

Warszawa, 20.06.2023r.

Politechnika Warszawska

Wydział Elektroniki i Technik Informacyjnych

Analiza semantyczna obrazu

Projekt

Dokumentacja końcowa

Zespół:

Jakub Sobolewski 300371

Piotr Paturej 304102

Jakub Płudowski 300464

## 1. Temat projektu

Tematem projektu jest program do nadzorowania kolejek. Program taki powinien wykorzystywać wytrenowany algorytm typu MOT (Multiple Object Tracking) do analizowania struktury kolejki. Kolejka powinna posiadać punkt docelowy (okienko, ladę lub jakiegokolwiek stoisko) i powinna składać się z min 5-6 osób stojących gęsiego. Program taki powinien na wejściu pobierać film (lub zestaw zdjęć w określonej kolejności), przeanalizować go i zwrócić raport w postaci analizy tego co działo się ze strukturą kolejki w trakcie filmu. Program powinien monitorować długość kolejki (dyskretną liczbę osób składających się na kolejkę) oraz raportować zdarzenia takie jak opuszczenie kolejki w postaci obsłużenia osoby oczekującej (osoba najbliższej punktu docelowego opuszcza kolejkę) lub rezygnacji (osoba która nie jest najbliższej opuszcza kolejkę) oraz dołączenie się do kolejki w postaci dołączenia do ogona kolejki lub wepchnięcia się w jej strukturę. Program taki dodatkowo może posiadać funkcję liczenia czasu ile dana osoba czekała w kolejce (czas opuszczenia kolejki - czas dołączenia do kolejki).

## 2. Wstęp teoretyczny

Algorytm MOT jest algorytmem do śledzenia wielu obiektów. Sam problem śledzenia obiektu polega na wyszukaniu w klatce filmu (obrazie) konkretnego obiektu, przypisania mu unikalnego ID oraz predykcji tego gdzie obiekt się przesunie. W przypadku MOT mamy do czynienia z analizą wielu obiektów równolegle. Proces ten zazwyczaj składa się z dwóch etapów. Pierwszym z nich jest detekcja obiektów w obrazie oraz znalezienie ich cech, a drugim jest predykcja zachowania obiektów na następnej klatce na podstawie dostępnych informacji.

## 3. Wykorzystywany zbiór danych: MOT20

Nasza praca nad projektem rozpoczęła się od poszukiwania odpowiedniego zbioru danych. Po przeglądzie dostępnych zbiorów, zdecydowaliśmy się wykorzystać zbiór MOT20, dostępny na stronie MOT Challenge.

MOT20 to kompleksowy zestaw danych przeznaczony do ewaluacji systemów śledzenia obiektów w scenach monitoringu miejskiego. Zawiera siedem długotrwałych sekwencji wideo, zarejestrowanych w różnych warunkach oświetleniowych i pogodowych. Zbiór ten zawiera zarówno sekwencje treningowe, jak i testowe, które obejmują różne scenariusze, takie jak ruch uliczny, ruch pieszy i tłumy. Każda sekwencja wideo jest starannie opisana, z identyfikacją unikalnych jednostek (osób) w każdej klatce.

#### 4. Wstępne eksperymenty: Dotrenowanie modelu FasterRCNN ResNet 50

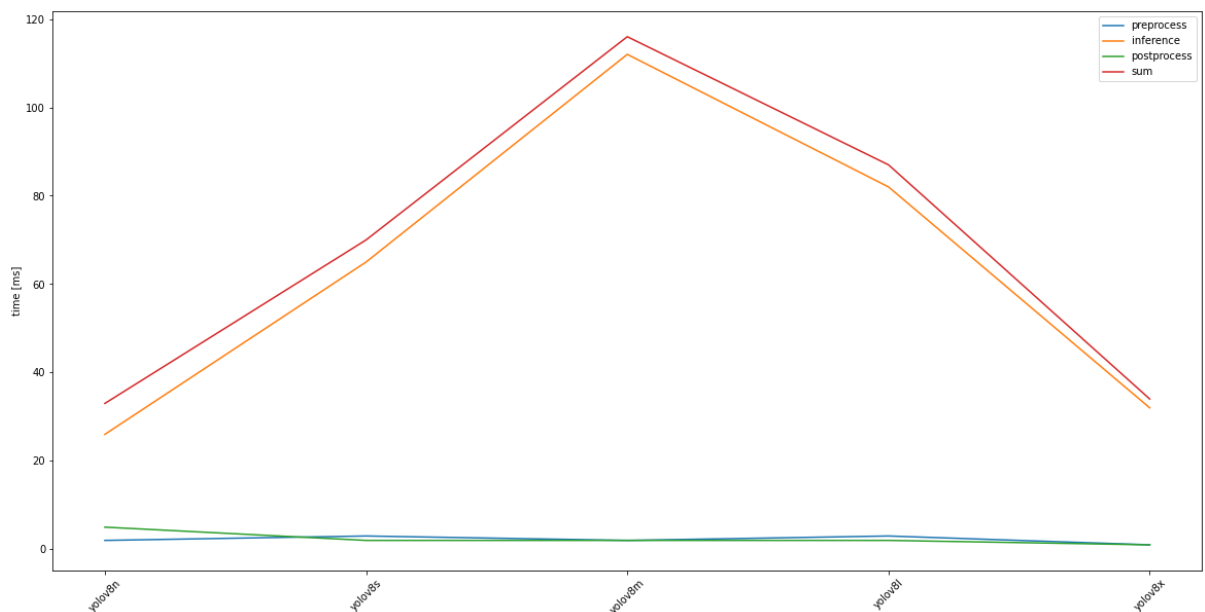
Po wyborze zbioru danych, naszym następnym krokiem było próba dotrenowania istniejącego modelu MOT, w tym przypadku FasterRCNN ResNet 50, na danych z MOT20. Celem było dostosowanie modelu do prawidłowego wykrywania ludzi na obrazach z kamery. Niestety, wyniki uzyskane w wyniku tego procesu nie były zadowalające. Okazało się, że model wykrywał tylko niewielką część ludzi na obrazach. Co więcej, model zaznaczał również kilka obszarów, w których nie było żadnego człowieka. Przy analizie wyników, zauważyliśmy, że model miał tendencję do "widzenia" ludzi tam, gdzie ich nie było, co sugeruje, że model mógł być nadmiernie dopasowany do danych treningowych lub mógł nie radzić sobie dobrze z różnorodnością scenariuszy zawartych w zbiorze MOT20.

#### 5. Ostateczny model YOLO

Ze względu na słabe wyniki rekurencyjnej sieci zdecydowano wykorzystać już wytrenowany model YOLO. Jest to rodzina architektur głębokich sieci neuronowych zawierająca wiele modeli. Sprawdzono 5 modeli o poniższych parametrach:

Model	size (pixels)	mAP <sup>val</sup> 50-95	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	params (M)	FLOPs (B)
YOLOv8n	640	37.3	80.4	0.99	3.2	8.7
YOLOv8s	640	44.9	128.4	1.20	11.2	28.6
YOLOv8m	640	50.2	234.7	1.83	25.9	78.9
YOLOv8l	640	52.9	375.2	2.39	43.7	165.2
YOLOv8x	640	53.9	479.1	3.53	68.2	257.8

Przetestowano wydajność pięci modeli i wybrano te, które najwydajniej się zachowywały czyli YOLOv8n oraz YOLOv8x zgodnie z poniższym wykresem:



## 6. Ostateczny dane, których użyto

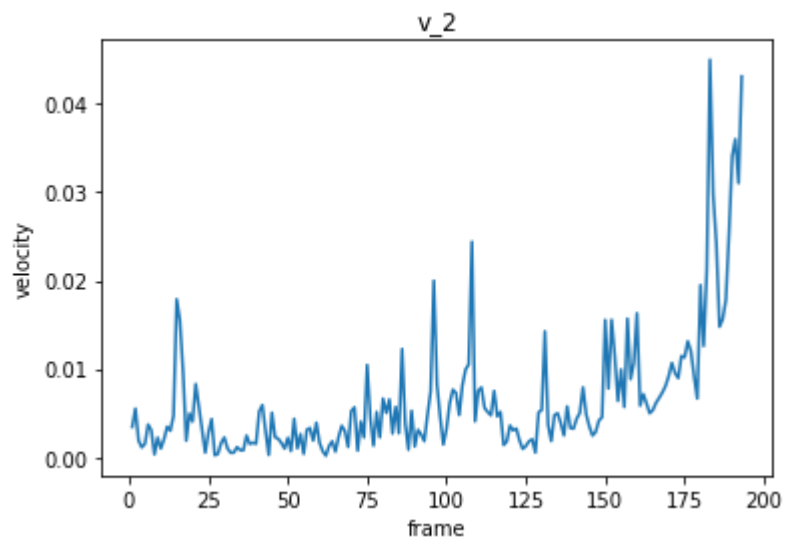
Ze względu na to, że model YOLO nie jest trenowany na danych pochodzących z kamer przemysłowych, dane z MOT20 niezbyt się nadawały do tego modelu. Z tego powodu poszukiwano danych za pomocą scrapowania stron zawierających inurl: 'webcam xp5'. Zdobyto w ten sposób dane z kamer. Niestety większość z nich nie dawała dobrych rezultatów, ponieważ YOLO niezbyt dobrze radzi sobie z perspektywą i widokiem z góry. Na

szczęście udało się zdobyć dane z platformy shutterstock, na której udało się stworzyć algorytmy, które zostały w ostatecznej formie.

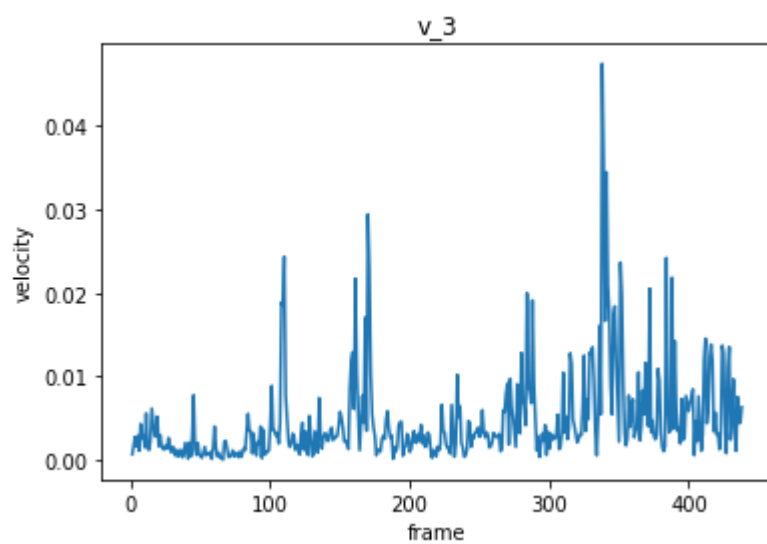


## 7. Wyniki

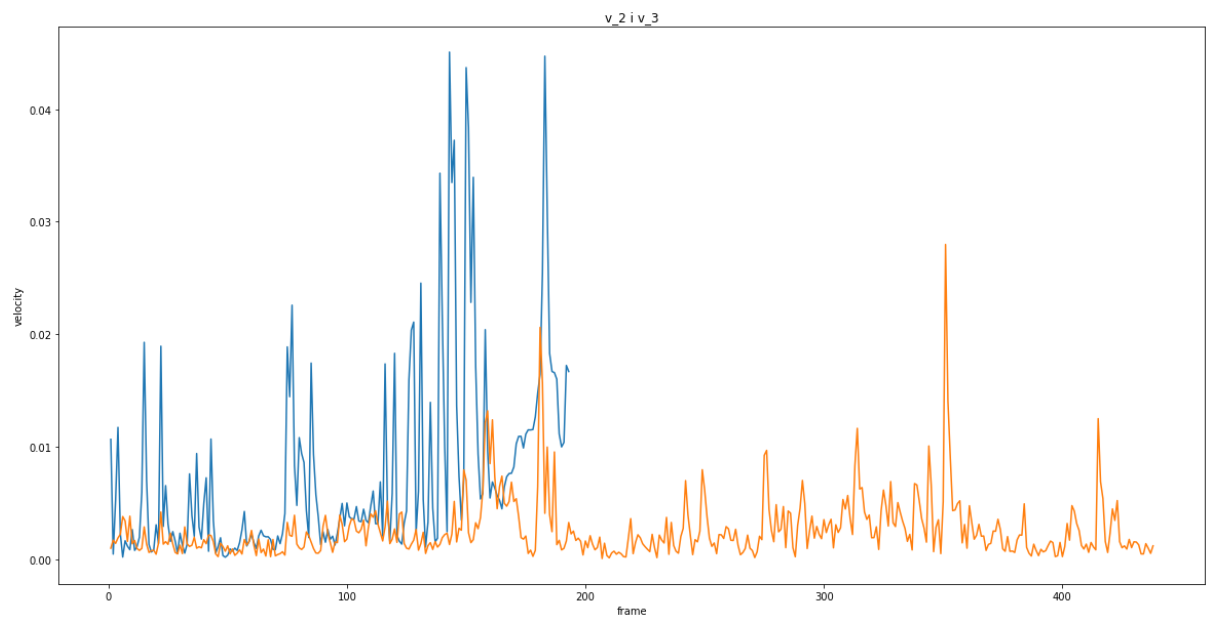
Wynikiem działania modelu na danych były bounding boxy opisujące wykrytego człowieka. Stworzono algorytm, który analizował centra tych bounding boxów i decydował czy stoją one w jednej linii i czy przesuwają się w tym samym kierunku. Następnie algorytm ten policzył prędkość poruszania się ludzi poprzez liczenie ile centro bounding boxu poruszyło się na kolejnej klatce względem poprzedniej. Te dane zostały następnie przetworzone i za ich pomocą narysowano wykresy przesuwania się przykładowych dwóch osób na nagraniu. Tam gdzie na wykresie jest pik, tam dana osoba poruszyła się, tam gdzie wykres jest płaski, stała w miejscu. Wykresy takie następnie wygładzono, aby pozbyć się szumu i zastosowano splot na obu wykresach.



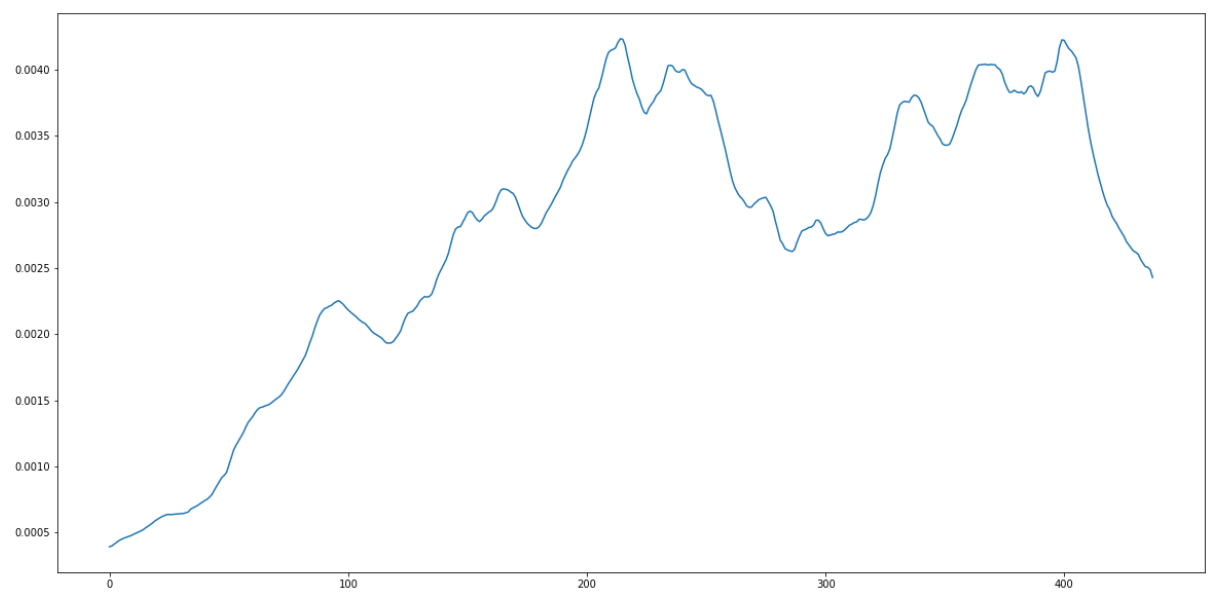
Wykres prędkości osoby 1



Wykres prędkości osoby 2



Oba wykresy na raz



Splot obu wykresów.

Na wykresie przedstawiającym splot widać, duże wartości pokazują miejsca gdzie oba wykresy podobnie się zachowywały. Widać na nim dwa wyraźne góry, które oznaczają, że obie osoby poruszyły się do przodu w podobnym czasie. Wykres ten opisuje więc charakterystyczną monotoniczność kolejki, w której osoby stają, następnie powoli się poruszają po czym znowu synchronicznie stoją. Wykres ten może służyć do analizowania zachowania się ludzi w kolejce. Gdyby ktoś wpychał się do kolejki lub próbował kogoś w niej wyprzedzić, byłoby to widoczne na takim wykresie. Dodatkowo z danych można wyciągnąć inne wnioski, np licznosc kolejki, odpowiada ona liczbie bounding boxów w danym czasie. Można również wyliczyć średni czas stania w kolejce poprzez wyliczenia średniego czasu bycia widocznym w kolejce. np człowiek o id 1 stał w kolejce jedynie 200 klatek, ale człowiek o id 2 już 400 klatek. Jeśli weźmie się pod uwagę wszystkie osoby które czekają w kolejce można obliczyć średni czas oczekiwania w niej. Niestety przykładowe dane są zbyt krótkie, aby zaprezentować na nich potencjał algorytmu. Na szczęście API sieci YOLO jest bardzo przyjazne, a algorytmy przez nas opisane są z nim kompatybilne dzięki czemu można tego użyć na innych dłuższych danych i na nich wyliczać konkretne metryki.