

Geautomatiseerde Afvalclassificatie met AI – Sensoring & Monitoring



Inhoud

Achtergrond	3
Groepsindeling	3
Opdrachtoomschrijving	5
Sensing	6
Monitoring	7

Achtergrond

Welkom bij het project “Slimme Zwerfafvalherkenning”!

Breda (de klant) ziet door de stad heen zwerfafval. De gemeente wil zijn medewerkers efficiënt in te kunnen zetten. Daarom is er vraag naar het kunnen detecteren van afval op de verschillende locaties en het kunnen maken van een optimale schoonmaakplanning.

In de komende weken ga je samen met je projectteam werken aan een innovatieve applicatie die zwerfafval kan herkennen én voorspellingen kan doen (afhankelijk van jullie keuze van subsysteem) over waar en wanneer afval zich ophoopt. Jullie zetten data science en slimme technologie in om een actueel probleem op te lossen en zo een bijdrage te leveren aan een schonere leefomgeving.

Daarnaast oefen je met vaardigheden die belangrijk zijn in de praktijk, zoals samenwerken in sprints, klantgericht denken en het presenteren van jouw oplossing. Je gaat ook aan de slag met ethiek en AI, waarbij je nadenkt over de maatschappelijke impact van de applicatie die jullie ontwikkelen. Denk bijvoorbeeld aan vragen als: *Hoe ga je verantwoord om met data? Wat zijn de ethische overwegingen bij het gebruik van AI voor het herkennen van zwerfafval?*

Het doel van dit systeem is om niet alleen afval te detecteren, maar specifiek te bepalen wat voor type afval het is. Dit draagt bij aan efficiënter afvalbeheer en recyclingprocessen. Jullie ontwikkelen een functioneel prototype waarbij de output van het sensing subsysteem automatisch wordt opgeslagen en geanalyseerd voor verdere monitoring en gebruik. Door computervisie en sensorintegratie te combineren, wordt het mogelijk om real-time detectiegegevens te verzamelen en inzicht te krijgen in afvalpatronen.

Het project is verdeeld in verschillende onderdelen:

1. **Sensorische detectie en classificatie:** In deze fase wordt afval in realtime herkend en geclassificeerd met behulp van een AI-model en een OAK-D-camera.
2. **Gegevensbeschikbaarheid en integratie:** De verzamelde en geclassificeerde afvaldata wordt via een API opgeslagen in een database en beschikbaar gesteld voor verdere analyse en visualisatie.
3. Het **analyseren** van deze afvaldetectiedata om patronen te ontdekken en voorspellingen te doen. Dit gebeurt door het inzetten van een AI-model als een REST API, waarbij **Decision Tree en/of Random Forest** worden gebruikt als AI modellen.
4. Het **monitoren** door middel van een applicatie die het AI model bevraagt en voorspellingen kan doen.

Groepsindeling

In dit project werk je in **groepen van 5 personen**. De groepen mogen jullie zelf maken. Samen gaan jullie aan de slag met het ontwikkelen van een innovatieve oplossing rond zwerfafvalherkenning. Elk team maakt een keuze: **Sensing** óf **Monitoring**.

- Kies je voor **Sensing**, dan richt je je op het verzamelen van data door beeldherkenning. Jullie zorgen ervoor dat de applicatie zwerfafval kan herkennen en relevante gegevens verzamelt, dus onderdelen 1 en 2.
- Kies je voor **Monitoring**, dan richt je je op de analyse en visualisatie van de data. Jullie team bouwt bijvoorbeeld een dashboard dat voorspellingen doet over waar en wanneer zwerfafval zich ophoopt, en dat inzicht geeft in de data die de Sensing-teams verzamelen, dus onderdelen 3 en 4.
- De **Sensing- en Monitoring-applicaties worden met elkaar geïntegreerd**. Zo zorgen jullie er samen voor dat de complete oplossing werkt: van het herkennen van zwerfafval tot het inzichtelijk maken van de data en voorspellingen.

Binnen de klas wordt er naar de verdeling gekeken. **Eén groep Sensing werkt met een Monitoring groep samen, maximaal met 2 groepen**. Het kan dus zo zijn dat er gevraagd wordt om te switchen van onderwerp, als de verhouding niet klopt binnen de klas.

Opdrachtomschrijving

Je start met het beschrijven en uitwerken van je casus. Je bent behoorlijk vrij om een creatieve casus te bedenken. Uiteraard moet het algemene doel, het terugdringen van zwerfafval, centraal staan. Omdat iedere sensing-groep samenwerkt met minimaal één monitoring groep is het belangrijk dat jullie gezamenlijk nadenken over de casus. **Stemt dit dus van te voren met elkaar af!**

Om jullie wat houvast te geven hierbij in het kort een tweetal voorbeelden van een casus.

1. Statische camera's

Als gemeente (Breda) wil ik een bepaald gebied monitoren op zwerfafval.

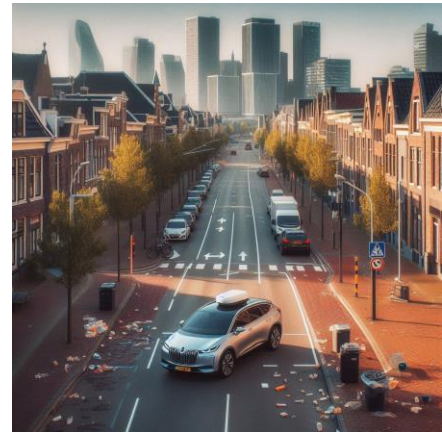


Figuur 1 Camera's in park. (bron Copilot)

Denk bijv. Aan een park of stationsgebied. De uitgangspunten zijn hierbij dat er op strategische plaatsen camera's zijn opgesteld die samen het compleet gebied kunnen waarnemen. Periodiek (bijv. ieder kwartier) wordt een afbeelding gemaakt en de huidige situatie wat betreft afval in kaart gebracht. Aan de monitoring kant kun je dan bijv. iets doen met de soorten en hoeveelheid afval om enerzijds de benodigde opruimcapaciteit te voorspellen en wellicht de bron te achterhalen of voorspellen om deze aan te kunnen spreken. Hierbij mag je bepaalde aannames doen: bijv. de camera's overlappen elkaar niet. Per tijdseenheid wordt de huidige situatie in kaart gebracht.

2. Dynamische camera

De gemeente (Breda) is van plan het zwerfvuil in een bepaald gebied te monitoren met de gevreesde scanauto. Naast het opsporen van foutparkeerders brengt deze auto nu ook het zwerfvuil in kaart. Om dit mogelijk te maken wordt een extra camera op de auto geplaatst, deze brengt de hoeveelheid zwerfvuil in kaart en koppelt dit aan de positie waar dit zwerfvuil is waargenomen. De achterliggende gedachte van deze casus is tweeledig: enerzijds het efficiënt inzetten van opruimcapaciteit, maar als belangrijkste doel het terugdringen van afval bij de bron. Zo zou bijvoorbeeld een plaatselijk café kunnen worden aangesproken als ergens veel plastic bekertjes worden aangetroffen.



Figuur 2 Camera op auto. (bron: Copilot)

Ook voor deze casus geldt dat je een aantal aannames mag doen: ga er bijvoorbeeld vanuit dat de camera's een gebied in kaart brengen als een soort panoramafoto. Iedere afbeelding past dus precies naast de vorige en ook hier is geen overlap. Je mag de locatie eventueel 'faken', wanneer je dit doet moet het wel zo realistisch mogelijk gebeuren: neem bijv. de coördinaten van een bepaalde straat en laat de auto een min of meer logische route rijden.

Wij zijn erg benieuwd naar jullie creatieve voorstel. Let op: bespreek dit voorstel vooraf wel met je (proxy-) productowner.

Sensing

Voor iedere casus geldt:

- **Train en implementeer een AI-model** dat verschillende soorten afval kan classificeren met behulp van YOLOv8.
- **Integreer de OAK-D-camera** om real-time detectie en classificatie mogelijk te maken.
- **Ontwikkel een API** die de gedetecteerde afvalcategorieën opslaat in een database.
- Zorg ervoor dat **de verzamelde data beschikbaar wordt gesteld** voor de monitoring groep, waar verdere verwerking en analyse plaatsvindt.
- Koppel met een relevante API naar keuze om de data te bekijken.

Randvoorwaarden

- De AI-classificatie moet in realtime functioneren.
- De output van de sensorische detectie moet gestructureerd en toegankelijk zijn voor gebruik in de monitoring fase.
- Er moet gebruik worden gemaakt van bestaande datasets en transfer learning om het model efficiënter te trainen, test met eigen videobeelden. Er is echter ook ruimte om een

eigen dataset te maken en een eigen model te trainen, mits rekening wordt gehouden met de compatibiliteit met de OAK-D-camera.

- Tools: .NET Core (API), SQL Server (database), Roboflow (datasetbeheer).
- API-functionaliteit: Moet minimaal POST (data opslaan) en GET (data ophalen) ondersteunen.

Minimale output per detectie:

- Confidence
- Datum & Tijd
- Label (wat voor afval)
- Afhankelijk van de door jullie gekozen casus kan het nodig zijn extra informatie op te slaan en beschikbaar te stellen. Denk bijv. aan locatie of camera(id).

Dit project vraagt om samenwerking, probleemoplossend vermogen en technische implementatievaardigheden. Zorg voor een duidelijke documentatie van jullie keuzes en aanpak. Succes!

Monitoring

Voor iedere casus geldt:

Randvoorwaarden

- **Het AI-model wordt ontwikkeld in Python** en gedraaid als een REST API met **FastAPI**.
- **De API wordt automatisch gedeployed als een container** met **Docker** en geïmplementeerd in **Azure**.
- **De dataset wordt ingelezen, verwerkt en geanalyseerd** om voorspellingen te doen.
- **Decision tree en/of Random Forest** worden gebruikt voor de AI-modellen.
- **De API moet een endpoint bieden** waar voorspellingen kunnen worden opgevraagd.
- Koppel met een relevante API naar keuze om de data te verkrijgen

Technologieën

- **Python (Scikit-learn, Pandas, NumPy, FastAPI)**
- **Docker (voor containerisatie van de AI-service)**
- **Azure (voor clouddeployment van de API)**
- **SQL Server**

Verwachte Output

1. **Een werkende REST API** waarin een gebruiker een locatie en omgevingsparameters kan invoeren en een voorspelling ontvangt over de ophaalprioriteit of het type afval.
2. **Een Decision Tree en Random Forest-model** dat is getraind en geëvalueerd op historische data.
3. **Een containerized AI-service**, gedraaid met Docker en gedeployed op Azure.
4. **Technische documentatie** waarin de keuzes en implementatiestappen worden toegelicht.
5. **Een Monitoring applicatie** waarin een gebruiker een locatie en omgevingsparameters kan invoeren en een voorspelling ontvangt over de ophaalprioriteit of het type afval door aanroepen van eerder genoemde API.