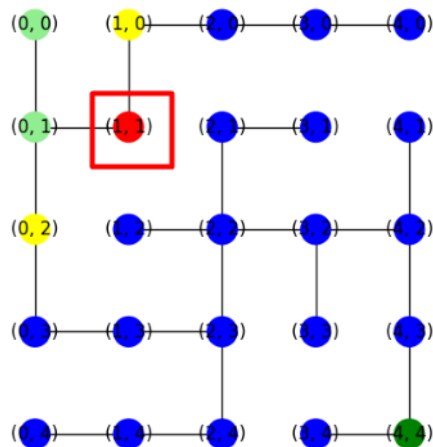
Розпізнано знак 5, обмеження: 50 км/год

Обмеження швидкості для дороги (0, 1) → (1, 1): 50 км/год

— поворот наліво

Завантаження зображення: signs/speed_sign_60.png

Агент знаходиться у (1, 1)



TELL: Вузол (1, 0) додано до відвіданих.

База знань оновлена: додано інформацію про вузол (1, 0)

Сканування сусідніх перехресть для вузла (1, 0): [(1, 1), (2, 0)]

TELL: Дорога (1, 0) ↔ (1, 1) додана в базу знань.

TELL: Дорожній знак (1, 0) → (1, 1): down додано.

Дорожній знак: з (1, 0) у (1, 1) → down

TELL: Дорога (1, 0) ↔ (2, 0) додана в базу знань.

TELL: Дорожній знак (1, 0) → (2, 0): right додано.

Дорожній знак: з (1, 0) у (2, 0) → right

Додано до бачених вузлів: (2, 0)

— Крок 4: Агент рухається до (1, 0)

Вирішені проблеми та тести

- **Повторне проходження тієї самої дороги:** якщо агент проходить по тій самій дорозі ще раз, розпізнана цифра не змінюється — знак фіксується при першому проходженні.
- **Різні швидкості в обидві сторони:** реалізовано підтримку напрямків — одна і та сама дорога в напрямках A→B та B→A може мати різні знаки і відповідно різну швидкість.
- **Захист від самозаміни:** значення швидкості, прив'язані до дороги, не перезаписуються при повторному русі.

Висновки

У ході лабораторної роботи було реалізовано згорткову нейронну мережу для класифікації зображень цифр 2–9 з датасету MNIST. Модель було успішно інтегровано в агента, який при кожному переміщенні отримує обмеження швидкості з відповідного знаку. Агент використовує цю інформацію для руху по графу з урахуванням правил дорожнього руху.