

Wykrywanie Strzałek za pomocą Modelu YOLO

Przegląd Projektu

Projekt skupia się na opracowaniu i wdrożeniu niestandardowego modelu do wykrywania obiektów opartego na YOLO (You Only Look Once), który został specjalnie dostrojony do wykrywania strzałek kierunkowych (up, down, left, right) na obrazach i strumieniach wideo. Model został wytrenowany na niestandardowym zestawie danych strzałek i podobnych symboli i jest zdolny do przetwarzania zarówno statycznych obrazów, jak i strumieni wideo w czasie rzeczywistym. Dokumentacja ta zawiera kompleksowy przewodnik po modelu, procesie treningowym, metrykach oceny i sposobie użytkowania.

Opis Modelu

Architektura YOLO

Architektura YOLO (You Only Look Once) jest zaawansowanym systemem do wykrywania obiektów w czasie rzeczywistym. W przeciwieństwie do tradycyjnych metod, które analizują obraz w wielu lokalizacjach i skalach, YOLO traktuje wykrywanie obiektów jako pojedynczy problem regresji. Model jednocześnie przewiduje ramki ograniczające oraz prawdopodobieństwa klasy dla wszystkich obiektów na obrazie w jednej ewaluacji. Dzięki temu YOLO jest szybkie i efektywne, umożliwiając wykrywanie obiektów w czasie rzeczywistym z wysoką precyzją.

Model do Wykrywania Strzałek

W tym projekcie zastosowano zmodyfikowaną wersję modelu YOLO, dostrojoną do wykrywania czterech klas strzałek kierunkowych:

- Up Arrow
- Down Arrow
- Left Arrow
- Right Arrow

Te klasy zostały zdefiniowane w zestawie danych stworzonym specjalnie dla tego zadania. Model został wytrenowany, aby rozpoznawać te strzałki w różnych orientacjach i kontekstach, zapewniając solidne działanie w różnych środowiskach.

Modele Wstępnie Wytrenowane

- **YOLOv9t.pt:** Był to początkowo zamierzony model, znany ze swojej wysokiej wydajności dzięki dużej liczbie warstw i parametrów. Jednak z powodu ograniczeń zasobów obliczeniowych, ten model był trudny do skutecznego wytrenowania.
- **YOLOc9e.pt:** Z powodu ograniczeń sprzętowych, użyto mniejszego modelu, YOLOc9e.pt, do końcowego treningu i wdrożenia. Pomimo mniejszych rozmiarów, model zapewnił zadowalające wyniki, chociaż wydajność mogłaby zostać poprawiona za pomocą potężniejszego sprzętu i modelu YOLOv9t.pt.

Zbiór Danych

Struktura Zbioru Danych

Zbiór danych składa się z obrazów z oznaczonymi ramkami ograniczającymi wokół strzałek kierunkowych. Obrazy te są podzielone na zestawy treningowe, walidacyjne i testowe, aby zapewnić, że model będzie dobrze generalizować do nowych danych.

- **Zbiór Treningowy:** Używany do trenowania modelu, zawiera różnorodny zestaw obrazów strzałek z różnymi orientacjami i tłami.
- **Zbiór Walidacyjny:** Używany w trakcie treningu do dostrajania hiperparametrów i zapobiegania nadmiernemu dopasowaniu.
- **Zbiór Testowy:** Używany po treningu do oceny wydajności modelu.

Zbiór Danych: <https://universe.roboflow.com/mdpindiv-m1xmm/datasetv1-ctr75>

Trening Modelu

Proces Treningowy

Proces treningu obejmował następujące kroki:

1. **Inicjalizacja Modelu:** Model YOLO został zainicjowany z wstępnie wytrenowanymi wagami z dużego zadania wykrywania obiektów (YOLOv9t.pt), co stanowiło solidny punkt wyjścia do dostrojenia na zestawie danych do wykrywania strzałek.
2. **Trening:** Model trenowano przez 10 epok, z każdą epoką składającą się z wielu iteracji nad zestawem treningowym.
3. **Zapis Punktów Kontrolnych:** Najlepiej działające wagi modelu były zapisywane po każdej epoce, co pozwalało na wybór optymalnego modelu na koniec treningu.

Metryki Treningowe

Wydajność modelu była monitorowana za pomocą kilku metryk w trakcie treningu:

- **Loss Function:** Główne metryki to `box_loss`, `cls_loss` i `dfl_loss`, które mierzą dokładność ramek ograniczających, klasyfikacji i strat ogniskowych dystrybucyjnych.
- **Precision and Recall:** Precyzja i czułość były monitorowane, aby zrozumieć równowagę między fałszywymi trafieniami a fałszywymi negatywami.
- **mAP@0.5 i mAP@0.5:0.95:** Te metryki były używane do oceny średniej precyzji przy progach IoU 0.5 oraz w zakresie od 0.5 do 0.95.

Logi Treningowe

Podczas 10 epok treningowych zaobserwowano następujące kluczowe metryki:

- **Precision:** Wzrosła z 0.352 do 0.987
- **Recall:** Wzrosła z 0.418 do 0.993
- **mAP@0.5:** Znacząco się poprawiła, osiągając wartość 0.994 w ostatniej epoce
- **mAP@0.5:0.95:** Osiągnęła wartość 0.815, wskazując na wysoki poziom dokładności przy różnych progach IoU

Szczegółowe logi treningowe wykazały, że model znacząco poprawił się w trakcie treningu, pomimo wyzwań związanych z ograniczeniami sprzętowymi.

Ocena Modelu

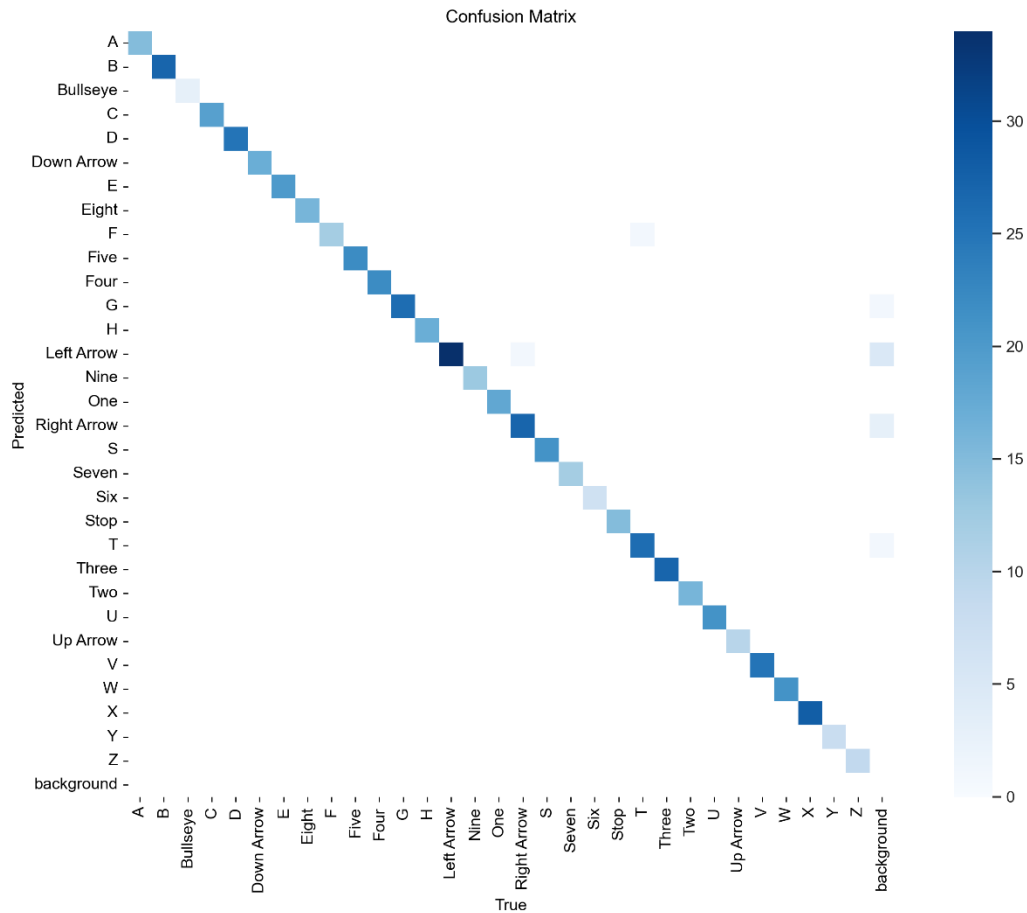
Model został oceniony na zestawie walidacyjnym, osiągając następujące wyniki:

- **Precision:** 0.987
- **Recall:** 0.993
- **mAP@0.5:** 0.994
- **mAP@0.5:0.95:** 0.815

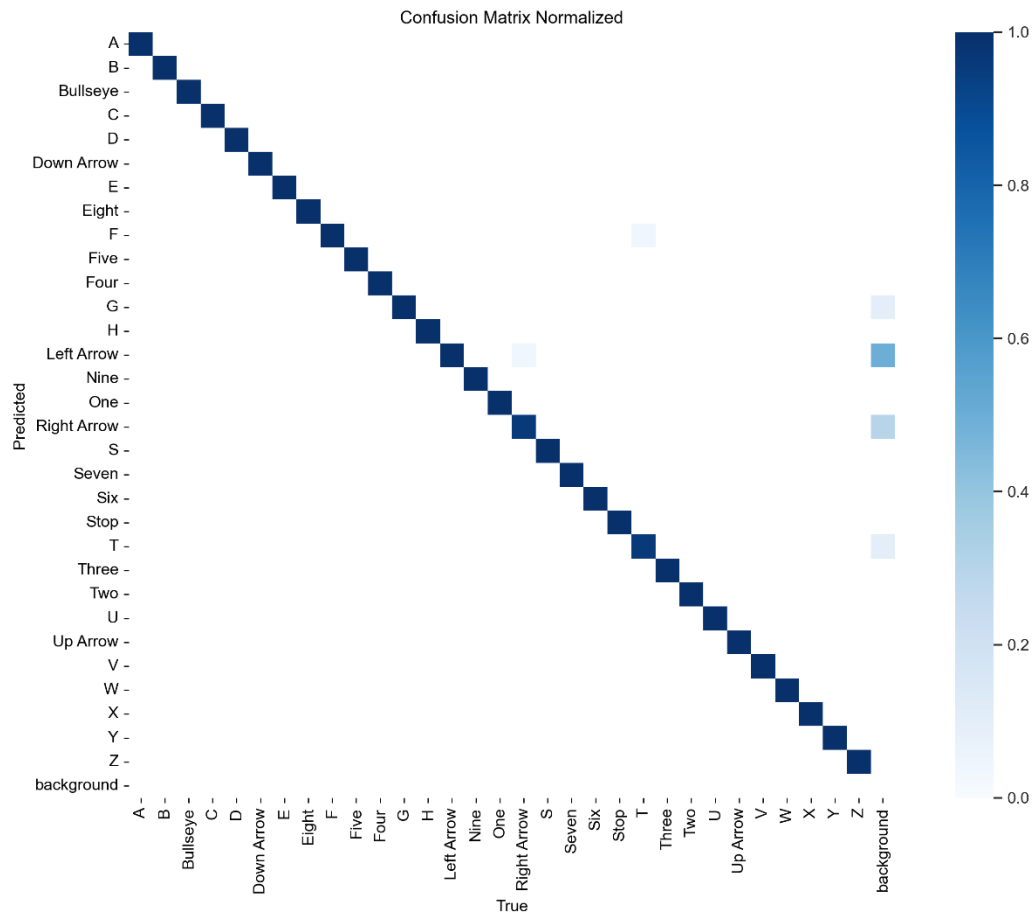
Te metryki wskazują, że model jest bardzo dokładny w wykrywaniu strzałek, osiągając doskonałe wartości precyzji i czułości. Metryki mAP pokazują, że model działa dobrze przy szerokim zakresie progów Intersection-over-Union (IoU), co oznacza, że potrafi skutecznie wykrywać obiekty nawet w trudniejszych przypadkach, zachowując wysoką dokładność.

Confusion Matrix

Macierz została wygenerowana, aby zobrazować wydajność modelu w różnych klasach. Znormalizowana wersja tej macierzy dodatkowo podkreśliła zdolność modelu do dokładnego klasyfikowania różnych typów strzałek.



- **Normalized Confusion Matrix:** The normalized version of the confusion matrix further highlights the model's ability to accurately classify different arrow types.



Testowanie Modelu

Testy na Obrazach

Model został przetestowany zarówno na pojedynczych obrazach, jak i na partii obrazów, aby ocenić jego wydajność w praktycznych scenariuszach:

- **Testy na Pojedynczym Obrazie:** Model był w stanie dokładnie wykrywać strzałki nawet na obrazach z złożonymi tłami.
- **Testy na Wielu Obrazach:** Model wykazywał spójną wydajność w całej partii testowych obrazów, z wysoką dokładnością w wykrywaniu prawidłowych kierunków strzałek.

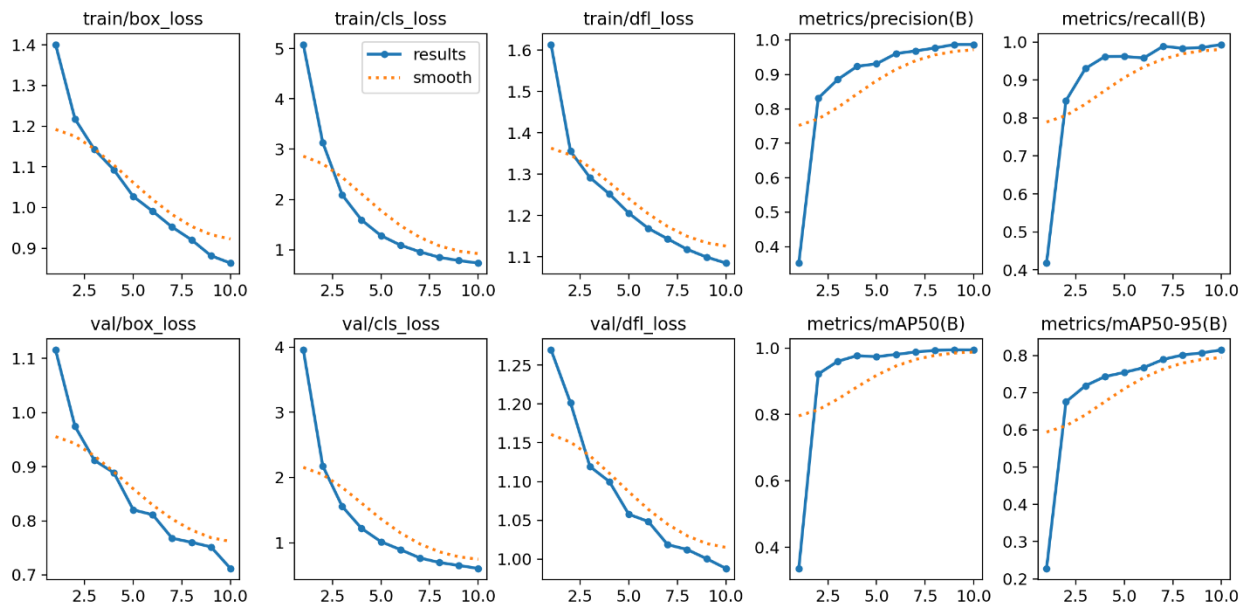
Wykrywanie w Czasie Rzeczywistym

Model został również przetestowany na strumieniach wideo w czasie rzeczywistym, aby wykrywać strzałki pojawiające się na żywo w kamerze. Pomimo ograniczeń sprzętowych, model działał wystarczająco dobrze, chociaż zauważono lekkie opóźnienia w przetwarzaniu spowodowane użyciem CPU.

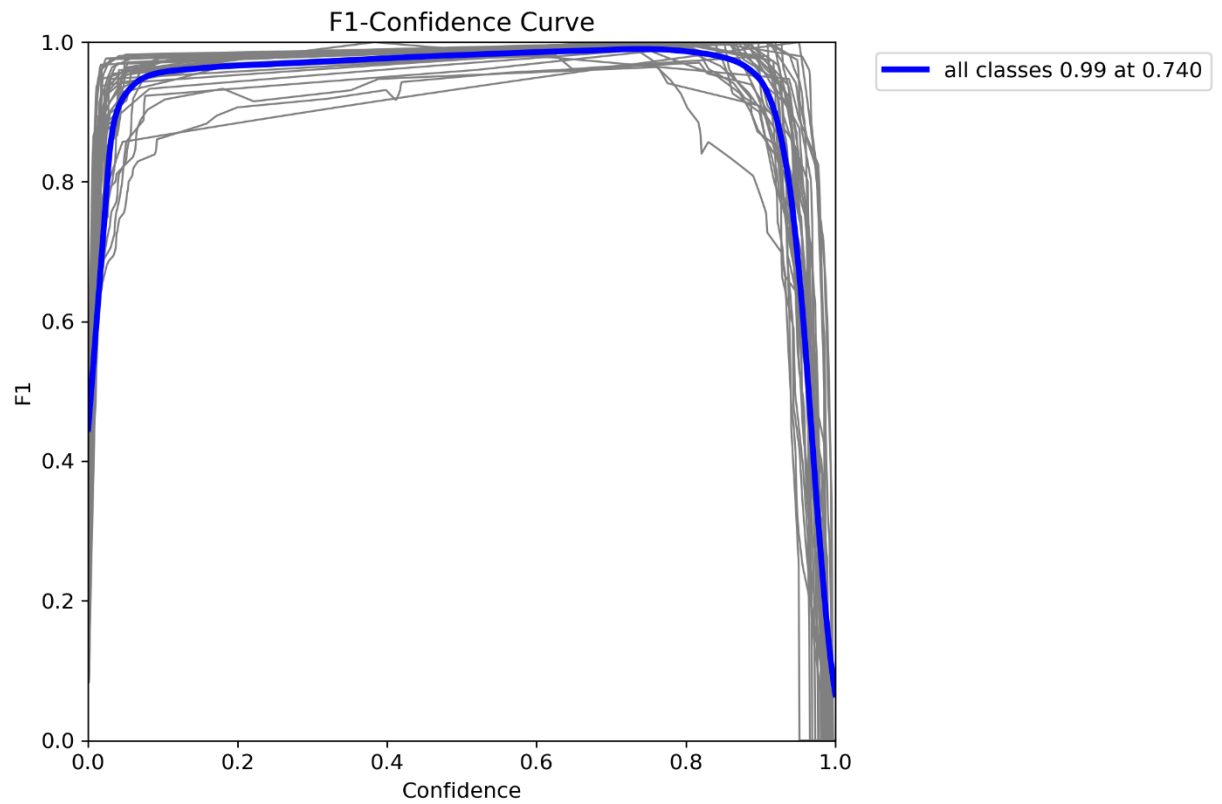
Wizualizacje

Stworzono kilka wizualizacji, aby lepiej zrozumieć wydajność modelu:

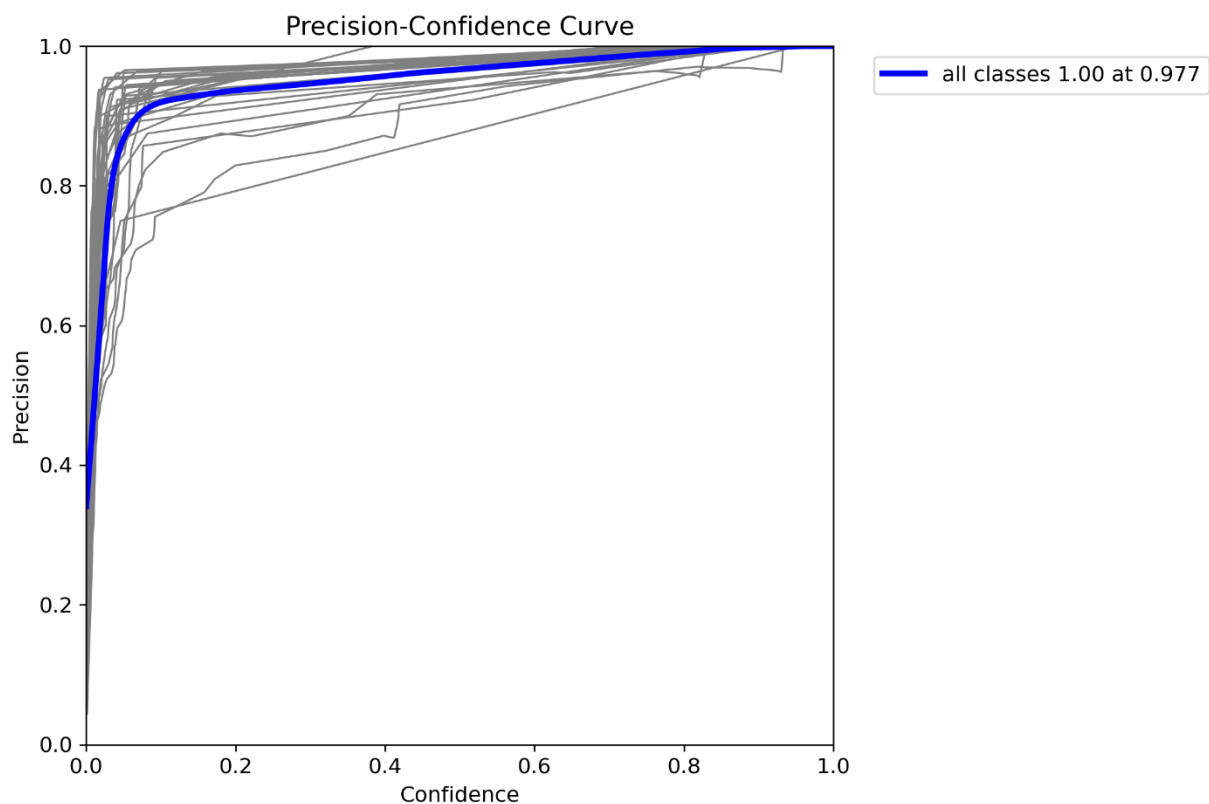
1. **Training Loss Curves:** Te krzywe pokazują zmniejszanie się wartości funkcji straty w trakcie epok, co wskazuje na skuteczne trenowanie modelu.



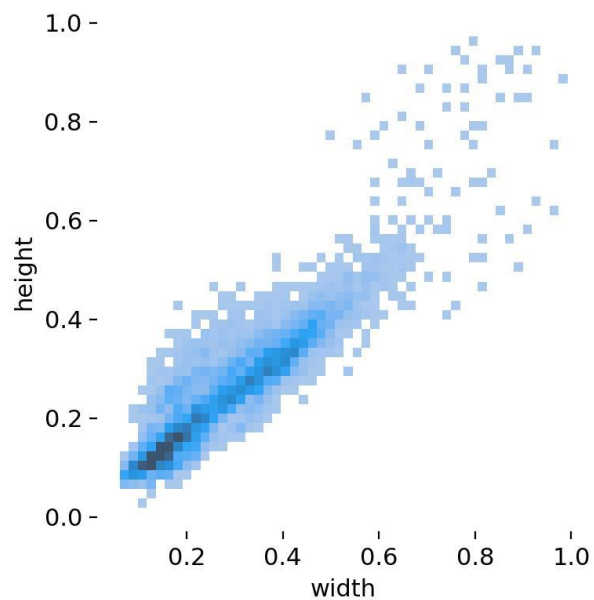
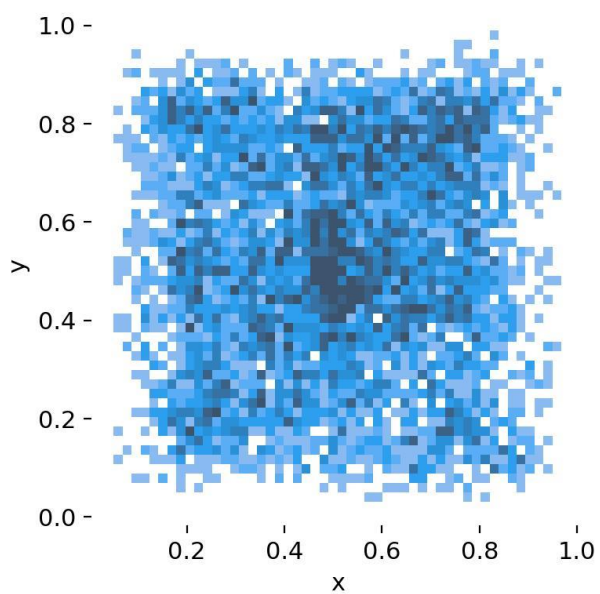
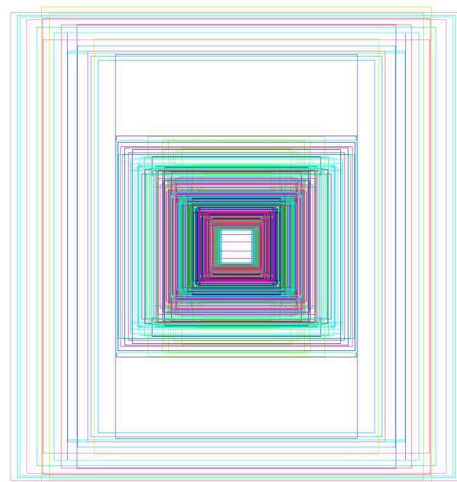
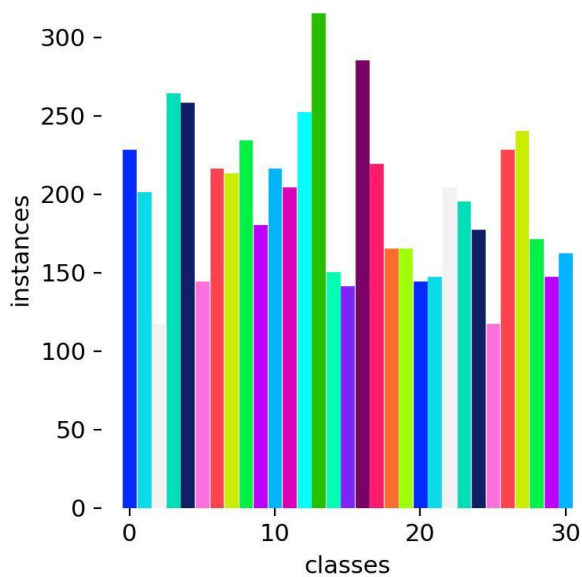
2. **F1-Confidence Curve:** Krzywa ta przedstawia zależność między progiem pewności a wynikiem F1, ukazując optymalną równowagę między precyzją a czułością.



3. **Precision-Confidence Curve:** Wizualizuje precyzję modelu przy różnych poziomach pewności, potwierdzając, że model utrzymuje wysoką precyzję nawet przy niższych progach pewności.



4. **Label Distribution and Correlogram:** Te wykresy dostarczają informacji na temat rozkładu etykiet i korelacji między różnymi atrybutami ramek ograniczających (np. szerokość i wysokość).



Wnioski

Projekt z powodzeniem demonstrowa wykorzystanie modelu YOLO do wykrywania strzałek kierunkowych na obrazach i strumieniach wideo. Pomimo ograniczeń wynikających z niewystarczających zasobów obliczeniowych, model osiągnął wysoką dokładność i dobrze radził sobie zarówno w statycznych, jak i dynamicznych scenariuszach. Użycie mniejszego modelu, YOLOc9e.pt, umożliwiło praktyczne wdrożenie, zapewniając jednocześnie solidne wyniki.

Możliwe Ulepszenia

Przyszłe usprawnienia mogą obejmować:

- **Użycie mocniejszego sprzętu:** Trening większego modelu YOLOv9t.pt na bardziej wydajnym sprzęcie może przynieść jeszcze lepsze rezultaty.
- **Augmentacja danych:** Zastosowanie bardziej zaawansowanych technik augmentacji danych mogłoby pomóc modelowi lepiej generalizować w różnych środowiskach.
- **Optymalizacja czasu rzeczywistego:** Dalsza optymalizacja wydajności w czasie rzeczywistym, zwłaszcza na urządzeniach krawędziowych, mogłaby uczynić model bardziej praktycznym dla zastosowań, takich jak pojazdy autonomiczne czy robotyka.