

CRASH COURSE VISION WITH PYTHON



Studiedag Landelijk Overleg Mechatronica
13 maart 2025
Gerard Harkema



Onderwerpen



Wat achtergrond informatie



Demonstraties



Brainstormen

Klassieke computer vision

De klassieke versie van **computer vision** maakt gebruik van traditionele beeldverwerkingstechnieken en algoritmes op basis van wiskunde en statistiek.

- Kleurruimte transformeren
 - RGB → Grayscale
 - RGB → HSV
- Filters
- Morphologie
- Blob detectie



Toepassingsgebieden

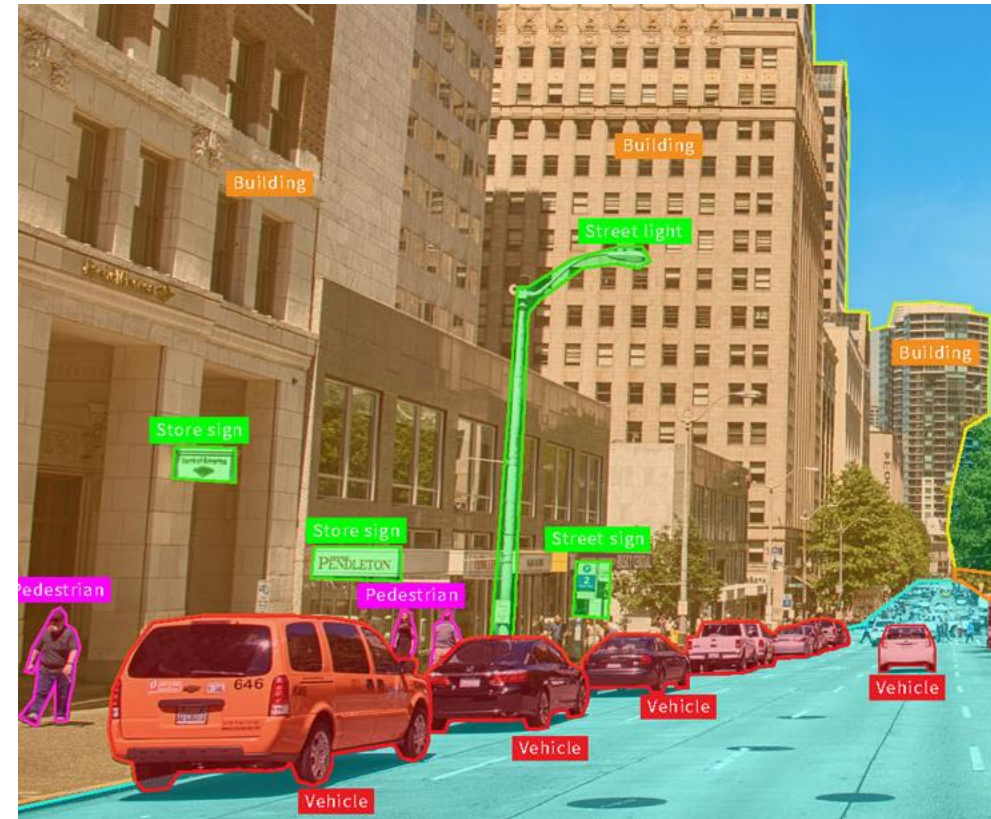
- ✓ **Industriële kwaliteitscontrole** – Automatische detectie van defecten in productielijnen.
- ✓ **Optische karakterherkenning (OCR)** – Het omzetten van gescande tekst naar bewerkbare tekst.
- ✓ **Bewegingsdetectie** – Wordt gebruikt in beveiligingscamera's en bewegingssensoren.
- ✓ **Medische beeldanalyse** – Verbetering van medische scans en röntgenfoto's.

Technieken

- ◆ **Randdetectie** – Algoritmes zoals Canny en Sobel om randen van objecten in een afbeelding te vinden.
- ◆ **Segmentatie** – Het verdelen van een afbeelding in verschillende regio's, bijvoorbeeld met k-means clustering of thresholding.
- ◆ **Contourdetectie** – Het identificeren van objectvormen met technieken zoals de Hough-transformatie.
- ◆ **Morfologische bewerkingen** – Het verbeteren van beeldkwaliteit met methoden zoals dilatie en erosie.
- ◆ **Feature extractie** – Het herkennen van patronen met descriptoren zoals SIFT (Scale-Invariant Feature Transform) en SURF

Computer vision (machine learning)

Machine Learning/Computer vision is een tak van kunstmatige intelligentie (AI) die computers in staat stelt om visuele informatie uit de wereld te interpreteren en begrijpen, net zoals mensen dat doen met hun ogen en hersenen. Het maakt gebruik van technieken zoals beeldherkenning, objectdetectie, gezichtsherkenning en videobeeldanalyse.



Toepassingsgebieden

- ◆ **Gezichtsherkenning** – Gebruikt in beveiligingssystemen en smartphones.
- ◆ **Autonome voertuigen** – Helpt zelfrijdende auto's om objecten, voetgangers en verkeersborden te herkennen.
- ◆ **Medische beeldvorming** – Assisteert artsen bij het analyseren van röntgenfoto's, MRI-scans en CT-scans.
- ◆ **Retail en productie** – Wordt gebruikt voor voorraadbeheer, kwaliteitscontrole en automatische kassasystemen.
- ◆ **Augmented reality (AR)** – Verrijkt de echte wereld met digitale objecten, bijvoorbeeld in games en navigatie.

Python

Python is een populaire programmeertaal die bekend staat om zijn eenvoud en leesbaarheid. Het wordt veel gebruikt voor

- Webontwikkeling
- **data-analyse(lees vision)**
- kunstmatige intelligentie (AI)
- machine learning
- automatisering, en wetenschappelijk onderzoek.



Ontworpen door Nederlander:
Guido van Rossem in 1989(al)



Waarom Python

- ✓ **Makkelijk te leren** – De syntaxis lijkt op gewone taal, wat het toegankelijk maakt voor beginners.
- ✓ **Groot ecosysteem** – Er zijn veel bibliotheken zoals NumPy, Pandas (data-analyse), TensorFlow (AI), OpenCV (beeldverwerking), Flask (webontwikkeling), en meer.
- ✓ **Veelzijdig** – Je kunt het gebruiken voor webapps, games, automatisering, data science en zelfs embedded systemen.
- ✓ **Grote community** – Omdat veel mensen Python gebruiken, is er veel documentatie en ondersteuning beschikbaar.

OpenCV

OpenCV (Open Source Computer Vision Library) is een open-source bibliotheek voor computer vision en beeldverwerking. Het biedt tools en functies om beelden en video's te analyseren en te manipuleren. OpenCV wordt veel gebruikt in toepassingen zoals gezichtsherkenning, objectdetectie en machine learning.



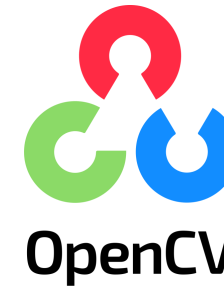
OpenCV

Toepassingen

- Gezichts- en objectdetectie
 - gezichtsherkenningssystemen
- Autonome voertuigen
 - Rijstrookdetectie
 - obstakelvermijding
- Medische beeldverwerking
 - analyse van MRI- en röntgenbeelden
- Augmented Reality (AR)
 - digitale objecten projecteren op echte beelden
- Robotica & industriële automatisering
 - kwaliteitscontrole in fabrieken



Kenmerken OpenCV



- ✓ **Beeldverwerking** → Filters, randen detecteren, kleuren aanpassen en transformaties.
- ✓ **Object- en gezichtsdetectie** → Functies voor gezichtsherkenning, contourdetectie en objecttracking.
- ✓ **Machine Learning** → Ingebouwde modellen voor patroonherkenning en classificatie.
- ✓ **Videobewerking** → Beweging detecteren, videoframes bewerken en analyseren.
- ✓ **Integratie met Deep Learning** → Werkt samen met frameworks zoals TensorFlow, PyTorch en ONNX.
- ✓ **Ondersteuning voor meerdere platforms** → Werkt op Windows, Linux, macOS, Android en iOS.

OpenCV met C++

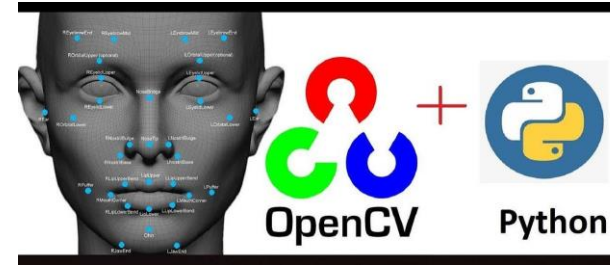
✓ **Voordelen:**

- **Sneller & efficiënter** → Geschikt voor real-time beeldverwerking en CPU-intensieve taken.
- **Betere controle over geheugen** → Handmatige geheugenbeheer voorkomt overhead van garbage collection.
- **Directe OpenCV-integratie** → OpenCV is oorspronkelijk in C++ geschreven, waardoor sommige functies beter presteren.
- **Geschikt voor embedded systemen** → Beter voor toepassingen op low-power hardware zoals Raspberry Pi of microcontrollers.

✗ **Nadelen:**

- **Trage leercurve** → Meer complexe syntax en expliciet geheugenbeheer.
- **Langzamere ontwikkeltijd** → Meer regels code nodig voor dezelfde functionaliteit.
- **Minder flexibel** → Moeilijker te integreren met AI/ML-bibliotheken die vaak in Python worden gebruikt.

OpenCV met Python



avans
hogeschool

✓ **Voordelen:**

Gebruiksvriendelijk & leesbaar → Python-code is eenvoudiger te schrijven en te begrijpen.

Snelle ontwikkeling → Sneller prototypes, geen include voor prototype definitie files

Rijke bibliotheken → Makkelijk te combineren met andere AI/ML-bibliotheken zoals TensorFlow en NumPy.

Dynamisch getypeerd → Geen gedoe met datatypes zoals in C++.

✗ **Nadelen:**

Langzamer dan C++ → Python is geïnterpreteerd en minder efficiënt bij zware berekeningen.

Beperkte controle over geheugen → Garbage collection kan onvoorspelbaar zijn.

Minder efficiënt bij real-time toepassingen → Vooral belangrijk bij high-performance toepassingen zoals videoverwerking.

Welke kies je



Kies Python:

als je snel wilt prototypen
AI/ML-integratie nodig hebt
minder ervaring hebt met C++.



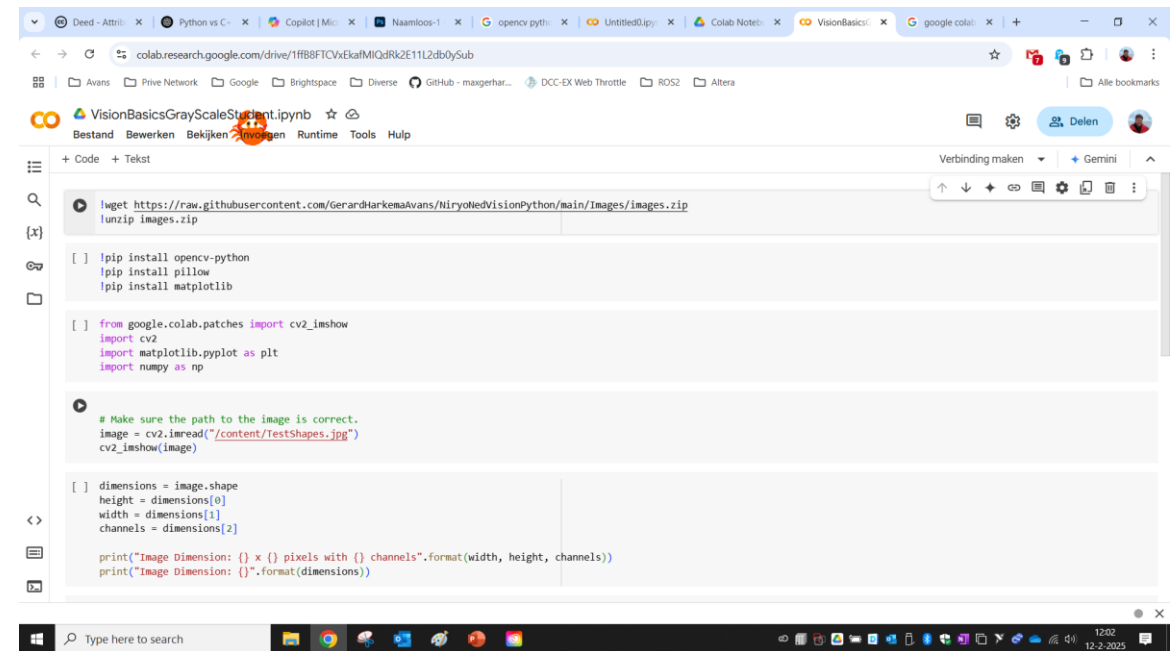
Kies C++ als prestaties:

real-time verwerking
embedded systemen belangrijk
zijn.

Colab (Google Colaboratory)



- Cloudbaseerde Jupyter Notebook-omgeving
 - **Machine learning,**
 - **Data-analyse**
 - **Python-programmering.**
- Het biedt een krachtige, eenvoudig te gebruiken interface waarmee je code kunt schrijven en uitvoeren zonder dat je software op je eigen computer hoeft te installeren.



Toepassingen



Machine Learning & Deep Learning

trainen van AI-modellen
• TensorFlow/PyTorch



Data-analyse & Visualisatie

Pandas/Matplotlib/Seaborn



Computer Vision & OpenCV-projecten

Beeldherkenning en objectdetectie



Natural Language Processing (NLP)

tekstanalyse en chatbot-ontwikkeling



Automatisering & Web Scraping

data verzamelen
• BeautifulSoup/Selenium

Kenmerken

- ✓ **Gratis toegang tot GPU's en TPU's** → Versnelt AI/ML-berekeningen.
- ✓ **Draait in de cloud** → Geen installatie nodig, werkt direct in de browser.
- ✓ **Ondersteunt Jupyter Notebooks** → Gebruik Python, NumPy, Pandas, TensorFlow, OpenCV en meer.
- ✓ **Gemakkelijk delen & samenwerken** → Net als Google Docs, meerdere mensen kunnen tegelijk werken.
- ✓ **Koppeling met Google Drive** → Sla bestanden en datasets op in de cloud.
- ✓ **Automatische omgevingsbeheer** → Geen gedoe met dependencies, alles wordt vooraf geïnstalleerd.

Waarom Colab



◆ **Ideaal voor beginners** → Geen installatie, makkelijk te gebruiken.

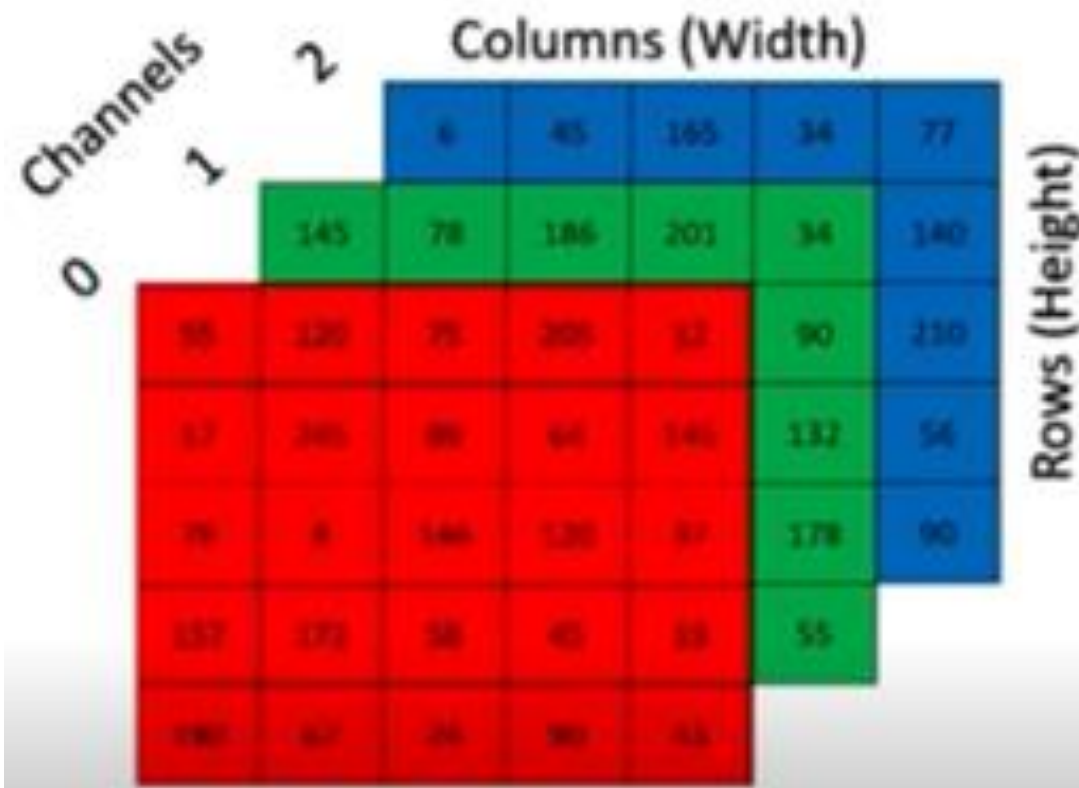


◆ **Krachtig voor AI & Deep Learning** → Gratis toegang tot GPU's en TPU's.



◆ **Perfect voor samenwerking** → Deel Notebooks met anderen in één klik.

Wat is een afbeelding?



A single RGB Image can also be represented as a 3D array with a depth of 3, on account of having 3 channels (R, G and B)

Demonstraties



De course website

<https://avansmechatronica.github.io/CrashCourseVisionWithPython/>

Brainstormen

- Hoe kan de toekomst van klassieke Vision in het Mechatronica onderwijs eruit zien?

