

НТУУ «КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. ІГОРЯ СІКОРСЬКОГО»
«ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ»
КАФЕДРА СИСТЕМНОГО ПРОЕКТУВАННЯ

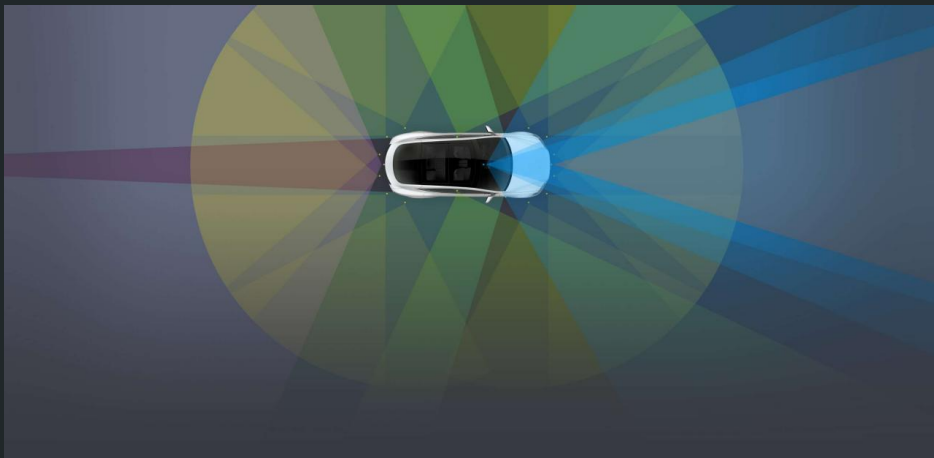
Допомога водію у керуванні транспортним засобом з використанням методів комп'ютерного зору

Виконав: Чернюк А.О. ДА-62
Науковий керівник: Булах Б. В.

Постановка завдання

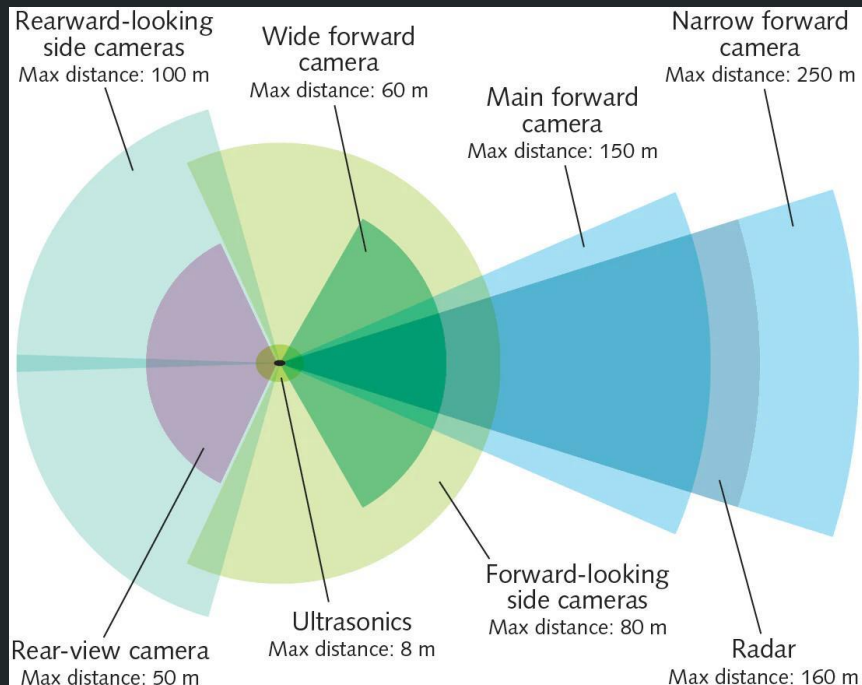
Розглянути систему допомоги в керування автомобілем, яка відповідає за розпізнавання зовнішнього світу, що має містити в собі функціонал розпізнавання проїзної частини дороги; детекцію інших транспортних засобів, як учасників руху; аналіз можливих зіткнень.

Актуальність



Питання безпеки дорожнього руху є завжди актуальним. Хоч системи допомоги водію стрімко розвиваються та впроваджуються, вони все ще доступні лише у деяких з серійних автомобілів. В той же час, відеофіксація поїздки вже давно є звичною, а можливості сучасних пристроїв з обробки відео вже дозволяють використовувати їх як елемент допомоги водію під час руху.

Система розпізнавання зовнішнього світу



Можливі частини системи:

- LiDAR
- RADAR
- **Набір відеокамер**

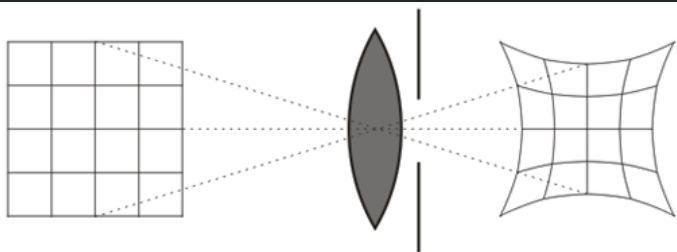
Модуль визначення проїзної частини дороги

- Компенсація дисторсії.
- Створення перпендикулярного поля зору засобу пересування
- Генерація ліній дороги

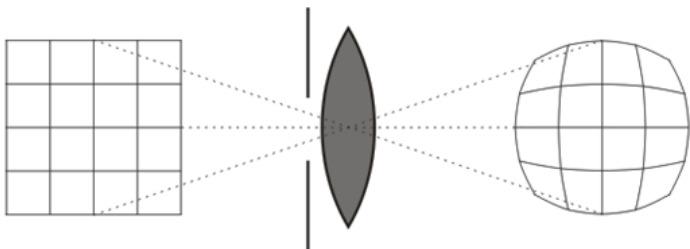
Модуль розпізнавання об'єктів на дорозі

- Конволюційні нейронні мережі.
- YOLO, її застосування в даному продукті. Принцип роботи.
- Tiny-YOLO проти YOLOv3.
- Сповіщення водія

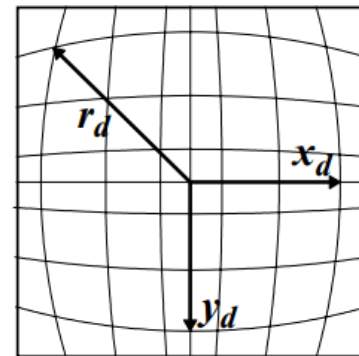
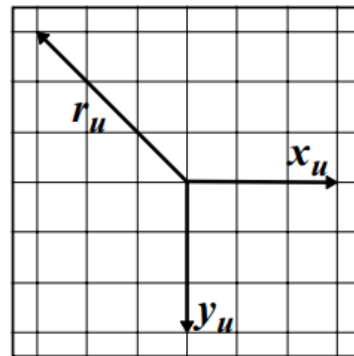
Компенсація спотворення зображення



Подушкове спотворення



Бочкове спотворення



Компенсація спотворення зображення здійснюється наступною формулою:

$$r_u = r_d * (1 + k * r_d^2)$$

Створення перпендикулярного поля зору засобу пересування

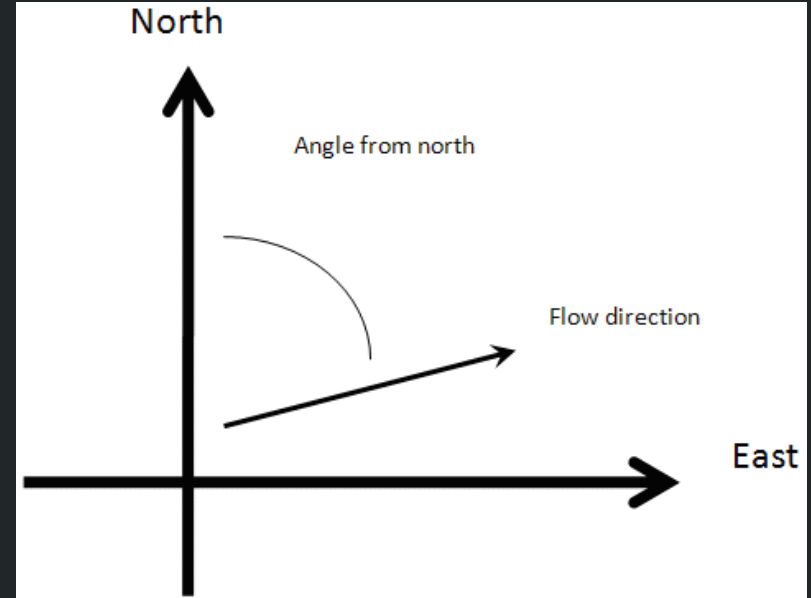
- Знаходження контурів
- NMS, фільтрація по інтенсивності і кольорам
- Зміна перспективи

Знаходження контурів

Градiєнт за осями x і y . Можемо побачити ступiнь зростання, або спадання функції

$$|G| = \sqrt{G_x^2 + G_y^2} \quad \text{- обчислене значення градиєнту}$$

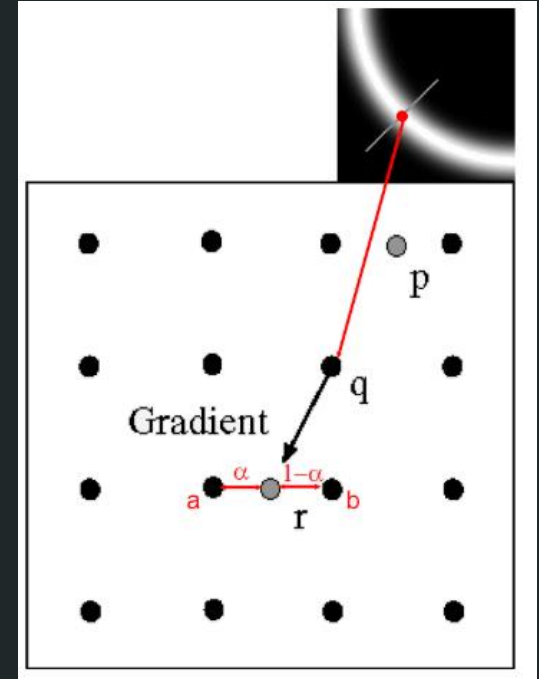
$$\phi = \arctan\left(\frac{G_y}{G_x}\right) \quad \text{- напрямок градиєнту}$$



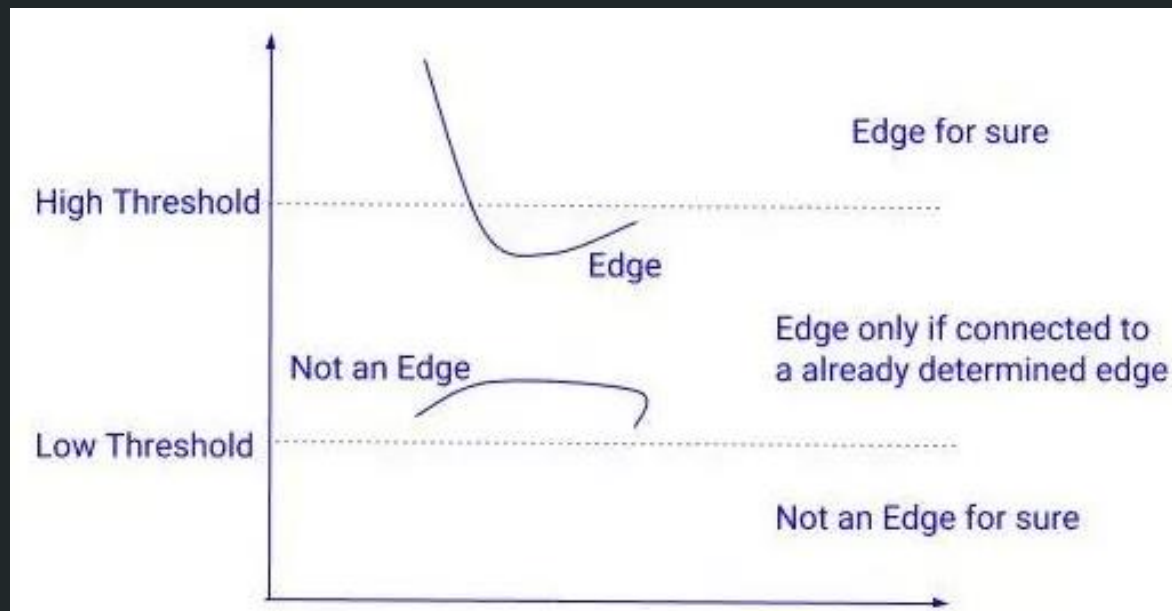
NMS (Non maximum suppression)

Метод поглинання базується на пошуку пікселя з найбільшим значенням серед всіх, що можуть виступати контурами. Як бачимо на зображенні, це відбувається, коли піксель q має більшу інтенсивність ніж пікселі p і r відповідно. Пікселі r і p знаходяться на шляху градієнта точки q . В тому разі, якщо умова виконується, піксель залишається на зображенні, інакше – ні.

$$r = \alpha b + (1-\alpha)a - \text{обчислення NMS}$$



Фільтрація по інтенсивності пікселя і кольорам

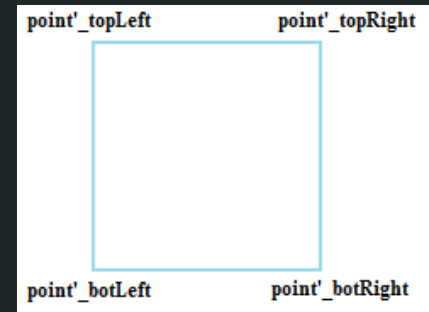
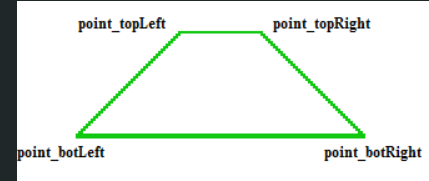


Сильні, слабкі і «ніякі» контури.

Зміна перспективи зображення

Зміна перспективи поліпшує виділення положення ліній на дорозі. Вирішується це завдання шляхом розв'язання системи лінійних рівнянь за допомогою пошуку коефіцієнтів матриці гомографії. Кожній точці на початковому зображенні ставиться у відповідність бажана точка.

$$(x_1, y_1) \approx (x'_1, y'_1)$$



Результат перетворення



Генерація ліній дороги

- Отримання положень ліній на зображенні, що було отримано.
- Розбиття обидвох ліній по вертикалі на N прямокутників.
- Обчислення зваженої суми положень точок у кожному виділеному прямокутнику.
- Інтерполяція точок по кожному прямокутнику, в якому присутній хоча б 1 виділений піксель

Інтерполяція ліній

В загальному виді
інтерполяційний поліном
має наступний вигляд:

$$y = \sum_{i=0}^N a_{N-i} * x^{N-i} + \varepsilon$$

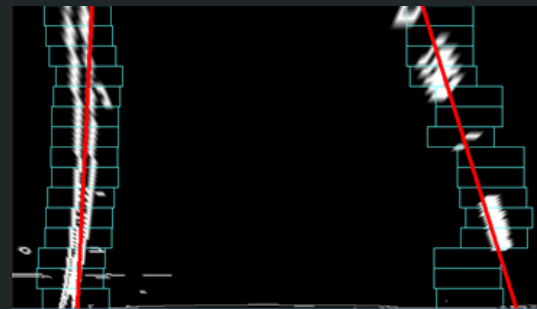
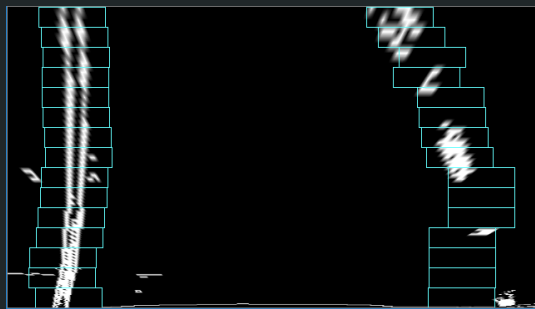
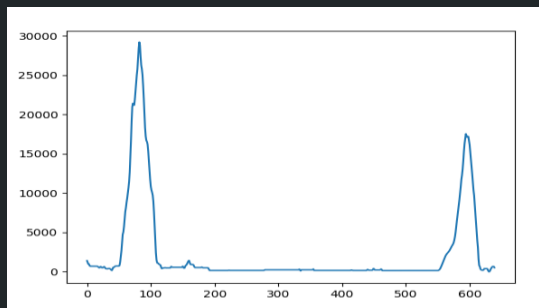
В даній роботі було використано МНК, що
зкладається в створенні поліному, відстань
точок якого до фактичних значень, що
інтерполують, буде мінімальною.

Умова МНК має наступний вигляд:

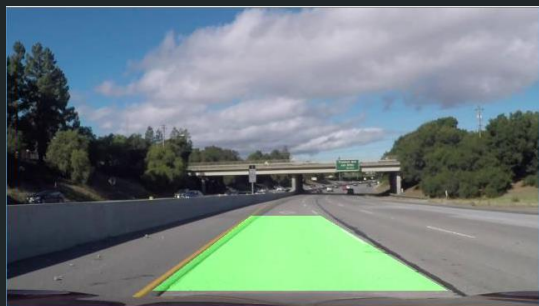
$$\begin{aligned} R^2 &= \sum_{i=1}^N [y_i - (a_0 + a_1 x_1 + \dots + a_k x_i^k)]^2 \\ &= \sum_{i=1}^N [y_i - y]^2 \end{aligned}$$

$$\begin{cases} \frac{\partial(R^2)}{\partial a_0} = 0 \\ \dots \\ \frac{\partial(R^2)}{\partial a_k} = 0 \end{cases}$$

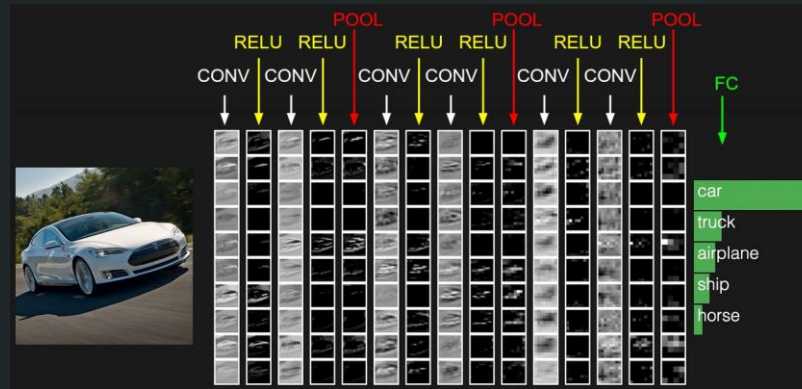
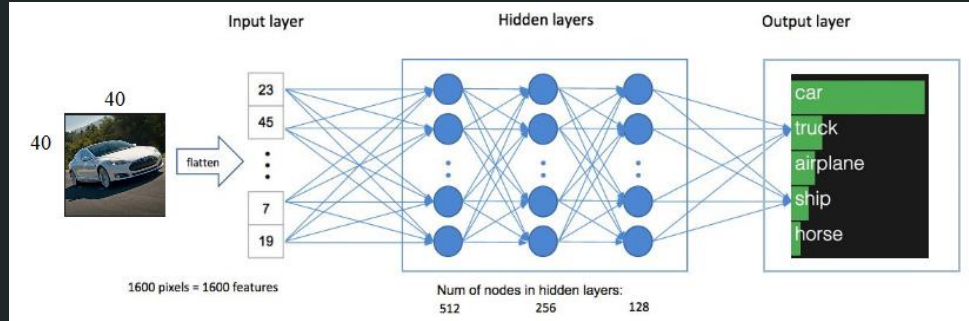
Результат генерації



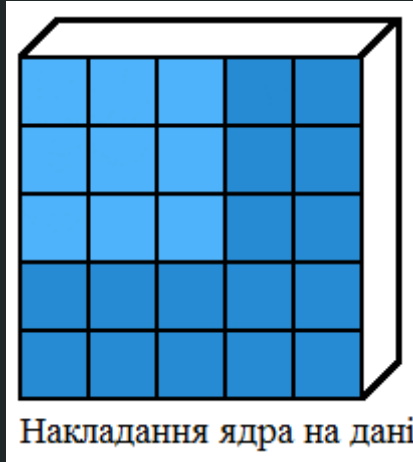
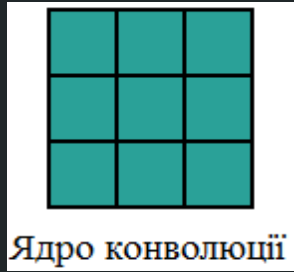
Результат роботи першого модулю програми



Конволюційні нейронні мережі

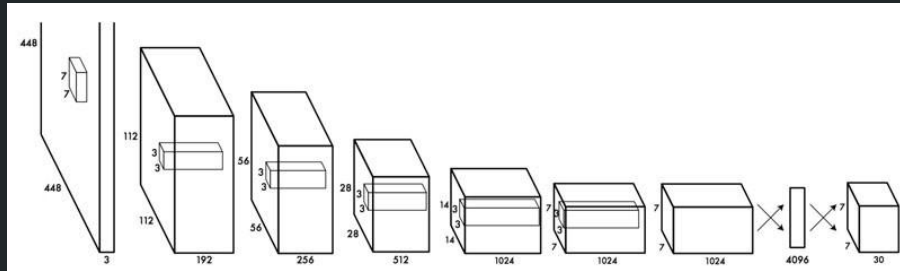


Конволюційні нейронні мережі. Конволюція і Субдискретизація



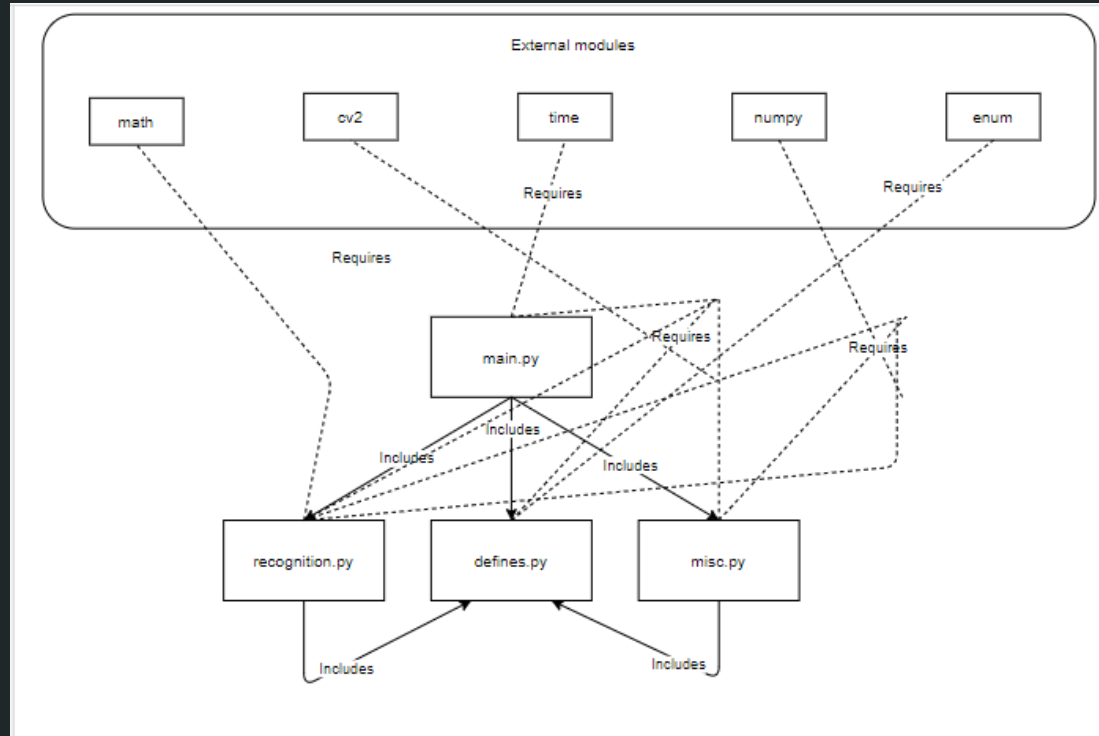
Архітектура YOLO. Принцип її роботи.

- YOLO – з класу SSD – Single Shot Detection. Використовує ядра конволюції і агрегації.
- На виході мережа генерує тензор. Він містить поділене на сітку зображення і вектор характеристик кожної клітинки.
 - Консолідація значень – скалярне множення вектору ймовірностей з вектором, що показує можливе знаходження об'єктів всередині клітини.
 - Фільтрація значень – відсіювання елементів з малим значенням консолідації.
 - NMS – фільтрація зайвих bounding-box – ів за допомогою IoU.



Tiny-YOLO проти YOLOv3

Бібліотека OpenCV. Архітектура програми



Порівняння результатів YOLOv3 і tiny-YOLO



Tiny-YOLO

YOLO

Порівняння результатів YOLOv3 і tiny-YOLO

0.7

Тип архітектури	Кількість кадрів з розпізнаванням	Відсоткове відношення (порівняно з 1140)
YOLOv3	959	76.72 %
Tiny-YOLO	86	6.8 %

0.3

Тип архітектури	Кількість кадрів з розпізнаванням	Відсоткове відношення (порівняно з 1140)
YOLOv3	1150	100 %
Tiny-YOLO	1028	89.40 %

Результати обчислень

Апарат	Обчислення першого модулю (секунд/кадр)		FPS
	Сер мін.	Сер макс.	Середня кількість кадрів за секунду
HP	0.1456	0.4219	18.4
Thinkpad	0.095	0.1847	42.68
Colab	0.0093	0.023	112.12
Апарат	Обчислення другого модулю (секунд/кадр)		
	Tiny-YOLO		YOLOv3
HP	0.22		1.56
Thinkpad	0.12		0.8
Colab	0.092		0.08

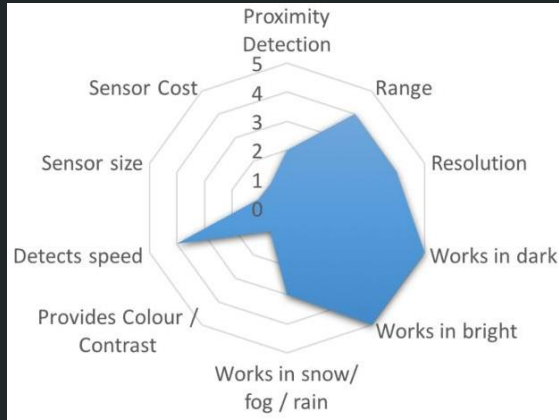
Висновки

- Реалізовано модуль визначення проїзної частини дороги.
- Реалізовано модуль розпізнавання об'єктів.
Проведено аналіз прототипів CNN на різних архітектурах

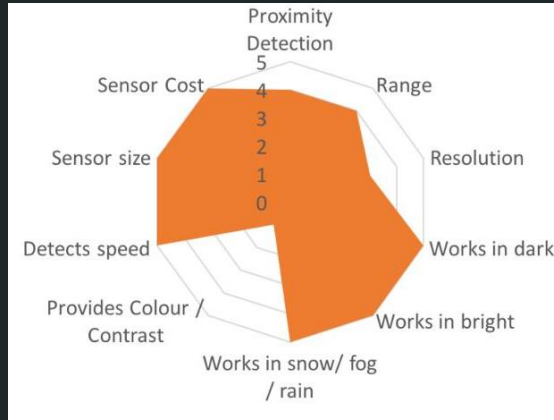
Дякую за увагу

Порівняння можливостей різних видів систем

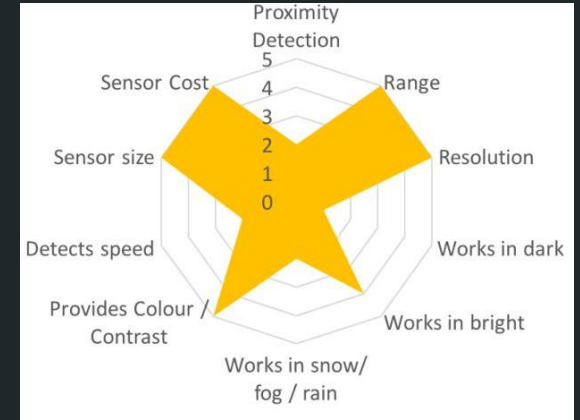
LiDAR



RADAR



Камера



Зміна перспективи зображення

За допомогою даних точок [11] формується система рівнянь, вирішення якої дасть можливість отримати коефіцієнти матриці гомографії.

Побудована СЛАР

$$PH = \begin{bmatrix} -x_1 & -y_1 & -1 & 0 & 0 & 0 & x_1x'_1 & y_1x'_1 & x'_1 \\ 0 & 0 & 0 & -x_1 & -y_1 & -1 & x_1y'_1 & y_1y'_1 & y'_1 \\ -x_2 & -y_2 & -1 & 0 & 0 & 0 & x_2x'_2 & y_2x'_2 & x'_2 \\ 0 & 0 & 0 & -x_2 & -y_2 & -1 & x_2y'_2 & y_2y'_2 & y'_2 \\ -x_3 & -y_3 & -1 & 0 & 0 & 0 & x_3x'_3 & y_3x'_3 & x'_3 \\ 0 & 0 & 0 & -x_3 & -y_3 & -1 & x_3y'_3 & y_3y'_3 & y'_3 \\ -x_4 & -y_4 & -1 & 0 & 0 & 0 & x_4x'_4 & y_4x'_4 & x'_4 \\ 0 & 0 & 0 & -x_4 & -y_4 & -1 & x_4y'_4 & y_4y'_4 & y'_4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} h1 \\ h2 \\ h3 \\ h4 \\ h5 \\ h6 \\ h7 \\ h8 \\ h9 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Гомогенні координати зображення

$$\begin{bmatrix} x'/\lambda \\ y'/\lambda \\ \lambda \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Інтерполяція ліній

Розкриття системи зі слайду 14 «Інтерполяція ліній» дасть можливість виразити умову у вигляді множення матриці на вектор з невідомими коефіцієнтами:

$$\begin{bmatrix} n & \sum_{i=1}^n x_i & \cdots & \sum_{i=1}^n x_i^k \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 & \cdots & \sum_{i=1}^n x_i^{k+1} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^n x_i^k & \sum_{i=1}^n x_i^{k+1} & \cdots & \sum_{i=1}^n x_i^{2k} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_i y_i \\ \vdots \\ \sum_{i=1}^n x_i^k y_i \end{bmatrix}.$$

Шляхом перетворень маємо матричне рівняння, де a – вектор коефіцієнтів інтерполяційного поліному:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^k \\ 1 & x_2 & x_2^2 & \cdots & x_2^k \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^k \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{bmatrix},$$

Конволюційні нейронні мережі

Зображення представляє собою набір пікселів. Нейронна мережа має специфічну будову – ваги шару нейронної мережі з'єднані з наступним шаром у відношенні «всі до всіх». Нехай, навіть, зображення має лише один канал глибини. У випадку використання звичайної нейронної мережі, на відміну від конволюційної, з'являється велика кількість зв'язків між шарами нейронної мережі. Це призводить до значного уповільнення швидкості роботи алгоритмів через наявність великої кількості обчислень

Конволюційні нейронні мережі

На відміну від повнозв'язної мережі, конволюційна нейронна мережа представляє собою структуру, яка складається з шарів конволюції та агрегування даних, що оброблюють зображення до потрапляння до кінцевого повнозв'язного шару. Нейрон в повнозв'язній мережі еквівалентний одиночному структурному елементу в одному з шарів конволюції або агрегації. Операції конволюція і агрегування, які формують ці шари, задіють так звані фільтри, або ядра конволюції і агрегування. В нейронній мережі фільтри конволюції мають значення вагів, як в повнозв'язних шарах; фільтри агрегування допомагають зменшити розмір структури, що оброблюється