

Лабораторна робота №3

Виконали:

Роман Харченко КА-77

Родіон Буханевич КА-77

Синиця Анна КА-75

Назва команди

Saitama San

Дескриптори на розгляді

SIFT, SURF, AKAZE

Предмети на розгляді

Сувенірна підкова, м'яка іграшка синиця, компакт-диск



З минулої лаби залишилось три датасети з зображеннями цих об'єктів. З них створили три тестових датасети. Також додали сет з зображеннями на яких відсутні дані об'єкти.

Завдання:

Маючи досягнення з попередньої серії, докрутити до лаби **класифікатор** на ваш особистий смак, натренувати його на ваш об'єкт (або ж якщо є час і натхнення то на інший), за необхідності розширюючи вибірку, і на об'єкт сусіда з таким же класифікатором на основі алгоритмів отримання характеристик зображення. Після того зробити дві речі: прогнати отримане комбо з класифікатора і дескриптора на тестовій вибірці збираючи як позитивні результати так і помилки першого і другого роду та замірюючи швидкодію; і записати демонстраційне відео з вашим об'єктом та згодувати його цьому ж комбо записуючи у окремий файл результат роботи (скажімо те ж відео з текстовим оверлеєм). Далі знаючи особливості вашого дескриптора колективно з кимсь у кого такий же класифікатор проаналізувати особливості класифікатора в комбінації з вашими дескрипторами. Бонусні очки за використання кількох класифікаторів (видів класифікаторів) і/або запис відео з візуалізацією процесу розпізнавання/детекції.

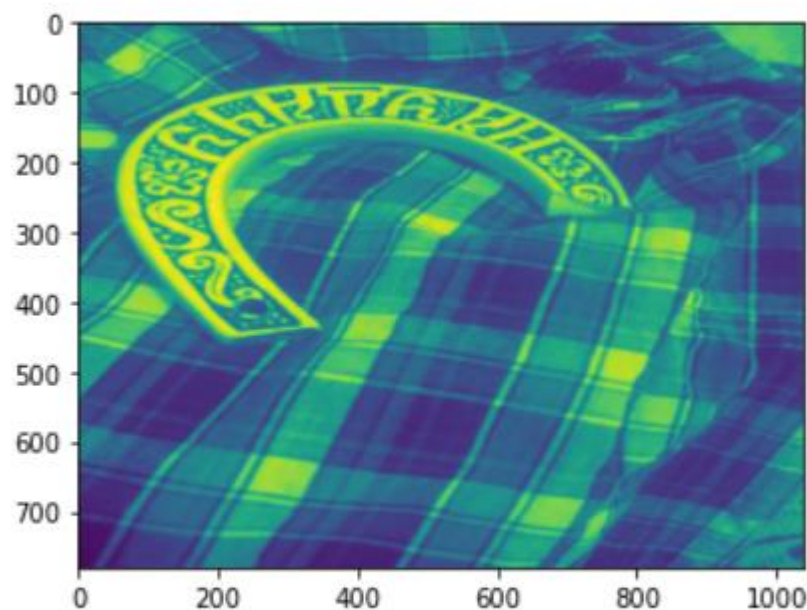
В якості класифікатора ми обрали метод опорних векторів, враховуючи те, що алгоритми детекції, зокрема SURF, SIFT, генерують дуже велику, а також і різну кількість ключових точок, ми використали підхід Bag of words для генерації словника ключових слів та для перетворення нашого набору дескрипторів у єдиний вектор ознак, що й подається на вхід класифікатора.

Ми сформували словник розміром (кількість класів) * 10 візуальних слів, для співставлення дескриптора ключової точки з деяким візуальним словом використовували алгоритм матчіну.

Сам метод опорних векторів було реалізовано таким чином, щоб він отримував на вхід вектор ознак зображення, а на виході видавав число від 1 до 4, де 1 - фото належить до класу зображень з підковою, 2 - фото належить до класу зображень з диском, 3 - фото належить до класу зображень з синицею, 4 - фото належить до класу зображень, що не містить жодного з наших об'єктів. Для розбиття на класи застосовуємо підхід "один проти всіх". В якості ядра для методу опорних векторів було обрано радіально базисні функції, оскільки навряд чи варто сподіватись на лінійну роздільність класів. Інші параметри методу обирались оптимальними за допомогою відповідної функції бібліотеки OpenCV. Тестова і навчальна вибірка містять однакову кількість об'єктів кожного класу.

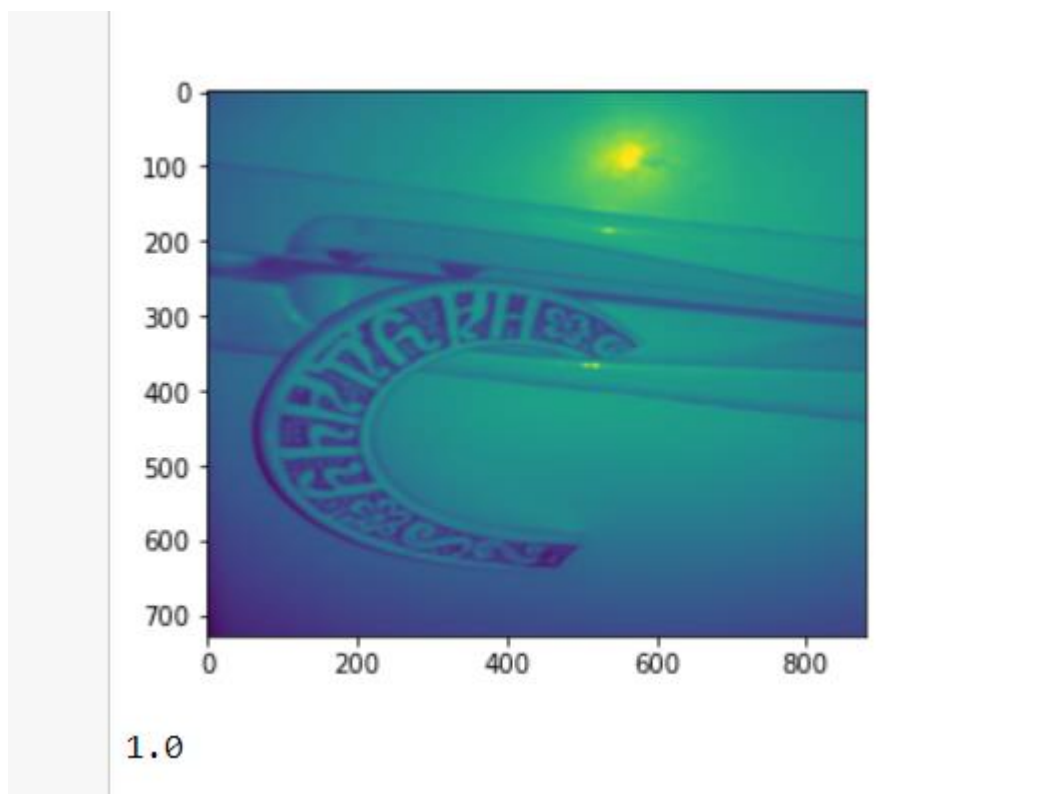
Результатом роботи є по 3 відео для кожного об'єкту.

Розглянемо деякі приклади

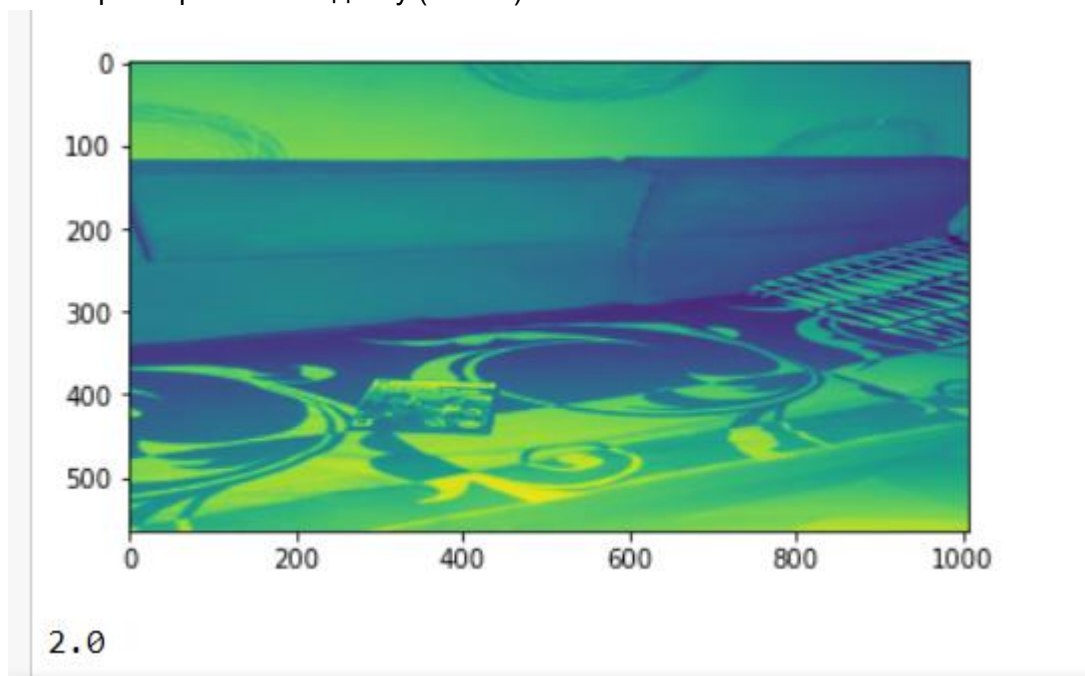


1.0

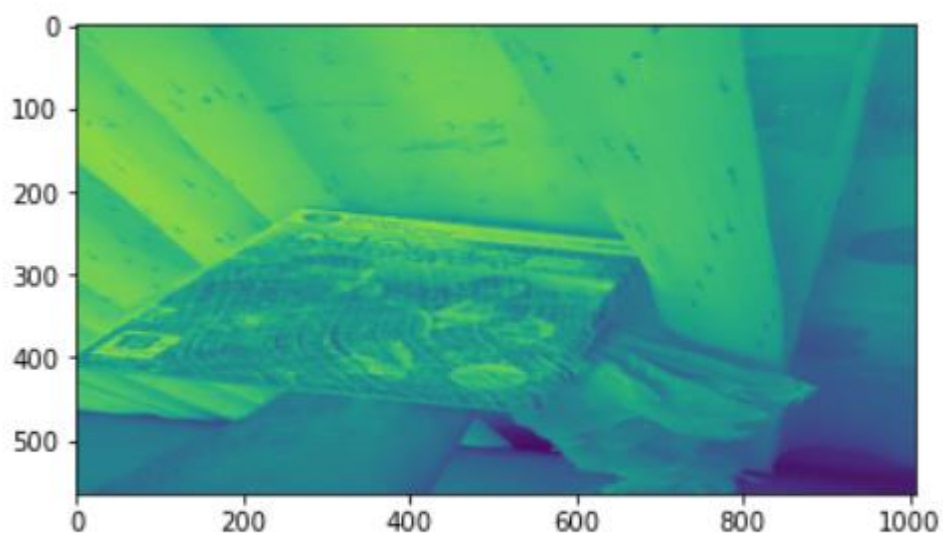
Класифікатор впізнає підкову (клас 1)



Класифікатор впізнає підкову (клас 1)

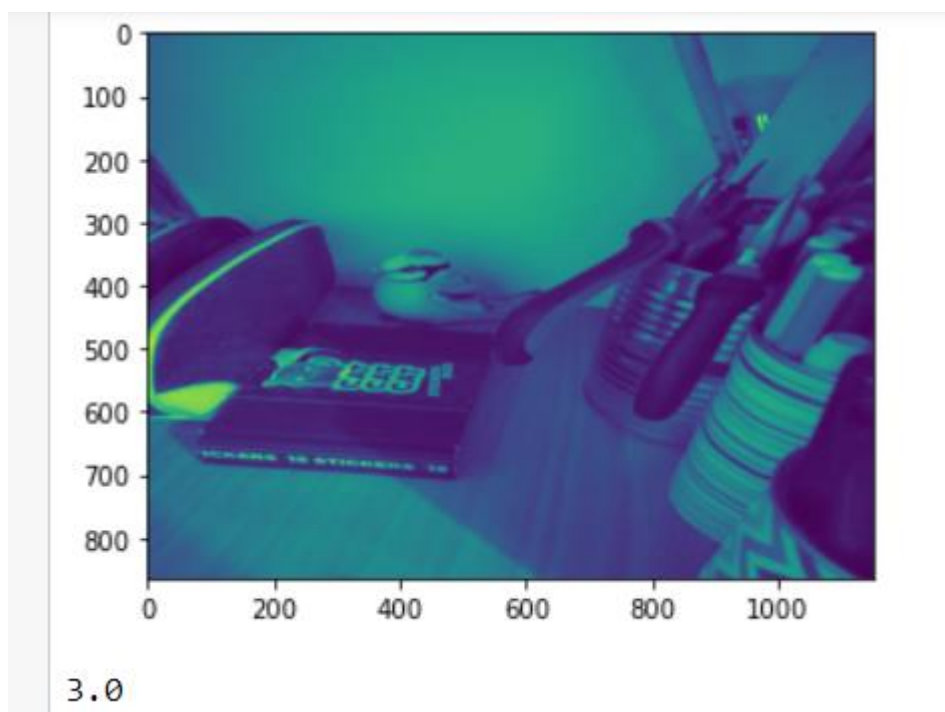


Класифікатор впізнає диск (клас 2)



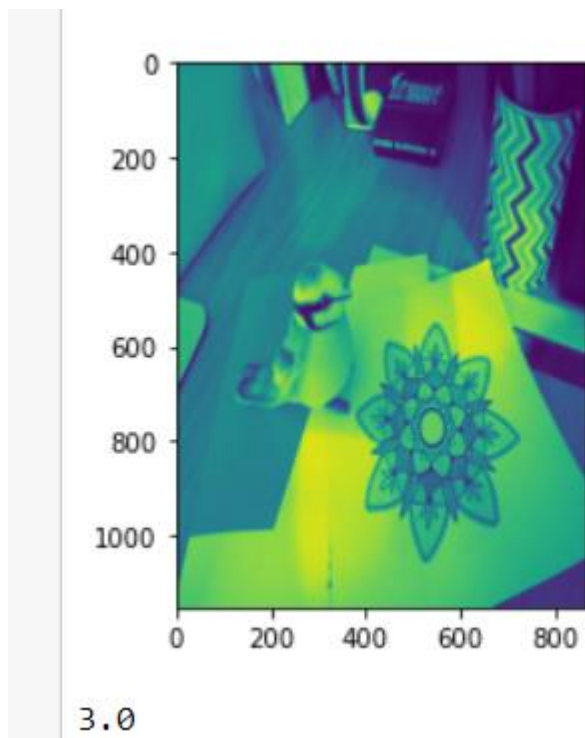
2.0

Класифікатор впізнає диск (клас 2)



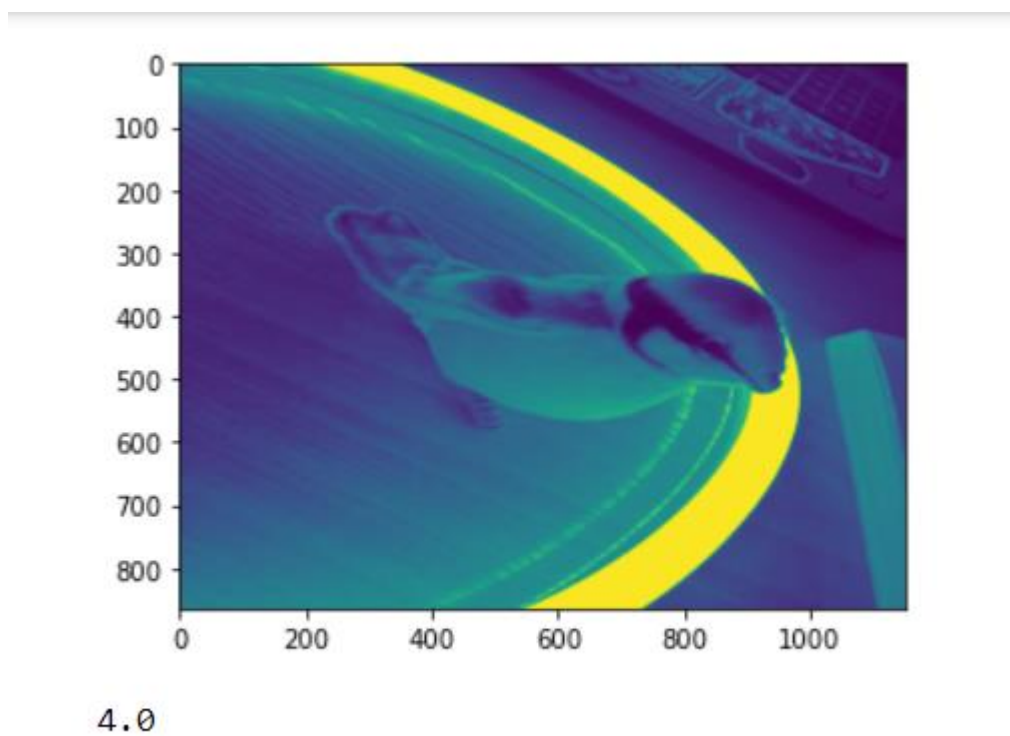
3.0

Класифікатор впізнає синицю (клас 3)

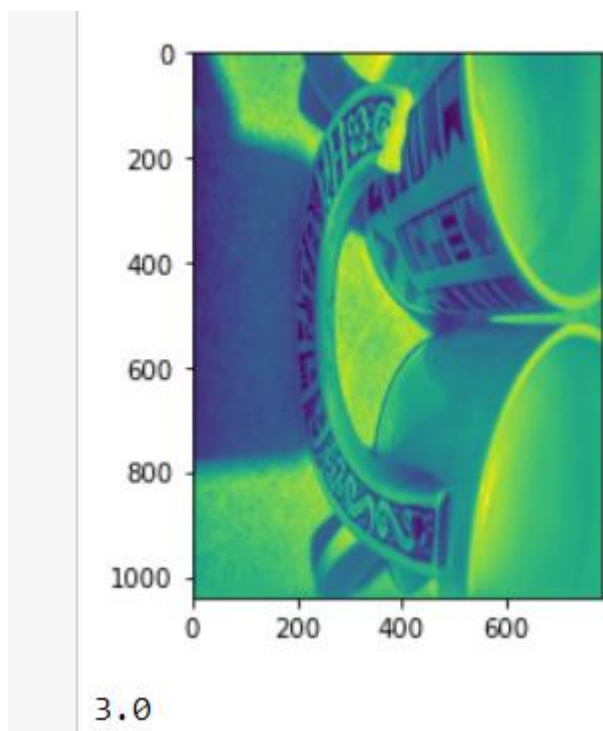


Ще одну синицю впізнав

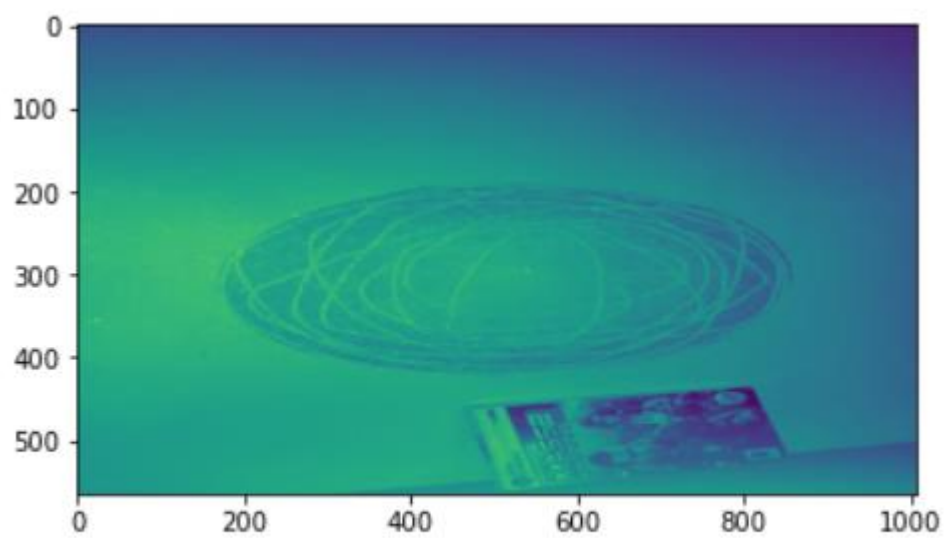
Невдалі приклади



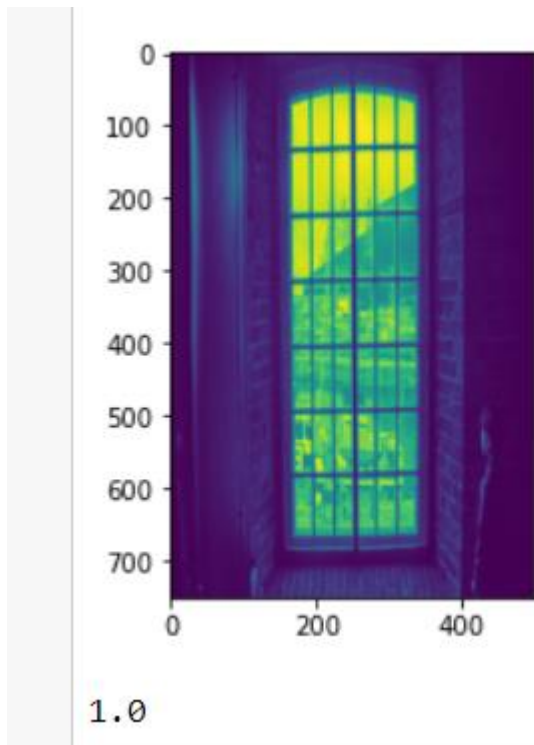
Класифікатор вважає, що тут немає наших об'єктів (клас 4)



Класифікатор вважає, що на фото синиця (клас 3)



Класифікатор вважає, що на фото синиця (клас 3)



Класифікатор вважає, що на фото підкова (клас 1)

Результати, отримані з csv файлів:

AS_dataset_SURF

Середній час обробки зображення - 0,86 с.

Кількість помилок першого роду - 27 з 40.

RK_dataset_SURF

Середній час обробки зображення - 0,7029 с.

Кількість помилок першого роду - 19 з 40 .

RB_dataset_SURF

Середній час обробки зображення - 0,769885 с.

Кількість помилок першого роду - 16 з 40.

noobj_dataset_SURF

Середній час обробки зображення - 1,012122 с.

Кількість помилок другого роду - 16 з 40.

Метод опорних векторів в комбінації з дескриптором SURF відносно непогано впорався з набором фото першого та другого класу і погано впорався з фото третього класу, що можна пояснити тим, що деякі зображення з категорії 4 - фото без об'єктів є схожими на фото 3 класу, до того ж об'єкт синиця є об'ємним та досить складним для розпізнавання, класифікатор також часто плутав цей об'єкт з об'єктами інших класів. Середній час обробки зображень приблизно однаковий для класів 1,2,3, для класу 4 трохи більший, що можна пояснити тим, що в цій категорії фото як правило містять багато інших різних об'єктів.

AS_dataset_AKAZE

Середній час обробки зображення - 0,624353 с.

Кількість помилок першого роду - 27 з 40.

RK_dataset_AKAZE

Середній час обробки зображення - 0,6442843 с.

Кількість помилок першого роду - 17 з 40.

RB_dataset_AKAZE

Середній час обробки зображення - 0,437552 с.

Кількість помилок першого роду - 16 з 40.

Noobj_dataset_AKAZE

Середній час обробки зображення - 0,7726825 с.

Кількість помилок другого роду - 22 з 40.

Метод опорних векторів в комбінації з дескриптором AKAZE працює швидше, але кількість помилок другого роду більша в порівнянні з SURF. Кількість помилок першого роду практично співпадає з результатами для дескриптора SURF. Найбільше помилок на фото класу 3, що підтверджує складність розпізнавання об'єкту синиця.

AS_dataset_SIFT

Середній час обробки зображення - 0,7208455 с.

Кількість помилок першого роду - 25 з 40.

RK_dataset_SIFT

Середній час обробки зображення - 0,63128 с.

Кількість помилок першого роду - 11 з 40.

RB_dataset_SIFT

Середній час обробки зображення - 0,4925344 с.

Кількість помилок першого роду - 15 з 40.

Noobj_dataset_SIFT

Середній час обробки зображення - 0,964714 с.

Кількість помилок другого роду - 19 з 40.

Метод опорних векторів в комбінації з дескриптором SIFT допустив найменше помилок першого роду, до того ж працює досить швидко. Кількість помилок другого роду трохи більша ніж у SURF та менша ніж у AKAZE.

Тренування нейронної мережи FasterRcnn, яка детектить об'єкти на фото, та результати роботи в ноутбучі Lab3_FasterRCNN.ipynb

Висновки

Практично всі дескриптори в комбінації з методом опорних векторів мають відносно невелику різницю в середнього часу обробки фото та показують кращі результати на одних і тих же класах зображень і відповідно гірші теж на одних і тих самих класах. Успішність класифікації залежить від складності досліджуваного об'єкта, складності фону, наявності перешкод. Метод опорних векторів також найкраще підходить для задач бінарної класифікації, а для розбиття на кілька класів працює гірше. Ще одним фактором, який вплинув на результат є необхідність зменшення розмірів фото програмно, оскільки вони були надто великими. Нейрона мережа FasterRcnn працює набагато краще ніж детектори опенсв, але потрібен більший час на її тренування.