

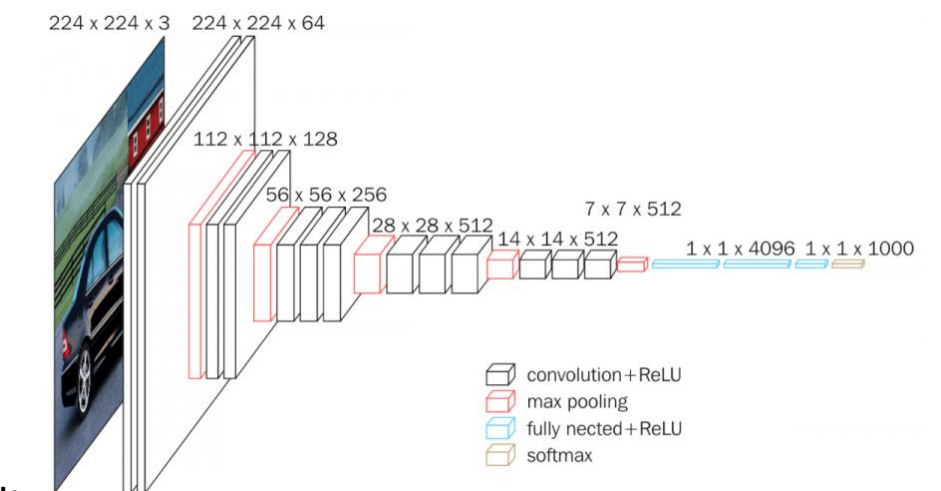
Objekterkennung für eine automatisierte Qualitätskontrolle mittels Deep Learning Algorithmen

MOTIVATION

- Von der Firma Simon iBV wurde ein Programm entwickelt, das 3 Klemmentypen finden und ihre Lage und Orientierung im Bild bestimmen soll.
- Dies erfolgt, gemäß konventioneller Bildverarbeitung, durch Thresholding nach Helligkeitswerten, Kantenerkennung sowie Position bezüglich bestimmter Fixpunkte.
- Dieser Ansatz zieht einige Nachteile nach sich:
 - Intoleranz gegenüber unterschiedlichen Oberflächenbeschaffenheiten der Objekte.
 - Intoleranz gegenüber wechselnden Umgebungsbedingungen (z.B. Beleuchtung, Entfernung).
 - Für jeden Klemmentyp muss ein separates Unterprogramm geschrieben werden.

IDEE

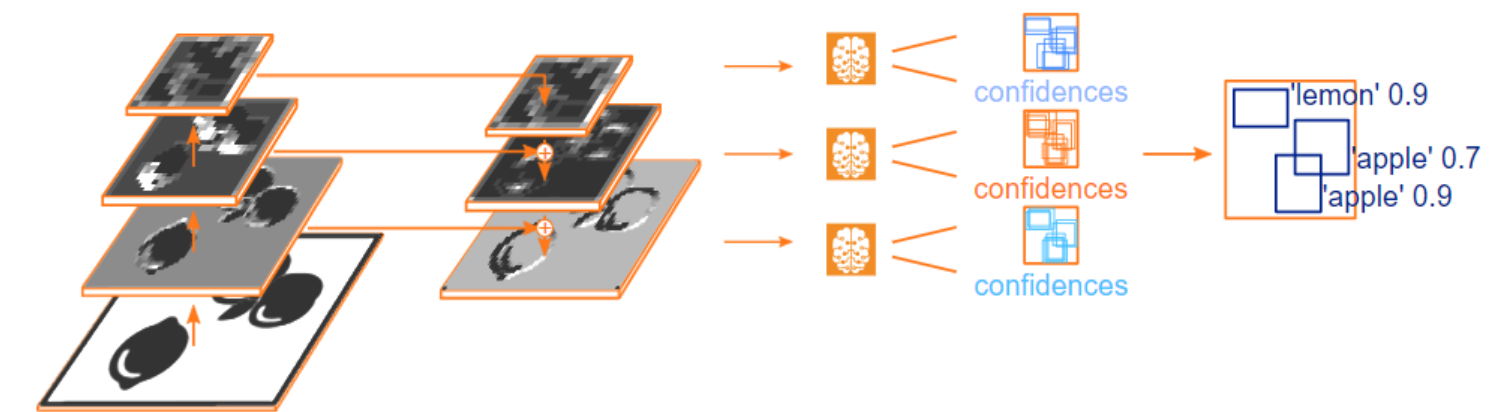
- Neuronale Netzwerke können anhand von Trainingsdaten Vorhersagen treffen.
- Convolutional Neural Networks (CNN's) sind eine Untermenge der neuronalen Netzwerke und sind auf Erkennung von Objekten auf Bildern spezialisiert.
- Die Idee ist also, einen Satz von Trainingsbildern zu erstellen, CNN's damit zu trainieren und deren Performance auszuwerten, um herauszufinden, ob dieser Ansatz eine mögliche Alternative darstellt.



Quelle: <https://neurohive.io/wp-content/uploads/2018/11/vgg16-1-e1542731207177.png>

IMPLEMENTIERUNG

- Erstellung von repräsentativen Trainings- und Testdatensätzen die zwei Anwendungsfälle abbilden:
 1. Aus einer Aufnahme pro vermutetes Objekt ein 300x300 Pixel Fenster ausschneiden und die Anwesenheit eines Objekts darin zu überprüfen (kleine ROI).
 2. Pro Aufnahme eine große ROI ausschneiden, die alle möglichen Objektpositionen abdeckt.
- Erstellung, Parametrierung und Training von neuronalen Netzwerken mit der Halcon Deep Learning Bibliothek bzw. dem Detectron2 Framework.
- Auswertung der Performance der Netzwerke durch einen Vergleich der Ground Truth Daten mit den Netzwerkvorhersagen und Messung der Ausführungszeit.



Quelle: https://www.mvtec.com/doc/halcon/1911/en/toc_deeplearning_objectdetection.html

ERGEBNISSE UND BEWERTUNG

- Es ist grundsätzlich möglich, Deep Learning Netzwerke zur Lösung dieses Problems einzusetzen.
- Allerdings ist dabei mit einem höheren Rechenaufwand und daher höheren Hardwareanforderungen zu rechnen.
- Der Programmieraufwand ist beim dem Deep Learning Ansatz niedriger als bei einer konventionellen Lösung und bleibt unverändert, wenn neue Objekte oder Situationen berücksichtigt werden müssen, da dazu keine weitere Programmierung nötig ist, sondern es lediglich durch Erweiterung des Datasets und Neujustierung des bereits bestehenden Modells erfolgen kann.
- Alle Netzwerke zeigen hohen Average-Precision-Wert, bei der Klasse BM4 ist dieser jedoch stets am niedrigsten. Dies könnte durch Ergänzung des Datasets und weiteres Training behoben werden.
- Lageerkennung ist zufriedenstellend, jedoch sollte die Präzision des Detectron2 Netzwerks bei der Winkelerkennung weiter verbessert werden.
- Auch die Ausführungszeit des Detectron2 Netzwerks ist zu langsam und bedarf ebenfalls weiterer Verbesserung.