

YOLO-Bilderkennung in der Automatisierung

Ein Machine Vision-Retrofit mit YOLO-KI-Funktionen direkt an der Kamera liefert die visuelle Intelligenz für neue Anwendungen in der Fabrik- und Gebäudeautomatisierung

YOLO FÜR AUTOMATISIERER

Wie moderne Echtzeit-Bilderkennung Bestandsanlagen intelligenter macht

-  Objekte erkennen in Echtzeit
-  Prozesse optimieren und Kosten senken
-  Einfach nachrüsten statt ersetzen
-  Sicher, lokal und DSGVO-konform



YOLO (You Only Look Once) ist ein beliebtes und weit verbreitetes KI-Modell zur Objekterkennung und Bildsegmentierung in Videodaten. Es wurde von Forschern der University of Washington in Seattle entwickelt und 2015 veröffentlicht. YOLO basiert auf einem hochoptimierten, einstufigen CNN (Convolutional Neural Network), das sich durch einen „One-Shot“-Ansatz besonders zur Echtzeitobjekterkennung, wie beispielsweise autonomes Fahren oder Videoüberwachung, eignet.

YOLO bearbeitet ein Bild in einem Durchlauf, um Bounding Boxes (Position und Größe aller im Bild erkannten Objekte) und deren Klassenhäufigkeiten direkt vorherzusagen. Theoretisch könnte YOLO bis zu 9000 unterschiedliche Objektklassen unterscheiden. Typischerweise sind YOLO-Modelle mit Hilfe des COCO (Common Objects in Context)-Datensatzes trainiert und können 80 verschiedene Kategorien (wie Personen, Autos, Tiere) erkennen. Durch die mittlerweile über 10 Jahre laufende YOLO-Entwicklung existieren inzwischen verschiedene Versionen [3]. Sehr verbreitet ist aktuell YOLOv8. Beim kommerziellen Einsatz ist zu beachten, dass YOLO unter der AGPL steht und daher die Pflicht zum Veröffentlichen der eigenen Quellcodes existiert [4]. Darüber hinaus gibt es kommerzielle Anbieter, die Firmenlizenzen für YOLO-Weiterentwicklungen anbieten sowie verschiedene alternative Entwicklungen, wie YOLOX [5] oder Roboflow RF-DETR [7].

Mehr als nur „Was“ und „Wo“

Wer an YOLO denkt, hat meist das Bild von bunten Rechtecken – also den bereits angesprochenen Bounding Boxes – vor Augen, die Autos, Personen oder Werkstücke in Echtzeit markieren. Doch während die dafür erforderlichen KI-Algorithmen in den Entwicklungslaboren der Anbieter, wie Ultralytics [9], sowie der Open-Source-Community, rasant in Richtung Version 11 und darüber hinaus gereift sind, hat sich das Einsatzspektrum fundamental erweitert. Für Maschinenbauer und Anlagenbetreiber bedeutet das: YOLO lässt sich nicht nur als reiner „Detektor“

einsetzen, sondern auch als multifunktionales Werkzeug für die intelligente Prozesssteuerung nutzen, mit dem sich sehr werthaltige Anwendungen realisieren lassen. Darüber hinaus ist eine YOLO-Anwendung für Maschinen und Anlagen im Rahmen eines Retrofits über eine passende Edge-AI-Hardware auch nachträglich hinzufügar.

Was befindet sich wo?

Dabei ist allerdings folgendes zu beachten: Die klassische YOLO-Objekterkennung lässt sich inzwischen zwar recht einfach realisieren, sie beantwortet allerdings lediglich die Frage: „Was befindet sich an welcher Stelle im Bild?“ Für moderne Retrofit-Lösungen ist das oft zu wenig. Hier setzt die anspruchsvollere Instanzsegmentierung an. Statt objektumfassender Rechtecke liefert sie pixelgenaue Masken. In der Qualitätssicherung ermöglicht dies beispielsweise nicht nur die Feststellung, dass ein Bauteil vorhanden ist, sondern auch die exakte Vermessung seiner Konturen oder die Identifikation von kleinsten Materialfehlern und Leckagen, die innerhalb einer rechteckigen Box schlicht untergehen würden.

Pose Estimation

Ein weiterer Gamechanger für Automatisierungsapplikationen ist die Pose Estimation. Hierbei erkennt das Modell spezifische Skelettpunkte eines Objekts. Was im Sport-Streaming für Bewegungsanalysen genutzt wird, dient in der Fabrik der Zukunft der Arbeitssicherheit. Mechanische Systeme können damit automatisch erkennen, ob ein Mitarbeiter eine ergonomisch kritische Haltung einnimmt oder ob eine Hand in einen Gefahrenbereich greift, noch bevor ein physischer Kontakt stattfindet.

Die Hardware-Aspekte

Der entscheidende Erfolgsfaktor solcher Lösungen ist die Verlagerung der Intelligenz auf ein passendes System-on-Module (SoM). Dank der hohen Effizienz moderner YOLO-Modelle ist keine Cloud-Anbindung erforderlich. Die Inferenz erfolgt somit direkt an der Kamera („at the Edge“).

Dies löst auf einen Schlag zwei der größten Probleme industrieller KI: 1. Die Latenz: Reaktionen erfolgen in Millisekunden; sowie 2. der Datenschutz: Da nur Metadaten (z. B. Zählwerte) statt Echtzeit-Videostreams übertragen werden, bleiben die Persönlichkeitsrechte gewahrt und die Netzwerklast ist minimal.

Autor:

Klaus-Dieter Walter
CEO

SSV Software Systems GmbH
www.ssv-embedded.de

VERGLEICH: DETEKTION vs. SEGMENTIERUNG IN DER INDUSTRIE

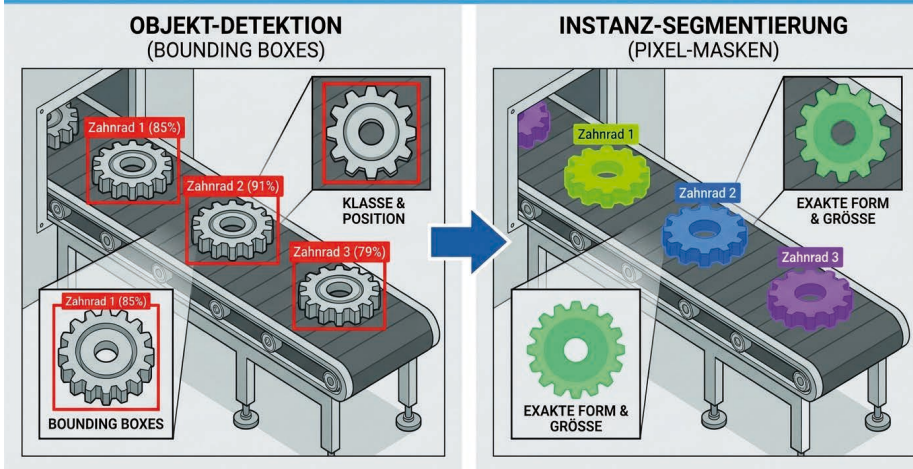


Bild 1: Aus den Videodaten einer Kamera, die im Rahmen eines Retrofits in einer Produktionslinie montiert wird, lassen sich mit Hilfe eines KI-basierten Objektdetektors sowohl Produktions- als auch Qualitätskennzahlen erzeugen.

Beispiel: Monitoring einer Produktionslinie

Wird eine Kamera zusammen mit einer Edge-AI-Hardware am Ende einer Produktionslinie nachgerüstet (Bild 1), ergeben sich durch den Einsatz von YOLO vielfältige Nutzungsmöglichkeiten, wie z. B.:

- **Betriebszustandserfassung:** Über die Echtzeit-Bilddatenanalyse rotierender Maschinenteile lässt sich zuverlässig feststellen, ob sich die Anlage im aktiven Betriebs- oder im Ruhezustand befindet. Dies ermöglicht ein automatisiertes Monitoring der Maschinenlaufzeiten ohne Eingriff in die SPS-Steuerung.
- **Stückzahl-Erfassung:** Die gefertigten Objekte werden automatisiert detektiert und gezählt. Die YOLO-Objekterkennung liefert hierfür präzise Daten, die pro Schicht in eine belastbare Produktionskennzahl übertragen lassen.
- **Qualitätssicherung:** Mittels Instanz-Segmentierung lässt sich die Geometrie und Oberflächenbeschaffenheit der Objekte prüfen. So kann direkt eine Qualitätskennzahl (Yield/Ausschussrate) errechnet werden, um ‚OK‘- von ‚Not-OK‘-Teilen zu unterscheiden.
- **Sicherheits-Monitoring:** Durch die Erkennung von Fremdkörpern oder unbefugtem Zutritt in den Gefahrenbereich der Linie kann das System Warnsignale auslösen und so zur Arbeitssicherheit beitragen.

Beispiel: Indoor-Navigation mit ArUco-Markern

ArUco-Marker sind quadratische, schwarz-weiße Referenzmarken (Fiducial Marker), die von den Kameras mobiler Roboter zur schnellen Positions- und Orientierungsbestimmung (Pose Estimation) in Echtzeit erkannt werden.

Edge-AI-Referenzplattform

Die Edge-AI-Referenzplattform für eine YOLO-Anwendung wäre aktuell zwar der Jetson Orin Nano von NVIDIA [10]. Er hat eine 6-Core Arm Cortex A78AE CPU sowie eine sehr spezielle GPU mit 1024 CUDA-Kernen und 32 Tensor-Knoten. Laut Herstellerangaben sind damit 67 TOPS bei INT8-KI-Modellen möglich. Zu berücksichtigen ist allerdings, dass sich die TOPS-Angaben der KI-Chiphersteller und die damit möglichen YOLO-Inferenzzeiten nicht direkt vergleichen lassen.

Zu hohe Kosten

Auf Grund der Hardwarekosten und des Strombedarfs kommt für viele YOLO-Anwendungen eine NVIDIA-Hochleistungsplattform nicht in Frage. Bei der Suche nach SoM-basierten Alternativen und der für eine bestimmte YOLO-Anwendung minimal erforderlichen TOPS-Anzahl [1] ist allerdings darauf zu achten, dass neben einer Mehrkern-CPU unbedingt auch eine GPU, TPU bzw. NPU vorhanden ist. Ohne eine solche Zusatzfunktion ist YOLO nicht wirklich Praxistauglich.

Anforderungen bestimmen die Auswahl

Am Anfang einer YOLO-Edge-AI-Entwicklung sollte man daher einfach die individuelle YOLO-Inferenzzeitspanne auf verschiedenen SoM-Eval-Boards ausmessen, mit der erforderlichen Zielbildrate (z. B. 5 FPS) multiplizieren, das Ergebnis mit den Anforderungen abgleichen und das jeweils beste Preis/Leistungsverhältnis wählen (zwischen einem Raspberry-Pi mit Coral Edge TPU am unteren Ende und dem Jetson Orin Nano als gegenüberliegenden Leistungsendpunkt existieren inzwischen viele interessante Edge-AI-Chips).

Alternativen

Lässt der angestrebte Zielpreis oder die Batteriespannungsversorgung den Einsatz einer Multi-Core-CPU plus GPU/TPU/NPU nicht zu, muss man auf YOLO verzichten und Alternativen sichten. Evtl. reicht OpenCV auf einer Low-End Embedded-Linux-Hardware (SoC mit einigen MBytes DRAM plus eMMC) aus – die visuelle Navigation eines Intralogistikroboters mit ArUco-Markern ist damit ohne weiteres möglich. Denkbar wäre auch die MCU-taugliche Speziallösung, wie z. B. „FOMO (Faster Objects, More Objects)“ des Anbieters Edge Impulse [2]. Für einfache Objekterkennungen lassen sich damit sehr kompakte KI-Modelle erzeugen, die sogar mit dem MCU-internen Flash- und RAM-Speicherressourcen auskommen.

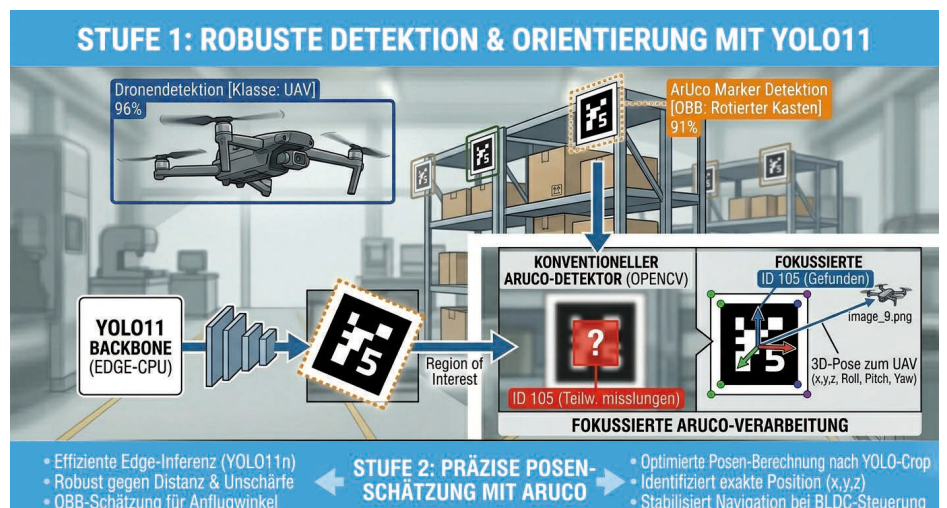


Bild 2: Für die Roboter- bzw. Drohnennavigation in der Prozessautomatisierung eignen sich bildbasierte Verfahren mit sogenannten ArUco-Markern

Mit YOLO-Instanzsegmentierung zonenbasiert heizen & Energie sparen.

Erkennt Personen und steuert Heizventile bedarfsgerecht in jeder Zone.



Bild 3: Mit KI-Werkzeugen wie YOLO lässt sich z. B. auch die Energieeffizienz in der Gebäudetechnik verbessern.

Sie bestehen aus einem breiten schwarzen Rand und einem internen Binär-Raster, das eine eindeutige ID codiert. Gängige ArUco-Marker-Rastergrößen reichen von 4×4 bis 7×7 Bit. Sie werden neben der Roboternavigation auch für Augmented Reality (AR) und andere Aufgaben des maschinellen Sehens eingesetzt.

Funktionsweise

Die Funktionsweise ist einfach: Eine Kamera erkennt den Marker und mittels Standard-Detektor-Algorithmen (z. B. per OpenCV [8]) werden Position und Neigung im Raum berechnet. Weitere Vorteile neben der schnellen Erkennung sind die einfache Erstellung und die insgesamt kostengünstige Lokalisierung.

Herausforderungen

In Produktions- und Lagerhallen sind die Leseergebnisse mit Standard-Detektoren häufig unzureichend. Die Ursachen sind zu große Leseentfernungen, schlechte Lichtverhältnisse, Schatten und Reflexionen an Metallregalen. Teilweise kommen bei z. B. Drohnen aber auch noch vibrationsbedingte Bewegungsunschärfen hinzu.

In solchen Problemfällen hilft eine zweistufige Lösung. In der ersten Stufe wird von YOLO das Gesamtbild nach dem Marker abgesucht. Auf den gefundenen Bildausschnitt (Crop Image) wird in der zweiten Stufe der klassische ArUco-Algorithmus angewendet, um die exakte 3D-Position und Orientierung (Pose) der Drohne zum Marker zu berechnen (Bild 2).

Beispiel: Energieeffizienz in Gebäuden

Auch im Bereich der Gebäudeautomatisierung lassen sich mit Hilfe von YOLO-KI-Modellen verschiedene innovative Anwendungen realisieren. Als Anwendernutzen eignet sich hier besonders die Optimierung der Energiekosten,

beispielsweise durch einen deutlich intelligenteren Betrieb der Gebäudeautomationssysteme für Beleuchtung und Klima.

In Bild 3 wird z. B. Hilfe der bereits vorgestellten Instanz-Segmentierung eine bedarfsgerechte Gebäudeheizung dargestellt. Im Gegensatz zu klassischen Präsenzsensoren erkennt das System mit einem KI-Modell nicht nur, ob sich Personen in einem Raum befinden, sondern auch deren genaue Position innerhalb definierter Heizzonen. Eine Deckenkamera erfasst den Raum kontinuierlich, während ein Edge-AI-Gateway die Bilddaten lokal verarbeitet und jeder Person eine spezifische Zone zuordnet. Dadurch können Heizkörperventile bedarfsgerecht angesteuert werden: Bereiche mit aktiver Nutzung werden komfortabel beheizt, ungenutzte Zonen automatisch abgesenkt. Das reduziert Energieverbrauch und Heizkosten erheblich, ohne den Komfort zu beeinträchtigen. Durch die lokale Inferenz verbleiben sämtliche Videodaten im Gebäude, wodurch datenschutzkonforme Lösungen für Büros, Besprechungsräume oder Gewerbeimmobilien möglich werden.

Visuelle Intelligenz als Mehrwert

Durch die Kombination einer Kamerasensorik mit den YOLO-KI-Funktionen sowie einem anwendungsspezifischen Deep Learning-Software-Stacks [11] entsteht eine anpassungsfähige visuelle Intelligenz mit völlig neuen Möglichkeiten. Sie ermöglicht nicht nur die Senkung von Betriebskosten sowie die Steigerung der Arbeitssicherheit. Für Anlagenbetreiber ist das Retrofitting mit einer solchen Funktionskombination darüber hinaus ein effizienter Weg, mit bestehender Technik den multidimensionalen Herausforderungen der Zukunft zu begegnen. Der Blick durch die Linse der KI ist heute schärfer, tiefer und wertvoller denn je.

Quellenangaben

- [1] Die Suche nach der minimal erforderlichen TOPS-Anzahl für eine YOLO-Applikation: https://www.reddit.com/r/embedded/comments/1gxhi7e/how_does_one_determine_the_minimum_tops/?tl=de
 - [2] Edge Impulse FOMO Algorithm <https://docs.edgeimpulse.com/studio/projects/learning-blocks/blocks/object-detection/fomo>
 - [3] Siehe „YOLO: Eine kurze Geschichte“ <https://docs.ultralytics.com/de/#where-to-start>
 - [4] Siehe „YOLO ist NICHT wirklich Open Source und du kannst es nicht kommerziell nutzen ...“ https://www.reddit.com/r/computervision/comments/1gxce90/yolo_is_not_actually_opensource_and_you_cant_use/?tl=de
 - [5] Siehe „YOLOX is an anchor-free version of YOLO, with a simpler design but better performance“ <https://github.com/megvii-base-detection/yolox>
 - [6] Eaton Easy E4 für Wohn- und Zweckbauten [https://www.eaton.com/content/dam/eaton/products/industrialcontrols-drives-automation-sensors/en-globalprime/programmable-logic-controllers-\(plc\)/easye4-programmable-relays/eaton-easye4-control-relay-flyer-f1050012de-de-de-de.pdf](https://www.eaton.com/content/dam/eaton/products/industrialcontrols-drives-automation-sensors/en-globalprime/programmable-logic-controllers-(plc)/easye4-programmable-relays/eaton-easye4-control-relay-flyer-f1050012de-de-de.pdf)
 - [7] Roboflow Detector <https://blog.roboflow.com/train-rf-detr-on-a-custom-dataset/sowie https://universe.roboflow.com/search?q=object+detection+model+model%3Arfdetr>
 - [8] OpenCV-Dokumentation „Detection of ArUco Markers“. Siehe https://docs.opencv.org/3.4/d5/dae/tutorial_aruco_detection.html
 - [9] Ultralytics YOLO26 als Echtzeit-Objektdetektor für Edge- und Low Power-Geräte <https://docs.ultralytics.com/de/models/yolo26/>
 - [10] NVIDIA „Deep Learning Institute“, siehe z. B. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-RX-02+V2
 - [11] MIT Open Course Ware „Deep Learning“ siehe, <https://ocw.mit.edu/courses/6-7960-deep-learning-fall-2024/download/>
- Siehe auch „Roboter im Alltag: OSS You Only Look Once (YOLO)“: <https://www.roboter-im-alltag.org/docs/oss-you-only-look-once-yolo/> ◀