**HCM AI CHALLENGE 2023**

1. Data preprocessing

A white rectangle with black text

Description automatically generated

* Sử dụng TransNetV2 để trích xuất ra các phân đoạn chuyển cảnh và lấy thêm keyframe trong từng cảnh trong bộ dữ liệu do ban tổ chức cung cấp. Đối với data batch 1, nhóm sẽ lấy thêm 3 keyframes (tổng cổng 5 keyframes ở một scenes), đối với batch 2 và batch 3 sẽ lấy thêm 1 (tổng cộng 3 keyframes ở một scenes)
* Công thức

+ Index frame đầu: F+(F-L)\*0,

+ Index frame 0.25: F+(F-L)\*0.25,

+ Index frame 0.5: F+(F-L)\*0.5,

+ Index frame 0.75: F+(F-L)\*0.75 và

+ Index frame cuối: F+(F-L)\*1

Với: F là vị trí frame đầu của scene

L là vị trí frame cuối của scene

* Nhóm sử dụng Google Drive để lưu dữ liệu (2TB). Đối với local thì sử dụng ổ cứng rời 512GB.

1. Indexing

A screenshot of a computer

Description automatically generated

* Sử dụng cosine similarity hoặc L2 similarity kết hợp với mô hình pre-trained ViT-B/32 của CLIP. Đầu vào là hình ảnh hoặc text - > Đầu ra là vector. Và lưu dưới dạng .NPY
* Sau khi đã encode tất cả keyframes trong database thì sẽ đưa qua mô hình Faiss của Meta để thực hiện tìm kiếm độ tương tự và gom dữ liệu lớn. Đầu vào là các vector biểu diễn trong không gian nhiều chiều -> Đầu ra: danh sách keyframes và index chưa top các feature vectors của vector đầu vào. Vào lưu vào file .bin cho tất cả 3 batch

1. Tìm kiếm

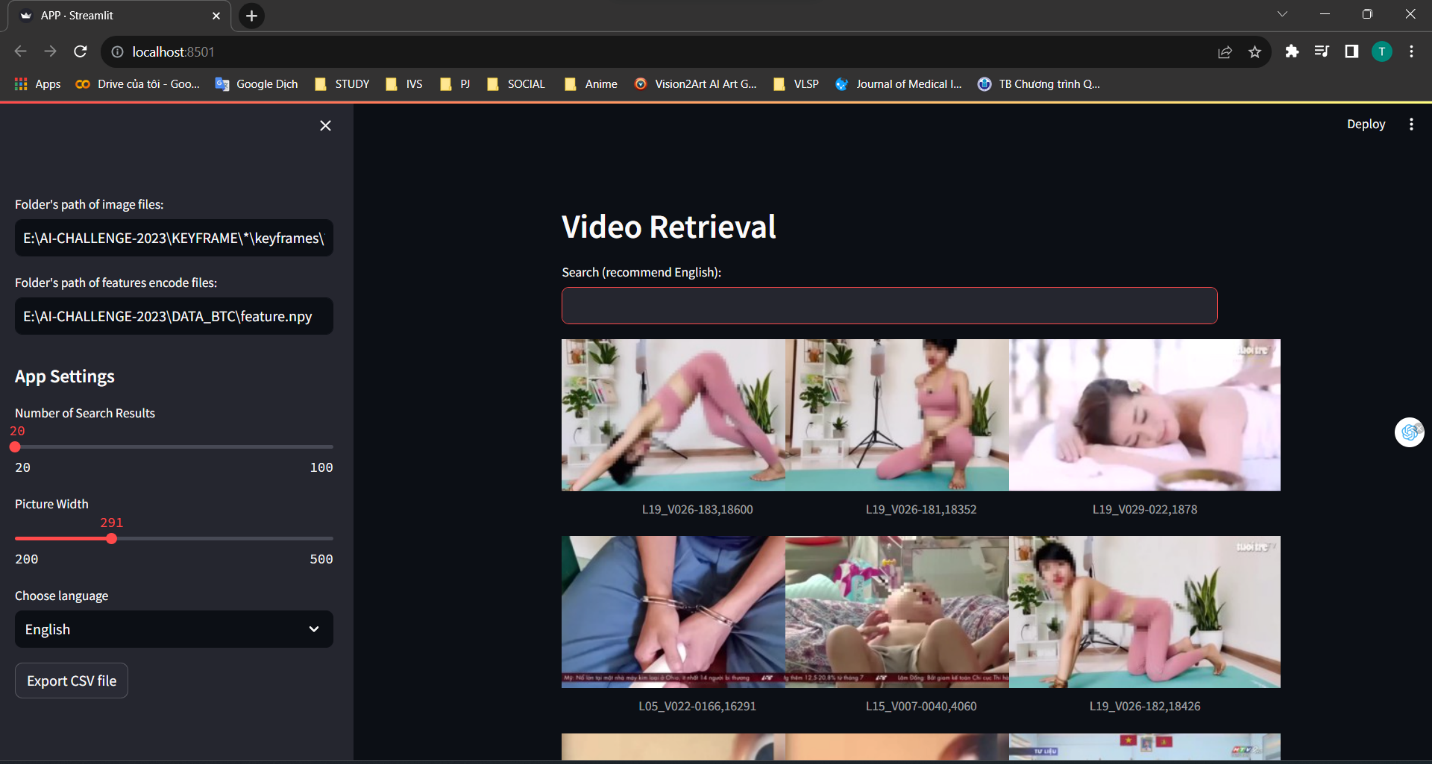
* Tìm kiếm bằng query đầu vào bằng text query.
* Quy trình đầu ra: sau khi text đưa vào sẽ đưa qua Faiss processing để tìm kiếm các tương đồng trong bộ dữ liệu thông qua file.bin và trả về các keyframe có độ tương đồng cao nhất với câu query (chọn k = 100 -> sẽ cho ra 100 hình ảnh)

1. Format

File Json sẽ có format:

|  |
| --- |
| {"0": ["L01\_V001-0000", "Data**\\**Keyframes\_L01**\\**L01\_V001**\\**0000.jpg"],  "1": ["L01\_V001-0001", "Data**\\**Keyframes\_L01**\\**L01\_V001**\\**0001.jpg"],  ...  } |

1. Web



**Hình 1. Demo web để tìm kiếm hình ảnh**