

Videoanalyse und Objekttracking

1st Bartolovic Eduard
Hochschule München
München, Deutschland
eduard.bartolovic0@hm.edu

2nd Thomas Willeit
Hochschule München
München, Deutschland
XXXXX@hm.edu

3rd Schäfer Julia
Hochschule München
München, Deutschland
j.schaefer0@hm.edu

Zusammenfassung—

I. KONZEPT

Ziel des Projektes ist es klassische Verfahren mit neuere DeepLearning Verfahren zu vergleichen. Dafür ist für beide arten eine Pipeline geschaffen worden. Für die Klassische Verfahren wurde für die Objekterkennung Gausmixture verwendet und für das Tracking Sort. Für die DeepLearnig Verfahren wurde für die Objekterkennung ein Neuronales Netz und für das Tracking wurde Deepsort verwendet.

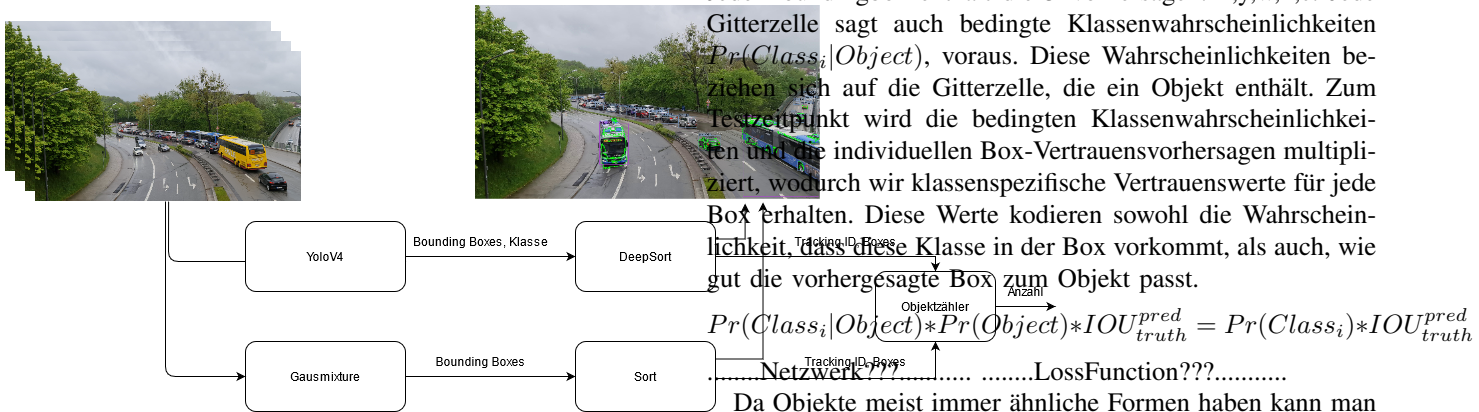


Abbildung 1. Konzept für das Projekt

II. OBJEKTERKENNUNG: GAUSS MIXTURE

III. OBJEKTERKENNUNG: YOLOV4

YOLO ist ein Neuronal Netz für die Echtzeit-Objekterkennung. YOLO ist die Abkürzung für 'You Only Look Once' was übersetzt 'Man sieht nur einmal hin' heißt. Die Aufgabe der Objekterkennung besteht darin, den Ort und die Boundingbox im Bild zu bestimmen, sowie die Objekte zu klassifizieren. Frühere Methoden, wie R-CNN und seine Variationen, verwendeten eine Pipeline, um diese Aufgabe in mehreren Schritten durchzuführen. Dies ist in der Ausführung langsam und aufwendiger zu trainieren, da jede einzelne Komponente separat trainiert werden muss. YOLO, erledigt beide Aufgaben mit einem einzigen neuronalen Netzwerk. Es betrachtet dabei die Objekterkennung als ein Regressionsproblem auf räumlich getrennte Bounding Boxes und zugehörige Klassenwahrscheinlichkeiten. [1]

Das Bild in ein $S \times S$ -Gitter mit den Residualblöcken aufgeteilt. Wenn der Mittelpunkt eines Objekts in eine Gitterzelle fällt, ist diese Gitterzelle für die Erkennung dieses Objekts zuständig. Jede Gitterzelle sagt B Bounding Boxes und C Klassen Konfidenzwerte für diese Boxen voraus [1]. Diese Klassen Konfidenzwerte zeigen, wie sicher das Modell ist, dass die Boundingbox ein Objekt enthält und auch wie genau die Box das Objekt beschreibt. Die Konfidenz wird wie folgt definiert:

$$c = Pr(Object) * IOU_{pred}^{truth}$$

Jede Boundingbox enthält die 5 Vorhersagen: x,y,w,h,c. Jede Gitterzelle sagt auch bedingte Klassenwahrscheinlichkeiten $Pr(Class_i|Object)$, voraus. Diese Wahrscheinlichkeiten beziehen sich auf die Gitterzelle, die ein Objekt enthält. Zum Testzeitpunkt wird die bedingten Klassenwahrscheinlichkeiten und die individuellen Box-Vertrauensvorhersagen multipliziert, wodurch wir klassenspezifische Vertrauenswerte für jede Box erhalten. Diese Werte kodieren sowohl die Wahrscheinlichkeit, dass diese Klasse in der Box vorkommt, als auch, wie gut die vorhergesagte Box zum Objekt passt.

$$Pr(Class_i|Object) * Pr(Object) * IOU_{truth}^{pred} = Pr(Class_i) * IOU_{truth}^{pred}$$

Da Objekte meist immer ähnliche Formen haben kann man eine gewisse Menge an sogenannten Ankerboxen definieren welche als Basis Boundingboxen fungieren. Diese Ankerboxen kann man mittels Clusteralgorithmen wie K-Means und den Boundingboxen aus dem Datensatz berechnen....

In diesem Projekt verwendeten wir die 4te und damit aktuell neueste Version von Yolo [2]. So bietet jede Version inkrementelle Verbesserung zum jeweiligen Vorgänger.

Unser Modell wurde bereits mit dem Microsoft COCO Datensatz trainiert. Es sind 80 verschiedene Klassen erkennbar. Wir interessieren uns aber nur für einen kleineren Teil wie:

- 1) Personen
- 2) Pkw
- 3) Lkw
- 4) Busse
- 5) Fahrräder
- 6) Züge

Je nach Szenario lassen sich per Parameter die relevanten Klassen auswählen. Das Modell prädiziert trotzdem noch alle Klassen. Die nicht relevanten werden einfach im Postprocessing herausgefiltert. Das Ergebnis einer Prädiktion ist eine

Liste von Objekten welches aus den x,y Koordinate des Mittelpunktes der Boundingbox, der Breite w, der Höhe h, der Klasse und des Konfidenzwerts. Ein weitere Postprocesingschritt ist die Non-Maxima-Suppression. BEISPIELBILD? Der Output kann überschneidende Boundingboxen besitzen die eigentlich zur selben Klasse gehören. Diese Duplikate können mittels der Non-Maxima-Suppression entfernt werden. Heirfür wird die IOU der Boxen berechnet. Sollte die IOU einen Threshold überschreiten dann wird das Duplikat entfernt.

A. Fehler in der Erkennung

...falsch Klassifizierung wegen schlechter Datensatz.... Zu großer Datensatz...

YOLO besitzt starke räumliche Beschränkungen für die Boundingbox-Vorhersagen, da jede Gitterzelle nur zwei Boxen vorhersagen kann und nur eine Klasse haben kann. Diese räumliche Beschränkung begrenzt die Anzahl der Objekte in der Nähe, die das Modell vorhersagen kann. ———— Problem zu viele Elemente nebeneinander..... wegen zu großen matschigen Grid zu wenig Ankerboxen????—————

IV. TRACKING: SORT

V. TRACKING: DEEPSORT

VI. ZÄHLEN VON OBJEKTEN

VII. ZUSTANDSERKENNUNG ZÜGE

LITERATUR

[1] <https://arxiv.org/pdf/1506.02640v5.pdf>

[2] <https://arxiv.org/pdf/2004.10934.pdf>

VIII. ANHANG