Trash Recogniser

PRG-8 Eindopdracht

Door Conan Spithoven, 0915670



Inhoudsopgave

[Concept 3](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774620)

[Beoogde doelgroep 3](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774621)

[Toegevoegde waarde van AI 3](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774622)

[Keuze voor gebruikt algoritme 4](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774623)

[Data & acquisitie 4](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774624)

[Eindproduct 4](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774625)

[Uitwerking prototype applicatie 5](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774626)

[Naamgeving prototype 5](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774627)

[Feedback doelgroep 5](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774628)

[Concept 5](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774629)

[UI 5](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774630)

[Technische werking 6](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774631)

[Accuracy 7](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774632)

[Techniek 7](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774633)

[Algoritme 7](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774634)

[Data 7](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774635)

[Training 8](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774636)

[Data Voorkeur 8](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774637)

[Accuracy 8](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774638)

[Privacy Concerns 8](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774639)

[Conclusie 9](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774640)

[Feedback 9](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774641)

[Bruikbaar als eindproduct? 9](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774642)

[AI waarde 9](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774643)

[Eindresultaat 9](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774644)

[Bruikbaarheid 9](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774645)

[Bronnen 11](file:///C:\xampp\htdocs\ML5\TrashRecogniser\Trash-Recogniser.docx#_Toc107774646)

# Concept

Gemeente Rotterdam noemt zwerfafval: “een van de grootste ergernissen in de stad. Het zorgt voor een rommelig straatbeeld en is schadelijk voor dieren en het milieu.” (Gemeente Rotterdam, z.d.).

In mijn TLE project doe ik een poging om het zwerfafval in de regio Rotterdam te verminderen door een applicatie te maken die recycling zou promoten. De applicatie maakt gebruik van fotoherkenning om zwerfafval waars statiegeld op zit te onderscheiden van zwerfafval waar geen statiegeld op zit. Door de statiegeld beloning duidelijk te maken hoop ik vooral jonge mensen te motiveren om deze in te leveren.

Om de scope van het project beperkt te houden focus ik mij op drinkverpakkingen. Buiten dat dit mijn project duidelijk afbakent, bestaat een substantieel deel van het zwerfafval binnen Roterdam uit drinkverpakkingen: “Het zwerfafval dat niet in de afvalbak terecht komt, bestaat vooral uit weggegooide of achtergelaten blikjes, petflesjes, patatbakjes, snoepverpakkingen, sigarettenpeuken en kauwgom.” (Gemeente Rotterdam, z.d.).

Deze drinkverpakkingen zijn grotendeels van PET Flessen, drinkpakken en blik. Op dit moment wordt er statiegeld geheven op PET flessen. Vanaf volgend jaar gaat er ook statiegeld geheven worden op blik (Statiegeld Nederland, z.d.), door het herkennen van blik nu al in de applicatie te bouwen is deze alvast klaar voor de (nabije) toekomst. Hoewel er geen statiegeld op geheven wordt op drinkpakken is deze wel aan te bieden voor recycling.

## Beoogde doelgroep

De beoogde doelgroep van de applicatie is jeugd. De redenen hiervoor zijn tweeledig: hoge mate van smartphone gebruik en het relatief klein besteedbaar inkomen.

De jeugd binnen Nederland scoort steeds hoger in digitale vaardigheid, volgens een onderzoek van de CBS scoort 78% van de jeugd meer dan basale digitale vaardigheden (Arends-Tóth, 2020). Daarnaast is het Smartphone gebruik onder de jeugd ook het grootst (idem, zie vorige referentie). Door de applicatie mobielvriendelijk te maken is de adoptie van de applicatie door jeugd het grootst.

Jeugd tussen de 15 en 25 jaar heeft gemiddeld een inkomen van maar €10000 per jaar. Hierdoor is de financiële beloning voor het inleveren van statiegeld een stuk aantrekkelijker dan voor een oudere doelgroep. Door de waarde van hun afval zichtbaar te maken hoop ik bij de kunnen dragen aan de vermindering van zwerfafval in de gemeente Rotterdam.

De beoogde eindgebruiker en doelgroep is hierom jeugd tussen de 15 en 25 jaar, gezien de financiële beloning van statiegeld hier het sterkt gevoeld worden en zij beschikken over het werktuig om gebruik te maken van de applicatie.

## Toegevoegde waarde van AI

Veel mensen zijn niet bewust dat steeds meer producten statiegeld opleveren of gerecycled kunnen worden. Een applicatie die afval kan scannen voor mogelijk statiegeld zorgt ervoor dat mensen die onbekend zijn met de meest recente regelgeving, toch gemotiveerd kunnen worden om afval in te leveren.

Helaas is een standaard beslissingsboom of het scrollen van referentieplaatjes niet iets wat bij beoogde doelgroep past. Om de drempel tot recyclen zo klein mogelijk te maken is er gekozen voor foto herkening. Dit is namelijk snel en makkelijk te doen vanaf een smartphone, iets wat 84% van de jeugd onder de 25 altijd bij zich draagt (Arends-Tóth, 2020). Daarnaast geeft het een geringe gamification element, wat het plezier van het recycling proces hopelijk vergroot. Dit helpt hopelijk de bereidheid tot recycling binnen de jeugd.

## Keuze voor gebruikt algoritme

Zoals hierboven genoemd wordt er gebruik gemaakt van fotoherkenning omdat dit snel resultaat oplevert en een zekere mate van gamificatie oplevert. Beide zouden de doelgroep moeten aansporen om meer te doen aan recycling van drankverpakkingen.

Ik heb gekozen voor de ML5 Feature Extractor voor mijn algoritme. Dit fotoherkenning algoritme maakt gebruikt van een training set die zelf aangeleverd kan worden en nieuwe foto’s kunnen geüpload worden voor herkenning. Verder is de AI door duidelijke documentatie en vele open-source voorbeelden snel te implementeren zodat er met een prototype getest kan worden.

Ik heb specifiek gekozen voor de ML5 Feature Extractor(FE) i.p.v de ML5 Image Classifier(IC) omdat de FE zich meer focust op het herkennen van objecten in een afbeelding en de IC zich meer focust op het herkennen van de afbeelding in zijn geheel, waardoor de accuracy van de app mogelijk zou dalen.

Een andere mogelijk keuze van de Teacheable Machine van Google, maar deze is niet gekozen omdat de library hiervan moeilijker is om uit te breiden door een gelimiteerde hoeveelheid functies die je aan kan passen.

## Data & acquisitie

Voor dit project heb ik twee sets data nodig: een training set en een controle set. De training set dient om de algoritme het verschil tussen verschillende drankverpakkingen te leren. Vervolgens wordt dit getrainde model losgelaten op de controle set om de accuracy van het model te meten.

Deze data wordt op twee manieren verzameld. Allereerst zal ik foto’s maken van zwerfafval in mijn eigen buurt. Daarnaast zal ik deze aanvullen met foto’s van zwerfafval van internet. Deze aanvulling is nodig om een bredere selectie voor de training set te creëren. Mijn omgeving hoeft namelijk niet representatief te zijn voor alle soorten drankverpakkingen vindbaar in de gemeente Rotterdam.

## Eindproduct

Het eindproduct zal de vorm aannemen van een website, door middel van responsive styling is de website ook gemakkelijk op mobile devices te gebruiken. Hierdoor is de drempel tot gebruik laag, er hoeft namelijk niets geïnstalleerd te worden en de applicatie is makkelijk toegankelijk voor de beoogde doelgroep.

De UI wordt zo simpel mogelijk gehouden om het gebruik te promoten. Voor eindgebruikers is er één pagina waar ze de foto kunnen uploaden en vervolgens het resultaat terugkrijgen.

Daarnaast zal er een aparte admin pagina waar het model ingeladen kan worden voor gebruik. Dit maakt het vervangen van het model door een nieuw getraind model mogelijk.

Verder kan het eindproduct later ook uitgebreid worden zodat het model tijdens gebruik verder getraind word, waardoor de accuracy constant verbeterd kan worden.

# Uitwerking prototype applicatie

Nadat ik besloten had op het ontwerp ben ik aan de slag gegaan met het uitwerken hiervan. Hier kwam al snel een prototype uit.

## Naamgeving prototype

Het prototype heeft de naam “Trashinator 3000”, het doel van de app is om afval uit afbeeldingen te herkennen en aan te geven wat voor soort afval het is en of het te recyclen is. Hierdoor kan de gebruiker snel weten of ze het stuk afval kunnen recyclen of rechtstreeks weg kunnen gooien (in een prullenbak).

Er is gekozen voor deze naam voor zijn speels karakter, zo hoop ik in te spelen op het geringe element van gamification die inherent is aan fotoherkenning. Dit laat een relatief stoffig onderwerp zoals recycling een stuk hipper overkomen.

## Feedback doelgroep

Na het maken van mijn prototype heb ik feedback gevraagd van meerdere mensen die passen binnen de beoogde doelgroep. Mijn feedback groep bestond uit medestudenten, collega’s die werken met Machine Learning en familieleden.

### Concept

De feedback was grotendeels positief. Het concept van het prototype werd goed ontvangen. Voor vele van de testgroep was het nieuws dat er ook op kleine PET flesjes statiegeld geheven wordt. Ook waren zij nog niet op de hoogte dat er volgend jaar ook statiegeld op blikjes geheven gaat worden. Zij onderstreepte dat het handig is als een applicatie dit voor hen in de gaten houd, zelf houden ze dit namelijk niet bij.

### UI

De meest gehoorde feedback was dat de UI simpel en goed te gebruiken, maar wel erg lelijk was. De gebruikers UI van dit prototype bestond uit 1 simpele upload knop op een beige achtergrond met als resultaat een stuk tekst in Times New Roman.

Ik heb deze feedback tot harte genomen en de UI aangepast. De structuur is grotendeels hetzelfde gebleven, aangezien deze functioneel erg positief beoordeeld werd. Maar ik heb de achtergrond, knoppen en output flink aangepast. De oorspronkelijk beige achtergrond heb ik veranderd naar een plaatje van grotendeels groen, versierd met verschillende plaatjes die passen bij recyclen. Dit plaatje heb ik van iStockphoto.com gehaald. De groene kleur past onbewust bij het thema van duurzaamheid.

Op deze groene achtergrond heb ik de knoppen veranderd van een iets lichtere beige naar blauw, zodat deze zich goed afsteken van de achtergrond. Dit versimpeld het doel van de app.

De lettertype van Times New Roman is veranderd naar Gadugi, dit geeft het een stuk vriendelijker uiterlijk.

### Technische werking

Een andere frequente vraag die ik kreeg was wanneer er gebruik gemaakt werd van de camera om een foto te classificeren de classificatie nooit stopt. Dit is erg onhandig als iemand via video een stuk afval probeert te laten checken.

Om deze feedback te verwerken heb ik een knop toegevoegd waarmee het classificeren gedaan wordt waarna het resultaat blijft staan. Deze wijziging werd erg positief ontvangen door de doelgroep.

Afbeelding met tekst, elektronica, scherm, schermafbeelding

Automatisch gegenereerde beschrijving

## Accuracy

De accuracy van de app wordt getoond zodra een afbeelding geclassificeerd wordt. Als de app minder dan 70% vertrouwen heeft in het gekozen resultaat, wordt er getoond dat de app onzeker is over het resultaat. Als de app meer dan 70% vertrouwen heeft, wordt het resultaat getoond samen met hoeveel procent vertrouwen de app heeft in het resultaat.



Ik heb geprobeerd de accuracy van de app te verhogen door het algoritme in plaats van de classifier, regression te laten gebruiken. Dit leverde alleen geen significante accuracy verbetering op. De reden hiervoor is dat de verschillende functies geen rechtstreeks effect hebben op de accuracy, maar meer op de manier waarop het resultaat teruggegeven wordt.

Aangezien de verschillende functies geen toegevoegde waarde hadden aan de accuracy, ben ik gaan experimenteren met verschillende trainingssets. Hierdoor kwam ik erachter dat de grootte van de trainingsset, maar ook de kwaliteit van de trainingsset het grootste invloed hadden op de accuracy van het model. De kwaliteit van foto’s werd hoger wanneer ik deze fotografeerde op een neutrale achtergrond en zelfs nog beter wanneer ik de achtergrond van de foto transparant maak.

Helaas had ik niet genoeg tijd om de achtergrond van alle foto’s van de trainingsset transparant te maken. Hierdoor heb ik voor dit prototype zoveel mogelijk afbeeldingen gebruikt en moest ik accepteren dat de accuracy minder was dan het had kunnen zijn.

In een volgende iteratie zou het bewerken van de trainingsset een grote accuracy verbetering opleveren.

# Techniek

## Algoritme

Voor het algoritme maak ik gebruik van de ML5 Feature Extractor, deze werkt zeer goed voor mijn toepassing, omdat deze de inhoud van een afbeelding herkent en niet de gehele afbeelding. Hierdoor kan de app herkennen wat voor afval object zich in de foto bevindt.

Als alternatief had ik de ML5 Image Classifier kunnen gebruiken, deze herkent net zoals de Feature Extractor afbeeldingen, maar deze focust zich meer op de gehele afbeelding, dan op de inhoud ervan. Tijdens het trainen merkte ik dat meer van de achtergrond in afbeeldingen meegenomen worden waardoor de accuracy van de app zou dalen.

Verder zou ook de TeachableMachine van Google gebruikt worden, deze werkt functioneel hetzelfde als de Feature Extractor en is door de online model creëer site makkelijk op te stellen en te trainen. Het nadeel van de TeachableMachine is dat de code moeilijker is om aan te passen dan de code van ML5 omdat de TeachableMachine meer voor prototypes en kleine applicaties ontworpen is.

## Data

Ik heb naar mijn mening genoeg data kunnen verzamelen om het model goed te trainen, alhoewel er een verschil is in de hoeveelheid data per classificatie. In mijn omgeving kon ik aan veel foto’s komen van blikjes en PET flessen, maar drinkkartonnen waren een stuk lastiger te vinden op straat. Hierdoor sloop er een data voorkeur in (zie hoofdstuk “Data Voorkeur”).

Uiteindelijk heb ik in totaal 168 foto’s kunnen verzamelen. Dit was in principe genoeg voor het algoritme, maar meer had beter geweest.

Na testen met de applicatie kwam ik erachter dat de accuracy van het model sterk verbeterd wanneer de data van de training set een meer neutrale of zelfs transparante achtergrond heeft. Hierdoor zou de AI minder tot geen achtergrond meenemen waardoor het herkennen van objecten beter zou werken.

Als deze app verder uitgewerkt zou worden zou het bewerken van foto’s tot foto’s met een transparante achtergrond hierom belangrijk worden.

## Training

De training van het model vind tijdens het ontwikkelen van de app plaats, hierna wordt het getrainde model in de live versie van de app ingeladen. Er kan achteraf door middel van een aparte admin pagina meer training op het model toegepast worden die vervolgens geüpload kan worden om door de live versie gebruikt te worden.

Er wordt dus tijdens het gebruik van het eindproduct geen extra training toegepast op het model.

## Data Voorkeur

Doordat het model getraind is met meer data voor blikken(101) dan voor flessen(47) en pakken(20), heeft de AI een grotere kans voor het classificeren van afbeeldingen als blikken als het niet zeker is over de inhoud van de afbeelding. Dit wordt ook wel een verborgen voorkeur genoemd.

Verborgen voorkeur is een term die wordt gebruikt om een consistente classificeer neiging aan te duiden. Als een AI getraind wordt met een ongebalanceerde trainingsset, kan er een ‘voorkeur’ ontstaan waardoor er een grotere kans is dat de afbeelding aan de hand van de voorkeur geclassificeerd wordt zonder dat dit ergens in de code genoteerd staat. Hierom wordt het ‘verborgen voorkeur’ genoemd.

## Accuracy

Voor de accuracy van het herkennen van het afval zou de app eigenlijk bijna 100% accuraat moeten zijn, omdat een mens dit zonder de app ook al kan doen. Als de app het afval niet goed herkent, kan de gebruiker niet goed geïnformeerd worden over de recyclebaarheid en mogelijke statiegeld van het afval. Hierom is het dus belangrijk voor het doel van de app dat deze het afval altijd correct identificeren.

Voor een prototype kan er getest worden als de accuracy lager is dan 100%, dit is omdat de meerwaarde van de app niet ligt in het herkennen van het afval, maar van het aangeven of het stuk afval te recyclen is en of er statiegeld op staat.

## Privacy Concerns

Er zijn bij de app geen privacy maatregelen nodig omdat alle data volledig lokaal verwerkt wordt en alle user content wordt alleen in de cache opgeslagen. Hierdoor is er geen opgeslagen of verstuurde variant van de mogelijk gevoelige data van de gebruiker.

Verder is het aanpassen van het getrainde model afgeschermd achter een aparte admin pagina waar een wachtwoord login met MD5 hash encryption op toegepast kan worden om de pagina veilig te stellen

# Conclusie

## Feedback

Toen ik mijn app presenteerde aan een groep kennissen die ook werken met Machine Learning, kreeg ik een aantal sterke feedback punten.

De meest gehoorde feedback was dat de UI er nogal saai uitzag. Hierom heb ik deze verfraaid met een andere achtergrond en de look van de knoppen en gebruikte lettertype aangepast.

Vanuit technisch oogpunt werd aangegeven dat wanneer er gebruik gemaakt werd van de camera om een foto te classificeren de classificatie nooit stopt. Dit is erg onhandig als iemand via video een stuk afval probeert te laten checken. Om dit te fixen heb ik een knop toegevoegd waarmee het classificeren gedaan wordt waarna het resultaat blijft staan.

## Bruikbaar als eindproduct?

De app zou als prototype voor het TLE project gebruikt worden omdat deze volledig functioneel is en als resterende probleem alleen de accuracy van de AI heeft. Dit kan opgelost worden door verder te werken aan de trainingsset waarmee de AI getraind wordt.

met de recente en toekomstige wijzigingen in statiegeld voor drankverpakkingen en het grote nadruk op zwerfafval binnen de gemeente Rotterdam is het concept zelf erg waardevol. Het concept is zeker interessant genoeg om voor het TLE project gebruikt te worden.

Er wordt jammer genoeg niet verder gewerkt aan dit product, omdat ik geen TLE project heb waarvoor deze verder uitgewerkt kan worden en ik er naast school geen tijd voor heb.

## AI waarde

De AI is een groot onderdeel van de basisfunctionaliteit van de app voor het actief herkennen van afval objecten in een afbeelding en het classificeren hiervan, zonder een AI zou het mogelijk zijn om een soortgelijke app te maken, maar deze leidt dan aan een daling in accuracy en een stijging in complexiteit van de code.

De hoofdwaarde van de app is dat deze het opruimen van afval gegamificeerd en tegelijkertijd de gebruiker kan informeren of er statiegeld op het stuk afval staat of niet. Iedereen kan namelijk zelf al herkennen of een stuk afval een blikje is of niet, maar niet iedereen weet dan ook gelijk of er statiegeld op staat of niet.

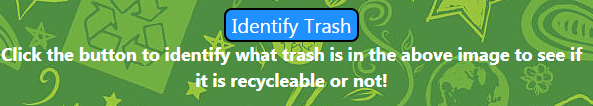
## Eindresultaat

Ik ben tevreden met de functionaliteit van het eindresultaat, met name de mogelijkheid om te kunnen wisselen tussen live beeld via je camera en een afbeelding die je kan uploaden, hier had ik namelijk redelijk wat moeite mee tijdens het programmeren.

De data acquisitie voor de trainingsset ben ik minder tevreden mee, zoals eerder genoemd is er een verborgen voorkeur ontstaan door een onbalans tussen de hoeveelheid data die per classificatie is geleverd. Verder is de AI vaak onzeker over zijn antwoord wat volgens mij komt doordat de afbeeldingen kleurvolle achtergronden hebben die ik niet transparant heb gemaakt.

## Bruikbaarheid

Naar mijn mening is de app redelijk makkelijk door de eindgebruikers te gebruiken, omdat het process simpelweg 1 knop is. Verder staat er op de pagina een kleine uitleg onder het beeld en de knop voor verdere verduidelijking van het doel van de app.



Verder bleek er uit de tests die ik met mijn doelgroep gedaan heb dat de app gemakkelijk te gebruiken is, alhoewel er een probleem was ondervonden met de constante update van het resultaat, waardoor het moeilijk was het resultaat van een stuk afval te zien. Dit is later verholpen door een knop in te bouwen voor het identificeren van het stuk afval. Ook was het snel duidelijk geworden dat de gebruikers geïrriteerd raakten als de app een stuk afval niet correct kon identificeren.

De app kan dus makkelijk door nieuwe gebruikers opgepakt worden, maar de accuracy moet eerst fors verbeterd worden voordat de app uitgebracht kan worden.

# Live app

De live app kan hier gebruikt worden: <https://ConanSpithoven.github.io>

# Screencast

[](https://www.youtube.com/embed/0JRRr8rFXpk?feature=oembed)

Alternatieve link: <https://youtu.be/0JRRr8rFXpk>

# Bronnen

Arends-Tóth, J. (2020). *ICT-gebruik van huishoudens en personen.* Geraadpleegd op 03 juli 2022, van <https://longreads.cbs.nl/ict-kennis-en-economie-2020/ict-gebruik-van-huishoudens-en-personen/#:~:text=In%202019%20maakte%2084%20procent,tegen%2060%20procent%20in%202014>

Gemeente Rotterdam (z.d.). *Zwerfafval*. Geraadpleegd op 03 juli 2022, van <https://www.rotterdam.nl/wonen-leven/zwerfafval/>

Statiegeld Nederland (z.d.). *Statiegeld op blik*. Geraadpleegd op 03 juli 2022, van <https://www.statiegeldnederland.nl/blik/>