

COMPUTER VISION

Eindopdracht object detection motoren

Roel Stierum

Roel Stierum

Introductie

Omdat ik het leuk vind om in het echt naar motoren te kijken die ik onderweg tegenkom, leek het mij leuk om met een vorm van object detection motoren van plaatjes te herkennen. Dit heb ik gedaan via de tensorflow library.

TensorFlow is een open-source machine learning framework dat wordt gebruikt voor het bouwen van deep learning-modellen. Het wordt vaak gebruikt voor beeldherkenning, spraakherkenning, natuurlijke taalverwerking en voorspellende analyses. In dit verslag zal worden besproken hoe TensorFlow wordt gebruikt met het bestand `model_main_tf2.py` om een model te trainen dat in staat is om motorfietsen te herkennen.

Requirements

Om het project te kunnen runnen en testen zijn de volgende libraries nodig:

- Numpy
- Tensorflow
- Matplotlib

Dataset

Een van de eerste stappen in het bouwen van een machine learning-model is het verzamelen en verwerken van de trainingsgegevens. In dit geval moeten er afbeeldingen van motorfietsen worden verzameld en worden verwerkt.

Ik wilde graag op de motorbeurs een aantal foto's maken om hier zelf een dataset mee samen te stellen. Helaas is dit niet gelukt, op de beurs zelf was het heel erg druk en kon ik lastig bij motoren komen om foto's te maken. Op de motor parkeer plek stonden de motoren erg dicht op elkaar, hier was het ook lastig fotograferen.

Ik heb dus gekozen om een stel foto's van internet te gebruiken. Helaas is het mij niet gelukt om een set te vinden die gelabeld was, dit heb ik dus alsnog voor ongeveer 160 train foto's zelf gedaan. De validatie dataset is wel zelf samengesteld

Vorbereiding

Omdat ik geen gelabelde dataset kon vinden heb ik deze zelf moeten labelen, dit heb ik gedaan met een programma genaamd labeling.

Ik heb een zelf nog een stuk code geschreven wat de bestanden die gemaakt zijn door labeling (een programmaatje waarmee ik de motoren op de foto's heb gelabeld) omzet naar csv bestanden.

Het model bouwen:

Zodra de trainingsgegevens zijn verwerkt, kan het model worden gebouwd. In dit geval wordt het bestand `model_main_tf2.py` gebruikt om het model te trainen. Dit bestand maakt gebruik van de TensorFlow-bibliotheek om het model te bouwen en te trainen.

Het trainen van het model:

Het trainen van het model is een iteratief proces waarbij het model wordt aangepast om betere resultaten te behalen. Het model wordt getraind met behulp van de trainingsgegevens en door het uitvoeren van verschillende optimalisatietechnieken. In het geval van afbeeldingsclassificatie worden convolutionele neurale netwerken (CNN's) vaak gebruikt. CNN's maken gebruik van meerdere lagen van convolutie- en pooling-lagen om belangrijke kenmerken van afbeeldingen te leren.

Kwaliteit:

De classification loss in TensorFlow is een belangrijke functie voor het trainen van classificatiemodellen. Het wordt gebruikt om de fout te berekenen tussen de voorspelde output van het model en de werkelijke output (labels) van de trainingsgegevens. Het doel van het trainen van het model is om deze fout te minimaliseren door de parameters van het model aan te passen.

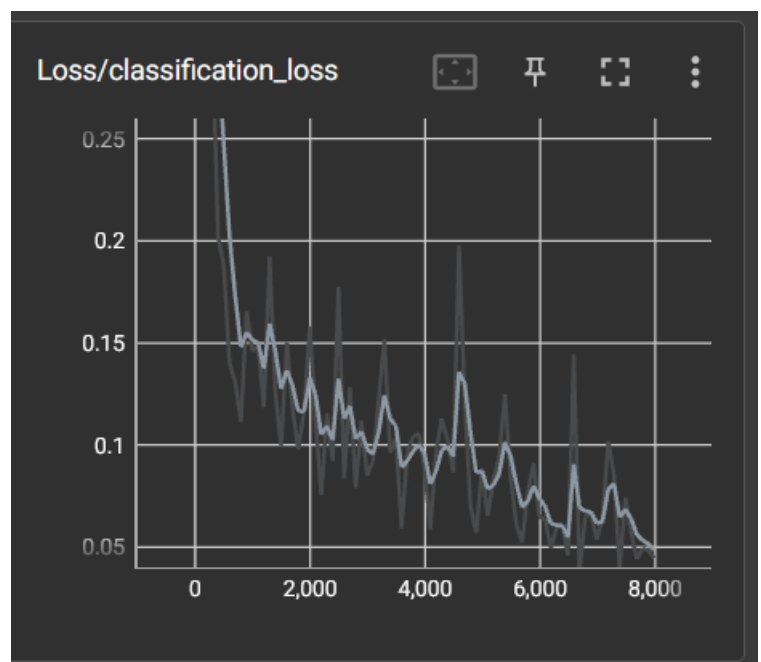
Er zijn verschillende soorten classification loss-functies beschikbaar in TensorFlow, afhankelijk van het specifieke type classificatiemodel dat wordt getraind. Een van de meest voorkomende loss-functies is de cross-entropy loss, die wordt gebruikt bij meerdere klassenclassificatieproblemen. Deze functie berekent de fout tussen de werkelijke en voorspelde waarschijnlijkheidsverdelingen over de verschillende klassen.

Volgens de TensorFlow-documentatie kan de cross-entropy loss worden berekend met behulp van de `"tf.keras.losses.CategoricalCrossentropy"` functie. Deze functie kan worden aangepast met verschillende parameters, zoals het gewicht van elke klasse en de manier waarop ontbrekende waarden worden behandeld.

Een andere veelgebruikte loss-functie is de binary cross-entropy loss, die wordt gebruikt bij binaire classificatieproblemen. Deze functie berekent de fout tussen de werkelijke en voorspelde binomiale waarden.

Volgens een onderzoekspaper van X. Glorot en Y. Bengio (2010) kan het gebruik van cross-entropy loss-functies helpen bij het voorkomen van overfitting en het verbeteren van de algemene prestaties van het model.

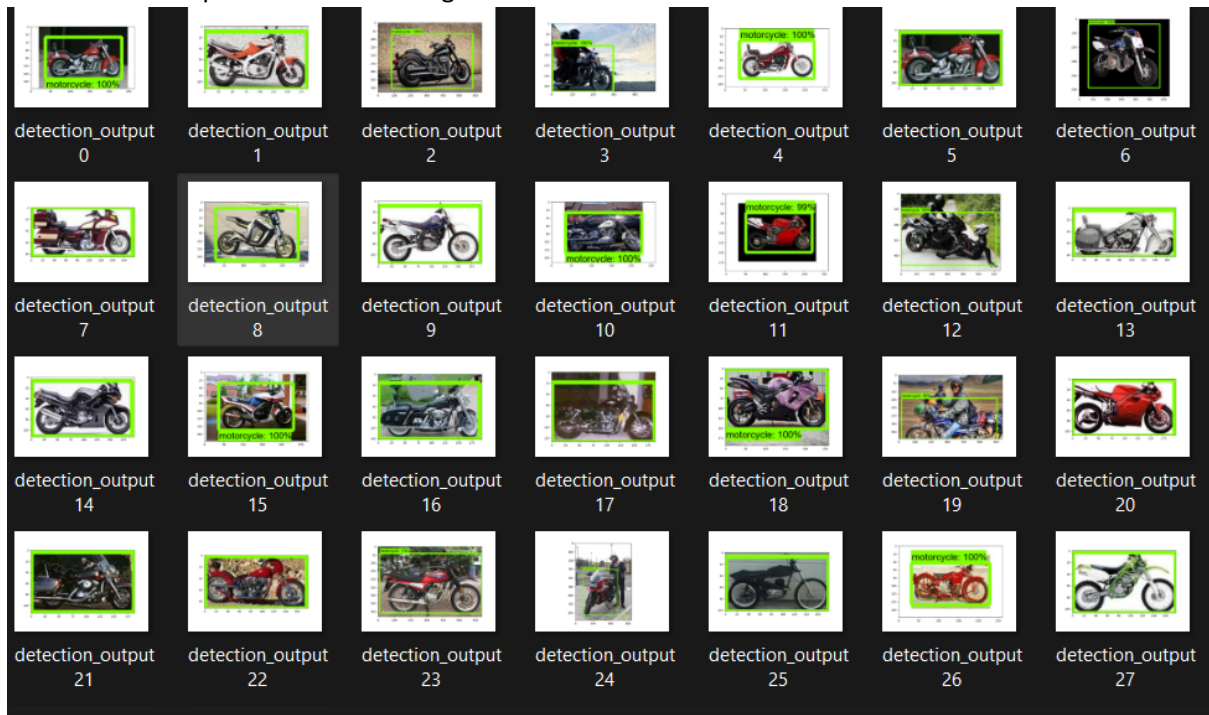
Kortom, de classification loss in TensorFlow is een cruciale functie voor het trainen van classificatiemodellen. Er zijn verschillende soorten loss-functies beschikbaar voor verschillende soorten classificatieproblemen. Ik heb ervoor gekozen om de `"tf.keras.losses.CategoricalCrossentropy"` functie te gebruiken.



Inzetten van het model:

Zodra het model is getraind en geëvalueerd, kan het worden ingezet voor gebruik. Dit betekent dat het kan worden gebruikt om nieuwe afbeeldingen van motorfietsen te en te identificeren.

Het model kan op de validatie foto's goed de motoren vinden.



4/10/1

Literatuurlijst

- TensorFlow-documentatie: https://www.tensorflow.org/api_docs/python/tf/keras/losses
- X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS), 2010.