

Візуально-мовні моделі на вбудованих пристроях

Виконав:

студент МП КН-2

Росада В.В.

Науковий керівник:

старший викладач, к.т.н.

Поздняков Є.К.

Актуальність

Приклади задач, що вирішують візуально-мовні моделі:

- Відповіді на запитання за візуальним контентом
- Семантичний пошук зображень за текстовим описом
- Класифікація зображень

Переваги роботи візуально-мовних моделей на вбудованих пристроях:

- Відсутність потреби в постійному інтернет-з'єднанні
- Забезпечення конфіденційності та безпеки даних
- Відсутність серверних експлуатаційних витрат
- Швидкість обробки даних

Приклади сфер застосування: медицина, дрони, охоронні системи.

Мета

Дослідження та розробка системи для семантичного пошуку візуальної інформації за текстовим описом на вбудованому пристрої з використанням візуально-мовних моделей.

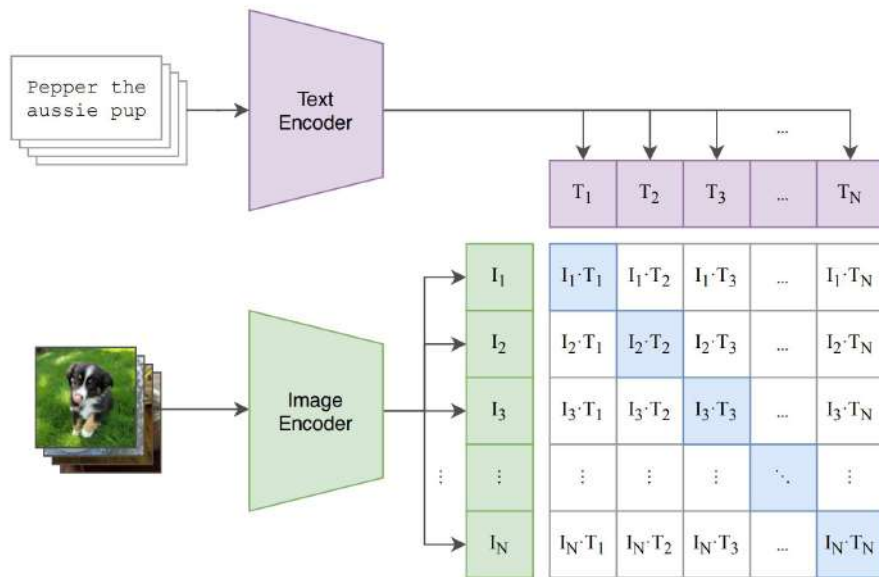
Щоб досягти мети:

- Провести порівняльний аналіз продуктивності та точності візуально-мовних моделей на вбудованому пристрої
- Розробити програмний застосунок для семантичного пошуку зображень за текстовим запитом на пристрої з апаратним прискорювачем.

Виклики

- Обмежені обчислювальні ресурси
- Обмежений обсяг пам'яті
- Вимоги до швидкості роботи системи в реальному часі.
- Необхідність адаптації моделей для пристроїв з апаратним прискорювачем

Моделі контрастивного навчання



<https://arxiv.org/pdf/2103.00020>

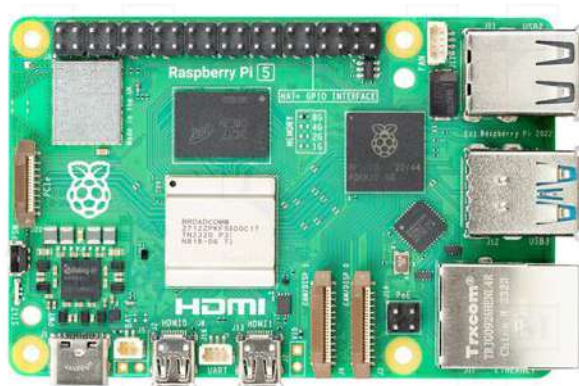
Датасет для тестування

Було використано 500 зображень з Flickr30k



1. A boat goes out to sea while someone watches on the beach
2. An orange boat leaves the shore while a person watches nearby
3. An orange lifeboat speeds away from its mobile launcher whilst a man in red watches it sail away
4. A vehicle has just released a boat into the water as a man in red stands on shore
5. The boat rushed off

Пристрої



Raspberry Pi 5:
CPU: quad-core Arm Cortex A76 processor @
2.4GHz



Ambarella CV75s:
CPU: dual Arm® Cortex-A76 1.6GHz cores
CVFlow 3.0 AI Accelerator

Формат запуску моделей

Raspberry Pi 5

- Моделі конвертувались в ONNX формат

Ambarella CV75s:

- Моделі конвертувались у спеціалізований формат для Ambarella Neural Vector Processor (NVP) за допомогою спеціального інструментарію для цієї платформи

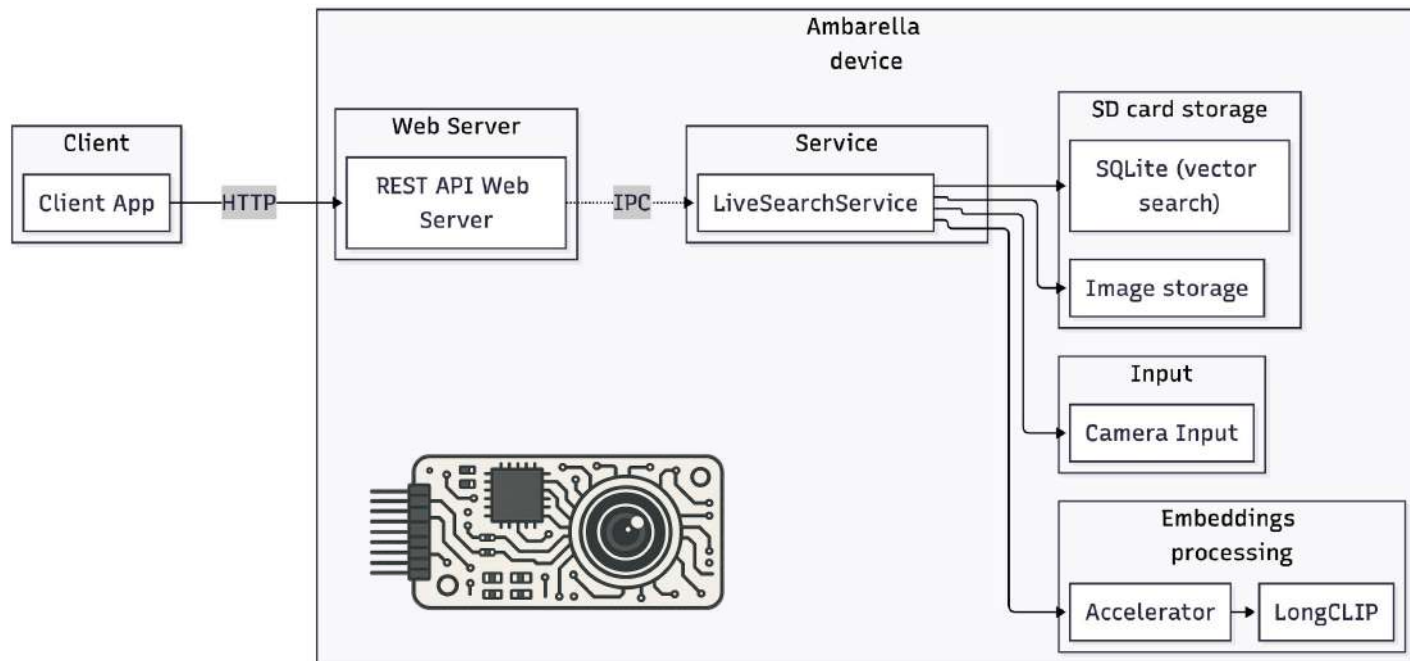
Результати порівняння на Raspberry Pi 5

Метрика	CLIP ViT-B/16 FP32	TinyCLIP ViT8-M16 FP32	LongCLIP ViT-B/16 FP32	CLIP ViT-B/16 INT8	TinyCLIP ViT8-M16 INT8	LongCLIP ViT-B/16 INT8
Recall@1 (%)	67.28	51.64	75.2	54.4	51.04	62.48
Recall@5 (%)	91.16	81.6	94.5	82.4	80.84	88.1
Recall@10 (%)	96.24	90	97.9	90.24	89.7	93.5
MRR	0.7753	0.6571	0.8383	0.6663	0.6403	0.7346
Average Image Embedding Time (ms)	547	59	541	171	33	174
Average Text Embedding Time (ms)	102	8	105	37	4	40

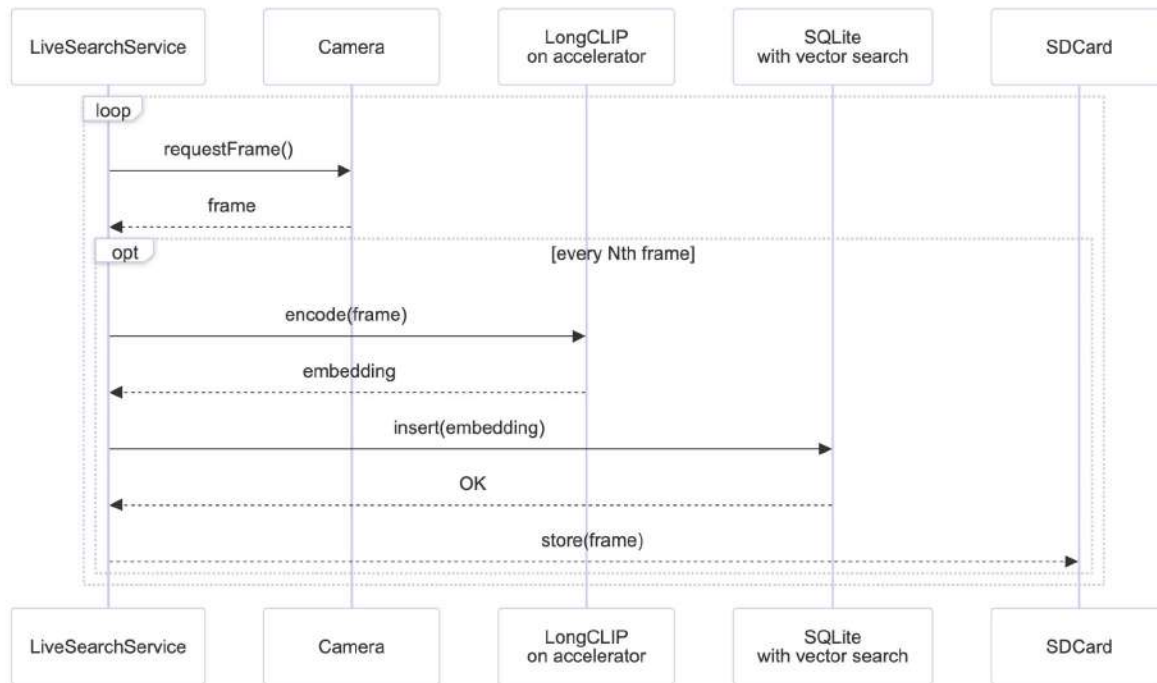
Розробка застосунку. Технологічний стек

- Мова програмування C++
- SDK для роботи з пристроєм та його векторним прискорювачем
- База даних SQLite та розширення sqlite-vec
- Візуально мовна модель LongCLIP адаптована для роботи з апаратним прискорювачем Ambarella

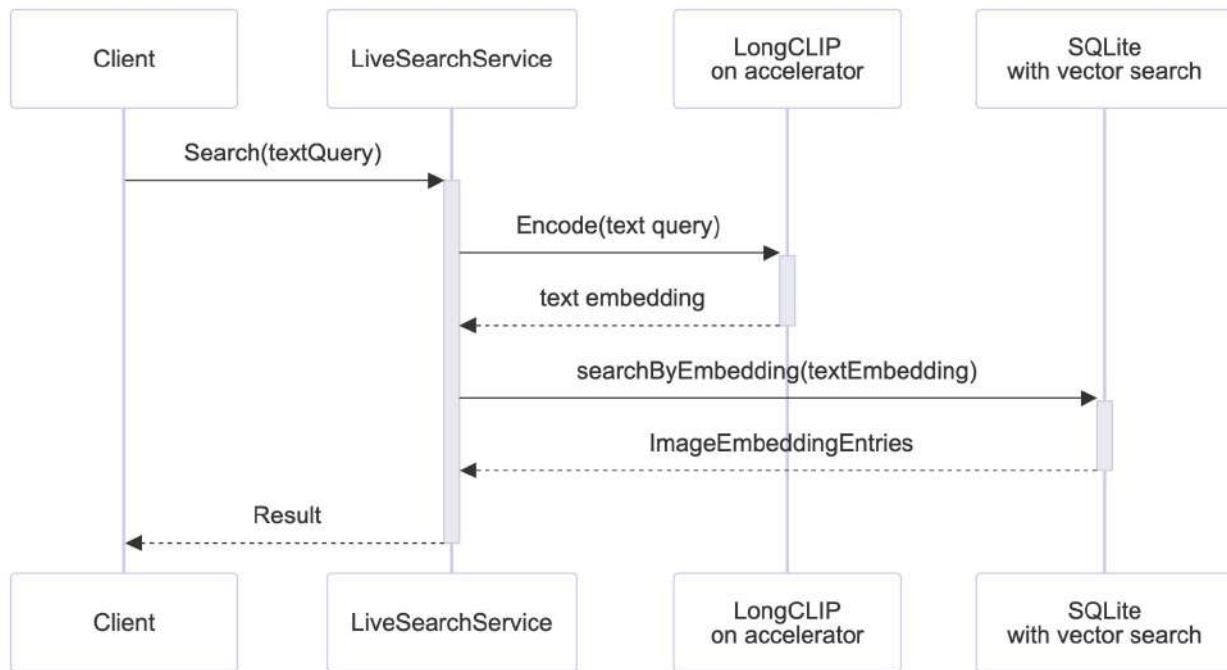
Архітектура застосунку



Діаграма обробки кадрів з камери

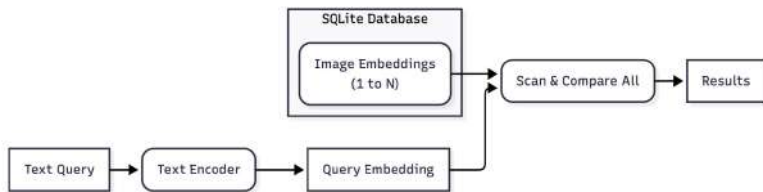


Діаграма виконання пошукового запиту



Оптимізація пошуку

Повне сканування бази (початковий підхід)

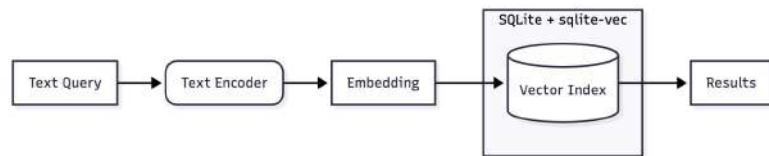


Послідовне порівняння з кожним ембедінгом зображення в базі.

Недоліки:

- Низька продуктивність
- Погана масштабованість

Векторний пошук з розширенням sqlite-vec

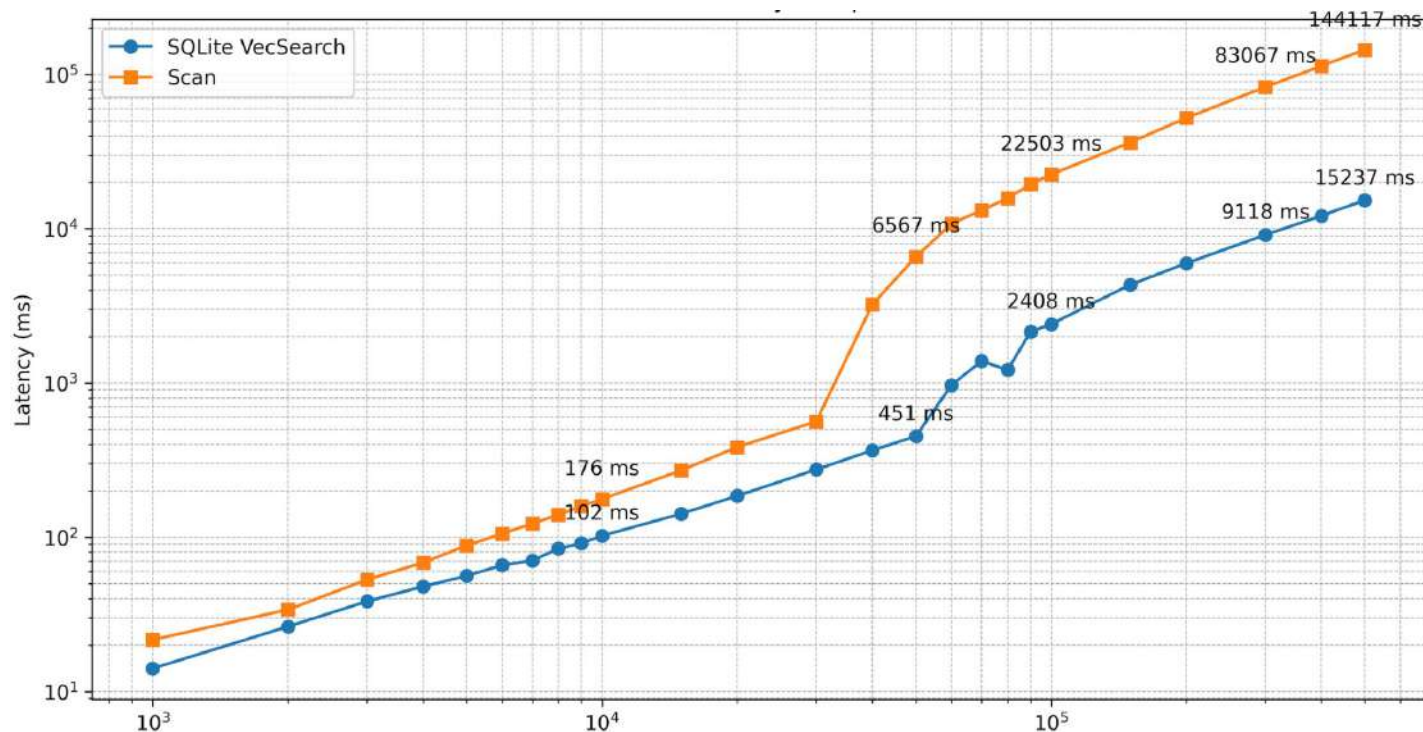


Використання розширення, яке підтримує векторний індекс та пошук

Переваги:

- Хороша продуктивність
- Висока масштабованість

Порівняння векторного пошуку зі скануванням



Приклади роботи

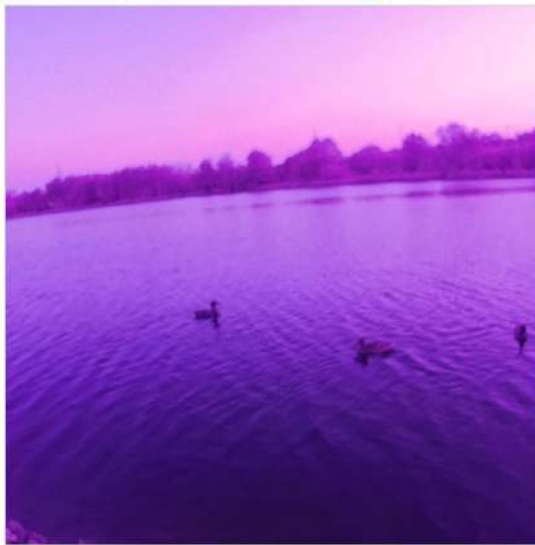
painting of racoon on a stack of books

Search



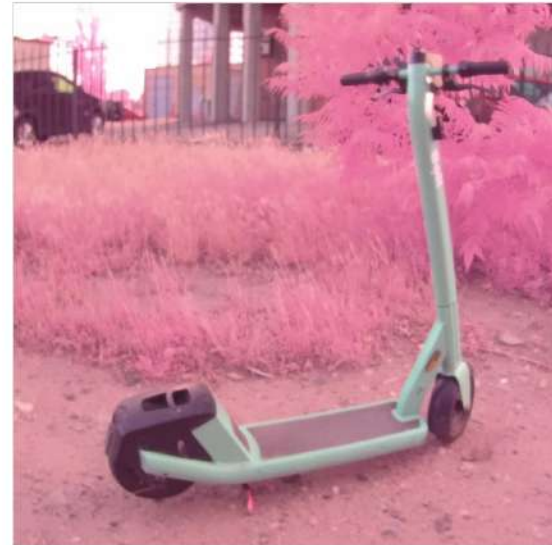
ducks swimming in a lake

Search



parked scooter

Search



Висновки

- Проведено порівняльний аналіз швидкодії та точності сучасних візуально-мовних моделей на вбудованому пристрої
- Розроблено C++ застосунок з використанням LongCLIP для задачі семантичного пошуку візуальної інформації за текстовим описом. Застосунок працює повністю на вбудованому пристрої з Ambarella CV75s, та використовує можливості його апаратного прискорювача.
- Інтегровано та протестовано використання розширення sqlite-vec для SQLite на вбудованому пристрою, що дозволяє здійснювати векторний пошук, що дає кратний приріст в продуктивності пошуку в порівнянні з повним скануванням на великих обсягах даних. Це зробило систему масштабованою.

Дякую за увагу