

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC VÀ KỸ THUẬT THÔNG TIN



BÁO CÁO LAB 4: SPARK

GVHD: CN. Nguyễn Hiếu Nghĩa

Lớp: IE212.Q11

Sinh viên thực hiện:

Họ và tên: Trương Hoài Bảo

MSSV: 22520126

☞ Tp. Hồ Chí Minh, 01/2026 ☞

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

[illegible]

....., ngày.....tháng.....năm 2026

Người nhận xét

(Ký tên và ghi rõ họ tên)

MỤC LỤC

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN ii

DANH MỤC HÌNH ẢNH..... iv

NỘI DUNG THỰC HÀNH..... 5

1.1Cài đặt requirement:..... 5

1.2Camera Server:..... 5

1.3Processor Server: 6

TÀI LIỆU THAM KHẢO 8

DANH MỤC HÌNH ẢNH

Hình 1.1-1 Set env	5
Hình 1.2-1 Camera config	5
Hình 1.2-2 Chạy server camera	6
Hình 1.3-1 Cấu hình spark.....	6
Hình 1.3-2 Chạy server processor	7
Hình 1.4-1 Kết quả thực hiện	7

NỘI DUNG THỰC HÀNH

1.1 Cài đặt requirement:

Sử dụng các thư viện cho phù hợp cho bài lab như:

opencv-python

numpy

mediapipe>=0.10

pyspark==3.5.6

py4j

pytest

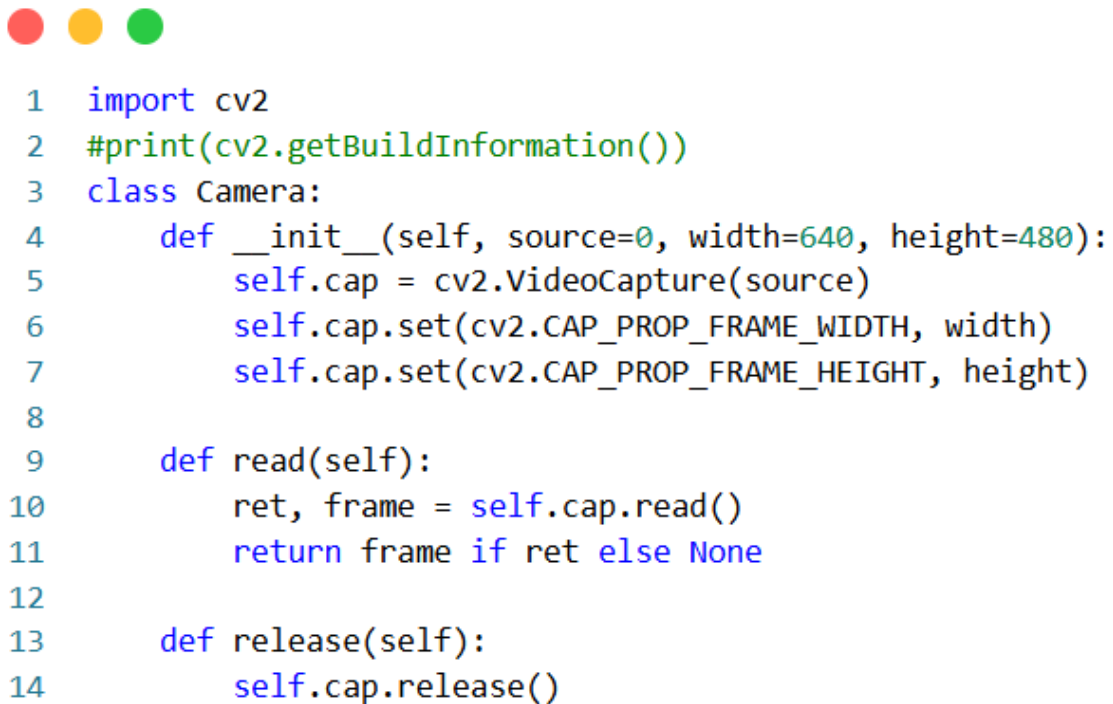
Và ngoài ra còn lưu ý một điều là để đảm bảo spark tương thích với python thì nên set env là Python 3.10.

```
1 import os
2 os.environ["PYSPARK_PYTHON"] = r"C:\Users\DELL\AppData\Local\Programs\Python\Python310\python.exe"
3 os.environ["PYSPARK_DRIVER_PYTHON"] = r"C:\Users\DELL\AppData\Local\Programs\Python\Python310\python.exe"
```

Hình 1.1-1 Set env

1.2 Camera Server:

Thực hiện cấu hình cho camera.



```
1 import cv2
2 #print(cv2.getBuildInformation())
3 class Camera:
4     def __init__(self, source=0, width=640, height=480):
5         self.cap = cv2.VideoCapture(source)
6         self.cap.set(cv2.CAP_PROP_FRAME_WIDTH, width)
7         self.cap.set(cv2.CAP_PROP_FRAME_HEIGHT, height)
8
9     def read(self):
10         ret, frame = self.cap.read()
11         return frame if ret else None
12
13     def release(self):
14         self.cap.release()
```

Hình 1.2-1 Camera config

Khởi chạy server camera

IE108.O21 – Báo cáo đồ án cuối môn

```
Bigdata-LAB > Lab 4 > camera_server > main.py > ...
1 import cv2
2 from camera_server.camera import Camera
3 from camera_server.frame_sender import FrameSender
4 import time
5
6 time.sleep(0.05)
7 camera = Camera(0, 240, 280)
8 sender = FrameSender("localhost", 9999)
9
10 try:
11     while True:
12         frame = camera.read()
13         if frame is None:
14             break
15
16         cv2.imshow("Camera Client", frame)
17
18         if cv2.waitKey(1) & 0xFF == ord('q'):
19             break
20
21         sender.send(frame)
22
23 except KeyboardInterrupt:
24     pass
25
26 camera.release()
27 sender.close()
28 cv2.destroyAllWindows()
```

Hình 1.2-2 Chạy server camera

1.3 Processor Server:

Khởi tạo và cấu hình spark, sau đó viết hàm để xử lý các packets:

```
Bigdata-LAB > Lab 4 > processing_server > spark_processor.py > process_frames
1 from pyspark.sql import SparkSession
2 from processing_server.background_remover import remove_background
3 from processing_server.saver import save_frame
4 # create spark session
5 spark = SparkSession.builder \
6     .appName("RMBG-Lab4") \
7     .master("local[1]") \
8     .getOrCreate()
9
10 def process_frames(packets):
11     rdd = spark.sparkContext.parallelize(range(len(packets)))
12     rdd.foreach(lambda _: None)
13
14     results = []
15     for packet in packets:
16         frame = packet["frame"]
17         frame_id = packet.get("frame_id", 0)
18
19         result = remove_background(frame)
20         save_frame(result, frame_id)
21
22         results.append(result)
23
24     print(f"Processed {len(results)} frames with SparkSession")
25     return results
26
```

Hình 1.3-1 Cấu hình spark

Hàm main để khởi tạo server:

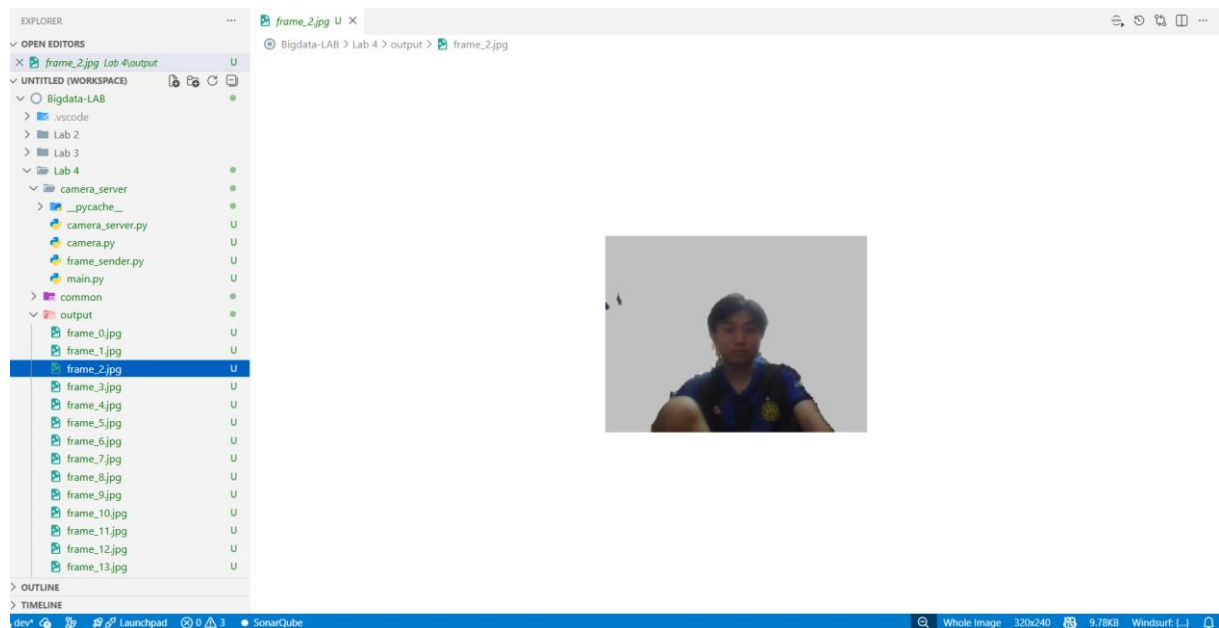
IE108.O21 – Báo cáo đồ án cuối môn

```
Bigdata-LAB > Lab 4 > processing_server > main.py > ...
1 import os
2 os.environ["PYSPARK_PYTHON"] = r"C:\Users\DELL\AppData\Local\Programs\Python\Python310\python.exe"
3 os.environ["PYSPARK_DRIVER_PYTHON"] = r"C:\Users\DELL\AppData\Local\Programs\Python\Python310\python.exe"
4 from processing_server.receiver import start_server
5 from common.protocol import unpack_frame
6 from processing_server.spark_processor import process_frames
7 conn = start_server()
8 buffer = []
9
10 try:
11     while True:
12         packet = unpack_frame(conn)
13         if packet is None:
14             break
15
16         buffer.append(packet)
17
18         if len(buffer) >= 10: # micro-batch
19             process_frames(buffer)
20             buffer.clear()
21
22 except KeyboardInterrupt:
23     print("Processing stopped")
24
25 conn.close()
26
```

Hình 1.3-2 Chạy server processor

1.4 Kết quả:

Kết quả sẽ được lưu ở file output.



Hình 1.4-1 Kết quả thực hiện

TÀI LIỆU THAM KHẢO