

Левкович Роман Юрійович

випускник (магістр) факультету прикладної математики та інформатики
Львівський національний університет ім. І. Франка, Україна

Колос Надія Мирославівна 

канд. фіз.-мат. наук, доцент кафедри дискретного аналізу та інтелектуальних систем
Львівський національний університет ім. І. Франка, Україна

АСИСТЕНТ ВОДІЯ НА ОСНОВІ ЗГОРТКОВИХ НЕЙРОННИХ МЕРЕЖ

При розробці автомобілів велика увага приділяється безпеці всіх учасників дорожнього руху. Для цього створюються різного роду додаткові системи, які мають на меті зменшити вплив людського фактора під час керування. Всі ці системи разом складають удосконалену систему допомоги водію (ADAS — advanced driver-assistance system). Прикладами таких рішень є антиблокувальна система, гальмівний асистент, додаткові камери сліпих зон, тощо.

Однією з таких важливих інтелектуальних систем, які потребують багато обчислень, є система розпізнавання дорожніх знаків. Її завданням є повідомляти водію про дорожні знаки, які знаходяться на дорозі.

Для вирішення цього завдання розроблено систему асистента водія, яка працює на основі Adaptive AUTOSAR, оскільки це відкрита система, яка використовується в автомобілях. Також, під час аналізу вирішено використати JSON як формат конфігураційних файлів та C++20 як основний стандарт.

Інтелектуальна система складається з таких додатків:

- Розпізнавач дорожніх знаків
- Генератор даних
- Графічний інтерфейс (GUI).

Розпізнавач дорожніх знаків. Для вирішення завдання розпізнавання дорожніх знаків використано згорткові нейронні мережі. Проаналізовано популярні архітектури які використовуються для даної задачі, та зосереджено увагу на архітектурі Faster R-CNN з основою SqueezeNet, як оптимальне рішення для даної проблеми.

Для імплементації нейронних мереж використано фреймворк PyTorch.

Для кожного навчання мережі спільним був такий набір гіперпараметрів:

- Навчання відбувається впродовж 100 епох;
- Як оптимізатор використано функцію Adam;
- Коефіцієнт швидкості навчання є змінним, за допомогою косинусної нормалізації.

Архітектура основної мережі SqueezeNet показана у (таб. 1).

Таблиця 1

Архітектура мережі SqueezeNet

№	Прошарок	Ядро/страйд	Вхідний	Пром. кан.	1x1 кан.	3x3 кан.
1	conv2	3x3/2				
2	maxpool2	3x3/2				
3	fire		64	16	64	64
4	fire		128	16	64	64

Продовження табл. 1

№	Прошарок	Ядро/страйд	Вхідний	Пром. кан.	1x1 кан.	3x3 кан.
5	maxpool2	3x3/2				
6	fire		128	32	128	128
7	fire		256	32	128	128
8	maxpool2	3x3/2				
9	fire		256	48	192	192
10	fire		384	48	192	192
11	fire		384	64	256	256
12	fire		512	64	256	256

взято з [2]

Для оцінювання роботи нейромережі використано метрики, які пропонуються на PASCAL VOC challenge. Вони базуються на основі таких параметрів як влучність, повнота та міра Жаккара. Нагадаємо, що $mAP@0.5$ — середнє влучностей, якщо міра Жаккара більша за 50%.

Щоб тренування даної мережі тривало розумний період часу необхідно використовувати графічні карти, з мінімум 4Гб пам'яті. Щоб отримати доступ до таких обчислювальних можливостей використано сервіс Google Collaboratory. Завдяки цьому моделі навчаються значно швидше, але час однієї сесії обмежений добою. Тому, для успішного застосування даного сервісу необхідно витримати правильний баланс між розміром вибірки, та можливістю швидко його розгорнути в даному середовищі.

Дорожні знаки України стандартизовані ратифікацією Віденської конвенції про дорожній рух 1968-го року. Відповідно, для того, щоб обрана вибірка мала застосування на території України, необхідно вибрати дані з країн, які теж ратифікували дану конвенцію.

GTSDb — це популярна вибірка з 1000 фото дорожніх знаків Німеччини за різних умов. Вона складається з 43 класів знаків, і є класичною вибіркою для знаходження об'єктів.

Результат тренування нашої нейромережі на даній вибірці даних був доволі низьким ($mAP@0.5=42\%$), зважаючи на кількість параметрів. Можливо, проблемою є розмір вибірки.

Mapillary traffic sign dataset — це вибірка фотографій дорожніх знаків з цілого світу. Якщо розглянути тільки повністю анотовану частину, то залишиться 30000 фото. В цій вибірці кількість класів спадає експоненційно. Це погано впливатиме на результат навчання, оскільки багато класів збільшує розмір мережі, а мала кількість прикладів не дозволить отримати хорошу точність. Тому проведено навчання лише на частині вибірки, а саме знаках:

- "Головна дорога"
- "В'їзд заборонено"
- "Дати дорогу".

В результаті фільтрації отримано вибірку з 5000 фото.

Нейромережа після навчання показала результат розпізнавання за метрикою $mAP@0.5$ рівний 62%. Це вже кращий результат, але 3 класи дорожніх знаків є невеликою вибіркою.

Після цього було застосовано підхід finetuning щоб збільшити вибірку: замінено шари моделі, які відповідальні за обчислення результатів з 3-х класів, які ми застосували в Mapillary на 44 класи GTSDb. В результаті нового навчання отримали mAP@0.5 = 73%.

Для перевірки стабільності моделі, було перевірено її роботу на перехресному затвердженні методом виключення по 100. В результаті, середнє значення метрики mAP@0.5 отримали 66%.

Значення інших відомих метрик з MS COCO (Common objects in context) також наведені у (таб. 2).

Таблиця 2

Результат перехресної перевірки

mAP@0.5	66%
AR (max=10)	70%
mAP@0.5 (small)	54%
mAP@0.5 (medium)	63%
mAP@0.5 (big)	65%
AR (max=100)	60%
AR (max=1)	47%

[авторська розробка]

Отже, попереднє навчання на великій вибірці з достатньою кількістю прикладів дозволяє отримати значно кращий результат.

Генератор даних. Даний блок імітує роботу сенсорів автомобіля, і генерує дані, які б генерував автомобіль під час руху: швидкість, роботу індикаторів, заряд батареї. Також, він генерує дані, які відповідають передній камері автомобіля. Цю функцію виконує відео, яке передається параметром командного рядка на старті сервісу.

Генератор даних є прикладом класичного сервісу-постачальника, який надає дані для довільної кількості споживачів. Тому методи публічного інтерфейсу створені за підходом публікація-підписки.

Графічний інтерфейс. Для розробки графічного інтерфейсу використано фреймворк Qt та мову QML для створення розмітки елементів. Даний сервіс працює впродовж всього життя системи, та є класичним прикладом користувача: він підписується на повідомлення від генератора даних, та розпізнавача знаків, і показує їх водію.

Список використаних джерел:

1. Christian Ertler, Jerneja Mislej, Tobias Ollmann, Lorenzo Porzi, Gerhard Neuhold, Yubin Kuang (2020) The Mapillary Traffic Sign Dataset for Detection and Classification on a Global Scale [Електронний ресурс] Вилучено з: <https://arxiv.org/abs/1909.04422>
2. Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, Kurt Keutzer. (2016) SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size [Електронний ресурс] Вилучено з: <https://arxiv.org/abs/1602.07360>
3. Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun. (2015) Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *Advances in Neural Information Processing Systems*, (28), 91–99. Вилучено з: <https://arxiv.org/abs/1506.01497>
4. AUTOSAR C++14 code guidelines. [Електронний ресурс] Вилучено з: https://www.autosar.org/fileadmin/standards/adaptive/18-03/AUTOSAR_RS_CPP14Guidelines.pdf