



Vlaanderen
is materiaalbewust



AI VOOR ZWERFVUIL MONITORING

HAALBAARHEIDSTUDIE EN PROOF OF CONCEPT

SAMEN MAKEN WE
MORGEN MOOIER

OVAM

WWW.OVAM.BE

OVAM

////////////////////////////////////
////////////////////////////////////
AI voor zwerfvuil monitoring

Haalbaarheidsstudie en Proof of Concept

publicatiedatum / 1.10.2021

////////////////////////////////////
////////////////////////////////////

Dit document werd opgesteld in samenwerking met



DOCUMENTBESCHRIJVING

- | | |
|--|---|
| 1 <i>Titel van publicatie:</i>
AI voor zwerfvuilmonitoring | 2 <i>Verantwoordelijke Uitgever:</i>
OVAM |
| 3 <i>Wettelijk Depot nummer:</i>
D/2021/5024/24 | 4 <i>Trefwoorden:</i>
zwerfvuil, zwerfafval, sluikstort, artificiële intelligentie |
| 5 <i>Samenvatting:</i>
Dit rapport tracht te antwoorden op de vraag of het mogelijk is om met AI zwerfvuilmonitoring te doen. | |
| 6 <i>Aantal bladzijden:</i> 18577 | 7 <i>Aantal tabellen en figuren:</i> 118 |
| 8 <i>Datum publicatie:</i> 2021 | 9 <i>Prijs*:</i> / |
| 10 <i>Begeleidingsgroep en/of auteur:</i> Gauthier Berth, Jonas De Vos, Yen De Clercq, Kimberly Hermans, Yoni Geerinck | 11 <i>Contactpersonen:</i>
Nils Schnitzler, Jan Vanstockem |
| 12 <i>Andere titels over dit onderwerp:</i> / | |

U hebt het recht deze brochure te downloaden, te printen en digitaal te verspreiden. U hebt niet het recht deze aan te passen of voor commerciële doeleinden te gebruiken.

De meeste OVAM-publicaties kunt u raadplegen en/of downloaden op de OVAM-website:
<http://www.ovam.be>

* Prijswijzigingen voorbehouden.

Inhoudstafel

Lijst van figuren	10
Lijst van tabellen	12
Lijst van vergelijkingen	13
Afkortingen- en begrippenlijst	14
Managementsamenvatting	16
1 Inleiding	18
1.1 Algemene context	18
1.2 Situering van de opdracht	18
1.3 Aanpak en onderzoeksvragen	19
1.4 rapport structuur	20
2 Technische Haalbaarheidsstudie	21
2.1 Datacaptatie	21
2.1.1 In-situ/ex-situ	21
2.1.2 Transport modus	24
2.1.3 Media formaat	27
2.1.4 Keuze camera	29
2.1.5 Besluit datacaptatie	30
2.2 Accuraatheid GPS	31
2.2.1 GPS werking en probleemstelling	31
2.2.2 Alternatieven	34
2.2.2.1 GoPro versus iPhone	34
2.2.2.2 ArduSimple SimpleRTK2B	36
2.2.2.3 Europees Galileoproject	36
2.2.3 Besluit	37
2.3 Zwerfvuil detectie en classificatie (AI)	38
2.3.1 Computer vision	38
2.3.1.1 Afbeelding classificatie	38
2.3.1.2 Object detectie	39
2.3.2 Evaluatiemetriecken	40
2.3.3 Algoritmes: Object detectie netwerken	45
2.3.3.1 Specificiteit: Bestaande zwerfvuildetectie modellen	50
2.3.3.2 Zwerfvuil datasets	54
2.3.4 AI obstakels	55
2.3.4.1 De nood aan annotatie	55
2.3.4.2 Overfitting	57
2.3.4.2.1 Trainingstijd	59
2.3.5 3D-modellen	60
2.3.6 Besluit	61
2.4 Zwerfvuiltelling	62

2.4.1	Dubbeltellingen	63
2.4.2	Object lokalisatie	64
2.4.2.1	Fotogrammetrie	66
2.4.2.1.1	Mono-fotogrammetrie	66
2.4.2.1.2	Stereo-fotogrammetrie	68
2.4.2.2	De Richting van de camera en IMU	71
2.4.2.2.1	Camera oriëntatie op basis van GPS sequentiële data	72
2.4.2.2.2	Camera oriëntatie op basis van sensor data	72
2.4.2.3	Bepalen van de geografische locatie van het object	75
2.4.2.3.1	Object lokalisatie conclusies	76
2.4.3	Object Tracering	77
2.4.4	Besluit	79
2.5	Infrastructuur	79
2.5.1	Cloud vs. Non-Cloud aanpak	79
2.5.1.1	Non-Cloudoplossing	80
2.5.1.2	Cloudoplossing	81
2.5.1.3	Voorstel Ordina	82
2.5.2	IaaS, PaaS of SaaS	82
2.5.2.1	IaaS	82
2.5.2.2	PaaS	83
2.5.2.3	SaaS	85
2.5.2.4	Conclusie Cloud type	85
2.5.3	Azure services voor POC en uitrolfase	85
2.5.3.1	Dataopslag	86
2.5.3.2	Data transformaties	86
2.5.3.3	Model training, testing en beheer	87
2.5.3.4	Conclusie datacaptatie en machine learning	88
2.6	Conclusie technische haalbaarheid en oplossing	89
2.6.1	Functionele flowchart	89
2.6.1.1	Verzameling en verwerking data	91
2.6.1.2	Annoteren data	91
2.6.1.3	Bouwen, trainen en testen model	91
2.6.1.4	Voorspellen nieuwe data	93
2.6.1.5	Bouw datamodel	93
2.6.1.6	Bouw rapport en dashboard	94
2.6.2	Architecturale mapping	94
2.6.2.1	Verzameling en verwerking data	95
2.6.2.2	Annoteren data	96
2.6.2.3	Bouwen, trainen en testen model	96
2.6.2.4	Voorspellen nieuwe data	97
2.6.2.5	Versiebeheer van het model	97
3	Juridische Haalbaarheid	98
4	POC	99

4.1	Positie camera en beeldkwaliteit	99
4.1.1	Beeldkwaliteit GoPro	99
4.1.1.1	Voertuig – Hoog	100
4.1.1.2	Voertuig – Laag	101
4.1.1.3	Bodycam	101
4.1.2	Beeldkwaliteit ZED	102
4.1.3	Foto's versus frames	103
4.1.4	Burst modus	104
4.1.5	Conclusie	105
4.2	AI – object detectie	106
4.2.1	Het capteren van data	106
4.2.2	Het annoteren van data	107
4.2.3	Trainen van de object detectie modellen	109
4.2.3.1	Resultaten blikjes model	112
4.2.3.2	Resultaten sigarettendoosjes model	116
4.2.4	Extrapolatie trainingsresultaten	119
4.2.5	Inference op video's en burst	122
4.2.6	Conclusie	123
4.3	Zwerfvuiltelling	123
4.3.1	Object lokalisatie	123
4.3.1.1	De ruimtelijke oriëntatie van de camera	123
4.3.1.2	De relatieve locatiebepaling van gedetecteerde objecten	126
4.3.1.3	De absolute locatiebepaling van objecten	128
4.3.2	Object tracering	130
5	Financiële studie	133
5.1	Set-up productie	134
5.1.1	Infrastructuur	134
5.1.1.1	Set-up kost	134
5.1.1.2	Opslagkost	135
5.1.2		136
5.1.3	Hardware	137
5.1.4	Processen	137
5.1.4.1	Creatie van een proces	137
5.1.4.2	Berekeningskost	139
5.1.5	Onderzoek en opzet	139
5.1.6	Annotatie	140
5.1.7	Modelontwikkeling	141
5.1.7.1	Beelden opnemen	142
5.1.7.2	Beelden annoteren	143
5.1.7.3	Model trainen	144
5.1.7.4	Model testing en <i>deployment</i>	144
5.2	Productie	145
5.2.1	Infrastructuur	145

5.2.2	Processen	146
5.2.3	Modelontwikkeling	147
5.2.3.1	Beelden opnemen	147
5.2.3.2	Beelden annoteren	148
5.2.3.3	Model training en testing	148
5.2.4	Technische ondersteuning	149
5.3	Overzicht kosten	149
5.4	risico's	151
6	Besluit	153
7	Bijlagen	156
7.1	Specifieke onderzoeksvragen	156
7.1.1	Technisch	156
7.1.2	Financieel	156
7.1.3	Juridisch	157
7.2	Technische uitdiepingen	157
7.2.1	Praktische Stereo-fotogrammetrische problemen	157
7.2.1.1	Overlap	157
7.2.1.2	Ruimtelijke oriëntatie van de camera's	159
7.2.1.3	Bepaling van de projecties van het te zoeken punt in het frame paar	160
7.3	Resultaten model	162
7.3.1	Blikjes model	162
7.3.2	Sigaretendoosjes model	166
7.4	Gegevensbeschermingseffectbeoordeling (GBEB)	172
7.4.1	Project overzicht	172
7.4.2	Identificeer de noodzaak van een DPIA	172
7.4.2.1	Doelstellingen en voordelen van het project	172
7.4.2.2	Voorafgaande screening	172
7.4.2.3	Conclusie	175
7.4.3	Beschrijf de verwerking	175
7.4.4	Overwegen om interne of externe belanghebbenden te raadplegen	176
7.4.4.1	Interne belanghebbenden	176
7.4.4.2	Externe stakeholders	176
7.4.5	Beoordeling van noodzaak en proportionaliteit	177
7.4.6	Identificeren en beoordelen van risico's	177
7.4.6.1	Privacy Risico voor personen	177
7.4.6.2	Wettelijke naleving Risico's	178
7.4.7	Identificeren en evalueren van maatregelen om het risico te verminderen	178
7.4.8	Ondertekenen en vastleggen van DPIA-resultaten	182
7.4.8.1	[Samenvatting DPO advies]	182
7.4.8.2	Samenvatting van de risico's voor de privacy en goedgekeurde oplossingen	182
7.4.8.3	Belanghebbende consultatie	182
7.4.8.4	ICO consultatie	182

7.4.8.5	Goedkeuring en aftekenen	182
7.4.9	Integratie van de DPIA-resultaten in het algemene projectplan	183
7.4.9.1	High-level acties	183
7.4.9.2	Specifieke acties	183

<i>bibliografie</i>		185
----------------------------	--	------------

LIJST VAN FIGUREN

Figuur 1: Onderdelen van een GPS signaal	31
Figuur 2: GPS tijdsverschuiving	32
Figuur 3: Afstand van een periode de modulatie van het GPS signaal	33
Figuur 4: Afstand van een periode in de draaggolf van het GPS signaal	33
Figuur 5:Wandeling iPhone vs. GoPro	34
Figuur 6 Wandeling - Open omgeving met bebouwing	35
Figuur 7 Wandeling - Dichtbebouwd	35
Figuur 8: Voorbeeld van classificatie, één typerend object op de foto	38
Figuur 9: Voorbeeld Object detectie, meerdere typerende objecten op foto	39
Figuur 10: Confusion matrix	41
Figuur 11: IoU verduidelijking	42
Figuur 12: Grafisch voorbeeld van de berekening van de mAP-metriek	43
Figuur 13: Grafische voorstelling Precision/Recall	45
Figuur 14: COCO-benchmark resultaten	47
Figuur 15: PASCAL VOC benchmark resultaten	48
Figuur 16: Links een foto uit de COCO-benchmark en rechts een foto uit een proefopstelling	48
Figuur 17: Test afbeeldingen gestuurd door het TACO_10 model	51
Figuur 18: Resultaten onderzoek Chung et al. (2018)	52
Figuur 19: Test afbeeldingen gestuurd door het Let's do it foundation model	54
Figuur 20: Eerste stap unsupervised object detectie	55
Figuur 21: Tweede stap unsupervised object detectie	56
Figuur 22: Laatste stap unsupervised object detectie	56
Figuur 23: Een voorbeeld van overfitting	59
Figuur 24: Resultaten trainingstijden op kangoeroe dataset	60
Figuur 25: Camera positie	64
Figuur 26: Eén frame capteert meerdere cellen	65
Figuur 27: Mono-fotogrammetrische rectificatie	67
Figuur 28: Stereoscopie	68
Figuur 29: Parallax voorbeeld	69
Figuur 30: Parallax mathematisch	69
Figuur 31: Voorwaartse insnijding	70
Figuur 32: Camera oriëntatie probleem	71
Figuur 33: Acceleratiemeter in stilstand	73
Figuur 34: Acceleratiemeter in beweging	73
Figuur 35: Gyrocoop	73
Figuur 36: Lokaal referentiesysteem van camera	74
Figuur 37: Rotatiehoek voor oriëntatie camera	75
Figuur 38: Rotatie van een assenstelsel	75
Figuur 39: Lambertcoördinatensysteem	76
Figuur 40: Het concept van object tracering	77
Figuur 41: Datamigraties in de POC	88
Figuur 42: Functionele flowchart	90

Figuur 43: Architecturale mapping	95
Figuur 44: Methoden van objectieve vergelijkbaarheid tussen modellen	97
Figuur 45: Constructie gelijkend op een vuilniswagen	100
Figuur 46: Vergelijking voertuig – hoog 4K (links) vs. 2K (rechts) kwaliteit.	100
Figuur 47: Ingezoomde vergelijking voertuig – hoog 4K (links) vs. 2K (rechts) kwaliteit.....	101
Figuur 49: Vergelijking bodycam 4K (links) vs. 2K (rechts) kwaliteit.....	101
Figuur 50: Ingezoomde vergelijking bodycam 4K (links) vs. 2K (rechts) kwaliteit.	102
Figuur 52: Originele frame (links) en ingezoomde frame (rechts) van een ZED-video	102
Figuur 53: Vergelijking 4K-frame (links) vs. 4K-foto (rechts)	103
Figuur 54: Verschil 4K foto (links) vs. 2K foto (rechts)	104
Figuur 55: vergelijking frame uit 2K-video (links) vs. foto uit 2K-burst (rechts)	104
Figuur 56: Ingezoomde vergelijking frame uit 2K-video (links) vs. foto uit 2K-burst (rechts)	104
Figuur 57: Vervolgtraject camera's	105
Figuur 58: De verschillende ondergronden waarop zwerfvuil werd gecapteerd	106
Figuur 59: Distributie zwerfvuil bounding boxes	108
Figuur 60: Vergelijking Let's do it foundation model (links) vs. YOLO v5 model (rechts)	109
Figuur 61: Trainingsopzet	112
Figuur 62: Resultaten blikjes model	112
Figuur 63: De invloed van de mAP_0,5:0.95 metriek	113
Figuur 64: Fouten blikjes model	115
Figuur 65: Projectie mAP blikjes model	116
Figuur 66: Resultaten sigarettendoosjes model	116
Figuur 67: Fouten sigarettendoosjes model	118
Figuur 66: Projectie mAP sigarettendoosjes model.....	119
Figuur 69: Inference op een frame uit een video (links) en een foto uit burst (rechts)	122
Figuur 70: Kalibratie van de magnetometer	124
Figuur 71: Camera oriëntatie ten opzichte van het noorden	126
Figuur 72: Resultaat testopstelling relatieve locatiebepaling.....	127
Figuur 73: Absolute locatiebepaling	129
Figuur 74: Output SimpleSort algoritme.....	131
Figuur 75: De geïntroduceerde ruis door onverwachte bewegingen	132
Figuur 75: Twee camera's: frame paren	158
Figuur 76: Tijdsaspect frame paren bij één camera.....	158
Figuur 77: Eén camera: frame paren	159
Figuur 78: Ruimtelijke oriëntatie van de camera's	159
Figuur 79: Gemeenschappelijke punten bepalen in frame paar	160
Figuur 80: Probleem van object matching bij één camera	161

LIJST VAN TABELLEN

Tabel 1: In-situ/ex-situ	23
Tabel 2: Voor- en nadelen van de verschillende modi	27
Tabel 3: Fout-oorzaken van GPS signalen	32
Tabel 4: Maximale afwijking GPS signaal per gebied	35
Tabel 5: Berekening mAP	44
Tabel 6: Geannoteerde open-source datasets	54
Tabel 7: Voor- en nadelen van een non-Cloud of on-premise oplossing	81
Tabel 8: Voor- en nadelen van een Cloudoplossing	82
Tabel 9: Voor- en nadelen van IaaS t.o.v. een non-Cloudoplossing	83
Tabel 10: Voor- en nadelen IaaS t.o.v. andere Cloud computing types	83
Tabel 11: Voor- en nadelen PaaS t.o.v. IaaS	85
Tabel 12: Voor- en nadelen manueel en automatisch trainen	96
Tabel 13: Extrapolatie moeilijkheid 24 zwerfvuilfracties	120
Tabel 14: Aantal trainingsdata en trainingstijd per moeilijkheidsniveau	121
Tabel 15: Kostenberekening set-up productie infrastructuur (exclusief btw)	134
Tabel 16: Kostenberekening set-up - productie infrastructuur – detailberekening (exclusief BTW)	135
Tabel 17: Kostenberekening set-up - productie infrastructuur opslag (exclusief BTW)	136
Tabel 18: Kostenberekening set-up – hardware (exclusief BTW)	137
Tabel 19: Kostenberekening set-up – productieprocessen (exclusief BTW)	137
Tabel 20: Kostenberekening set-up – productieprocessen creatie (exclusief BTW)	138
Tabel 21: Kostenberekening set-up – productie processen berekeningskost (exclusief BTW)	139
Tabel 22: Kostenberekening set-up – productie onderzoek & opzet (exclusief BTW)	140
Tabel 23: Kostenberekening set-up – productie opzet annotatie-infrastructuur (exclusief BTW)	141
Tabel 24: Kostenberekening set-up – productie model development (exclusief BTW)	141
Tabel 25: Kostenberekening set-up – productie model development beeldopnames (exclusief BTW)	143
Tabel 26: Kostenberekening set-up – productie model development beelden annotatie (exclusief BTW)	144
Tabel 27: Kostenberekening set-up productie model development training (exclusief BTW)	144
Tabel 28: Kostenberekening set-up productie – model development testing en deployment (exclusief BTW)	145
Tabel 29: Kostenberekening productie – infrastructuur	146
Tabel 30: Kostenberekening productie – processen	147
Tabel 31: Kostenberekening productie – model development	147
Tabel 32: Kostenberekening productie – model development beelden opnemen	148
Tabel 33: Kostenberekening productie – model development beelden annoteren	148
Tabel 34: Kostenberekening productie – model development training en testing	149
Tabel 35: Totale kosten – minimaal (exclusief BTW)	150
Tabel 36: Totale kosten – maximaal (exclusief BTW)	150
Tabel 37: Projectie van de geaccumuleerde kosten van meetcampagnes	151
Tabel 38: Risicotabel	151

LIJST VAN VERGELIJKINGEN

Vergelijking 1: Precision	41
Vergelijking 2: Recall	42
Vergelijking 3: IoU	42
Vergelijking 4: AP	45
Vergelijking 5: Schaalfactoren berekening	67
Vergelijking 6: Perspectief berekening	67
Vergelijking 7: Pixel coördinaat berekening	67
Vergelijking 8: berekening groeicoëfficiënt hoeveelheid trainingsdata	120
Vergelijking 9: Berekening trainingstijd	121
Vergelijking 10: Omzetting magnetische inductie naar een hoek in graden	124
Vergelijking 11: Afstandsberekening in drie dimensies	128
Vergelijking 12: Rotatie- en translatie formules	129
Vergelijking 13: Berekening opnametijd validatievideo	142
Vergelijking 14: Berekening totaal aantal te annoteren zwerfvuilobjecten	144

AFKORTINGEN- EN BEGRIPPENLIJST

ADLS	AZURE DATA LAKE STORAGE
AI	ARTIFICIËLE INTELLIGENTIE
AML	AZURE MACHINE LEARNING
ANNOTEREN	HET MANUEEL PROCES WAARBIJ TRAININGSDATA MOET VOORZIEN WORDEN VAN LABELS. DEZE LABELS ZIJN NODIG OM HET MODEL TE TRAINEN
AP	AVERAGE PRECISION
API	EEN API (APPLICATION PROGRAMMING INTERFACE) IS EEN SOFTWARE-INTERFACE DIE HET MOGELIJK MAAKT DAT TWEE APPLICATIES MET ELKAAR KUNNEN COMMUNICEREN
AVG	ALGEMENE VERORDENING GEGEVENSBESCHERMING
BCDC	BELGIAN CIVILIAN DRONE COUNCIL
BOUNDING BOX	EEN RECHTHOEK DIE AANTOONT WAAR HET TE DETECTEREN OBJECT ZICH BEVINDT
CLOUD	TECHNIEK WAARMEE VIA EEN NETWERK OP AANVRAAG HARDWARE, SOFTWARE EN GEGEVENS TER BESCHIKKING GESTELD KAN WORDEN
COMPUTER VISION	HET DOMEIN VAN ARTIFICIËLE INTELLIGENTIE DAT ZICH BEZIGHOUDT MET AFBEELDINGEN EN VIDEO'S
CONFUSION MATRIX	EEN MATRIX BESTAANDE UIT HET AANTAL FP, TP, FN EN TN
DABM	DECREET ALGEMENE BEPALINGEN INZAKE MILIEUBELEID
DEPLOYMENT	HET DEPLOYEN VAN SOFTWARE ZIJN AL DE ACTIVITEITEN DIE NODIG ZIJN OM EEN SYSTEEM KLAAR TE MAKEN VOOR GEBRUIK
DGPS	DIFFERENTIAL GPS
DISCRIMINATOR	EEN MACHINE LEARNING MODEL DAT ALS TAAK HEEFT VERSCHILLEN TUSSEN TWEE DATASTROMEN TE IDENTIFICEREN
ESA	EUROPEAN SPACE AGENCY
FN	FALSE NEGATIVE
FP	FALSE POSITIVE
FPS	FRAMES PER SECOND
FRAMES	FRAMES ZIJN SNAPSHOTS UIT VIDEO'S
GBEB	GEGEVENSBESCHERMINGSEFFECTBEOORDELING
GENERALISATIE	EEN EIGENSCHAP VAN EEN AI MODEL DIE OPGEDANE KENNIS IN DE TRAININGSFASE KAN TOEPASSEN OP ONGEZIENE DATA
IAAS	EEN VORM VAN CLOUD COMPUTING. DE CLOUD PROVIDER ZORGT ENKEL VOOR DE INFRASTRUCTUUR EN LAAT DE GEBRUIKER DE KEUZES MAKEN OMTRENT HET AANSCHAFFEN, INSTALLEREN, CONFIGUREREN EN BEHEREN VAN DE SOFTWARE DIE DE INFRASTRUCTUUR GEBRUIKT
IMU	INERTIAL MEASUREMENT UNIT
MAP	MEAN AVERAGE PRECISION
NGI	NATIONAAL GEOGRAFISCH INSTITUUT
OBJECT DETECTION	HET DOMEIN BINNEN COMPUTER VISION DAT ZICH BEZIGHOUDT MET HET DETECTEREN VAN OBJECTEN
ON-PREMISE	IN TEGENSTELLING TOT DE CLOUD WORDT ON-PREMISE SOFTWARE LOKAAL GEÏNSTALLEERD EN GEDRAAID OP DE COMPUTERS VAN DE PERSOON OF ORGANISATIE DIE DE SOFTWARE GEBRUIKT

OVAM	OPENBARE VLAAMSE AFVALSTOFFENMAATSCHAPPIJ
PAAS	EEN VORM VAN CLOUD COMPUTING. HET CONFIGUREREN, BEHEREN EN MONITOREN VAN DE ONDERLIGGENDE INFRASTRUCTUUR, SOFTWARE UPDATES, BEDRIJFSSYSTEMEN, ETC. WORDEN OPGENOMEN DOOR DE CLOUD PROVIDER. DE GEBRUIKER IS ZELF ENKEL VERANTWOORDELIJK VOOR DE APPLICATIE DIE GEHOST WORDT OP DIT PLATFORM
POC	PROOF OF CONCEPT
PRE-TRAINED MODEL	EEN MODEL DAT REEDS GETRAIND IS DOOR EEN DERDE PARTIJ
SAAS	EEN VORM VAN CLOUD COMPUTING. DIT ZIJN VOLLEDIGE, KANT-EN-KLARE SOFTWARETOEPASSINGEN DIE VIA HET INTERNET GELEVERD WORDEN. HIERBIJ WORDT ALLES UITBESTEED AAN DE CLOUD PROVIDER
SDK	SOFTWARE DEVELOPMENT KIT
SUPERVISED LEARNING	EEN MACHINE LEARNING METHODE WAAR GEANNOTEEERDE DATA VEREIST IS OM TE TRAINEN
TN	TRUE NEGATIVE
TOM	TECHNISCHE EN ORGANISATORISCHE MAATREGEL
TP	TRUE POSITIVE
TRANSFER LEARNING	HET GEBRUIK VAN EERDER OPGEDANE KENNIS IN ARTIFICIËLE INTELLIGENTIE
TUNING	HET AANPASSEN VAN EEN MODEL ZODOENDE BETERE RESULTATEN TE BEKOMEN
VVSG	VERENIGING VAN VLAAMSE STEDEN EN GEMEENTEN

MANAGEMENTSAMENVATTING

Na een uitvoerige voorstudie voor de fractietelling van het zwerfvuil, wil de OVAM de haalbaarheid van AI-beeldherkenning als robuust monitoringsinstrument voor zwerfvuil onderzoeken. Door middel van deze studie kan de OVAM kosten en baten afwegen en een gefundeerde beslissing maken voor een eventueel vervolgtraject. De studie werd opgedeeld in drie luiken: de haalbaarheidsstudie (technisch en juridisch), een Proof of Concept (POC) en een financiële studie.

In het technische luik van de haalbaarheidsstudie werden de vereiste componenten geïdentificeerd en geanalyseerd. Hoewel de technologieën achter het beoogd systeem werken en op basis van de POC verwacht wordt dat men daadwerkelijk een AI-systeem voor zwerfvuilmonitoring kan bouwen, zijn er hardware-technische en operationele uitdagingen die een robuust systeem momenteel op de helling zet. Het eerste hardware-technisch probleem betreft de video kwaliteit, dewelke scherpte mist ten opzichte van foto's. Het maken van video's is essentieel in het bouwen van een automatische oplossing. Ordina verwacht dat deze uitdaging te verhelpen valt, mits extra onderzoek naar camera's die zeer snel kwalitatieve foto's na elkaar kunnen maken (*'bursting'*). De tweede, grootste uitdaging is de huidige precisie van moderne GPS-systemen. In dit opzicht werden twee mogelijke oplossingen gedefinieerd. De eerste oplossing legt zich neer bij een ontoereikende precisie van twee tot dertig meter en accepteert een minder robuust systeem waarin de link tussen een meting en de betreffende type-omgeving niet altijd juist gelegd zal kunnen worden. De tweede oplossing vertrekt van de assumptie dat een accurate locatiebepaling in de toekomst realiseerbaar is door het Galileo project, waardoor een robuust systeem kan worden ontwikkeld dat elk gevonden zwerfvuilobject linkt aan een geografische plaats. Galileo is een Europees satellietensysteem met een gelijkaardige functionaliteit als GPS. Beide uitdagingen vereisen verder onderzoek.

Naast de hardware-technische uitdagingen, dienen twee operationele uitdagingen vermeld te worden. Het eerste betreft de effectiviteit van een camera op een wagen, dewelke de gewenste captatievorm is. Objecten die het zicht versperren kunnen met deze captatievorm namelijk niet vermeden worden, wat leidt tot gemist zwerfvuil. Om dit probleem te verhelpen werd een combinatiesysteem voorgesteld dat gebruik maakt van de verschillende grondmodi: auto, fiets en bodycam. Dergelijk systeem verhoogt de complexiteit (bv. integratie van de afzonderlijk werkende modi). De laatste operationele uitdaging betreft de gedefinieerde "rest" groep fracties van het AI-systeem. Deze fracties, zoals bv. een stuk papier of karton dat niet meer onder een specifieke fractie valt of onherkenbaar is geworden, zullen resulteren in onbetrouwbare resultaten zolang geen duidelijkere afbakening wordt gedefinieerd.

Verder werd aangetoond dat bepaalde fracties automatisch gedetecteerd en geteld kunnen worden met behulp van AI. Meer bepaald werden twee AI modellen ontwikkeld welke succesvol sigarettendoosjes en blikjes kunnen

identificeren. De inspanningen werden bijgehouden en geëxtrapoleerd naar de overige fracties. Bijkomend werd een systeem ontwikkeld dat gedetecteerde objecten kan koppelen aan een geografische coördinaat. Mits extra inspanningen kunnen deze tot nu toe afzonderlijk werkende onderdelen gecombineerd worden tot een productiewaardige oplossing.

In het juridisch luik werden geen significante uitdagingen gevonden, zolang men niet afstapt van de gemaakte adviezen. Door bijvoorbeeld te kiezen voor lucht-modi, zal aan een strikt wettelijk kader voldoen moeten worden en contact opgenomen moeten worden met de Cel Drones van de FOD Mobiliteit. De doelstelling van dit systeem is zwerfvuilmonitoring en -telling en heeft dus niet de verwerking van persoonsgegevens tot doel. Daarnaast zou de OVAM voor de eventuele verwerking van persoonsgegevens zich moeten kunnen beroepen op artikel 6.1.e AVG (Algemene Verordening Gegevensbescherming). Er wordt wel geacht van de OVAM dat men conform de stand van de techniek poogt de verwerking van persoonsgegevens te minimaliseren, bijvoorbeeld door gebruik te maken van *blurring* van gezichten van eventuele betrokkenen-passanten.

De financiële studie brengt alle resultaten samen en trachtte een antwoord te bieden op de vraag: “Is het economisch verantwoord om te investeren in een AI systeem voor zwerfvuilmonitoring?”. Op het eerste zicht lijkt het economisch verantwoord. Hoewel de ontwikkelingskosten hoog zijn, zijn het de lage recurrente kosten die een duidelijk financieel voordeel opleveren in de toekomst. Er wordt, in het meest negatieve geval, ingeschat dat vanaf de vierde uitoefening van het beoogd systeem de totale kost kleiner is dan wanneer het huidige manuele systeem zou gehandhaafd blijven. Hierbij is het belangrijk te melden dat dit uitgaat van (a) het Galileo systeem, wat de precisie van de geolocalisatie normaal gezien verhoogt, (b) het een gebruik van een videocamera met *bursting*, maar vooral (c) dat men de rest fracties niet langer hanteert of fijner definieert en (d) dat men de afwisseling tussen de verschillende grond modi minimaliseert.

Concluderend kan gesteld worden dat het beoogde systeem financieel interessant is, maar door operationele en hardware-technische uitdagingen op dit moment nog niet volledig haalbaar lijkt. Ordina raadt dan ook aan de aangehaalde uitdagingen eerst uit te klaren, zoals het herdefiniëren van de “rest” fracties en te wachten met een verdere ontwikkeling van het systeem tot een preciezer locatiebepaling mechanisme voorhanden is of extra onderzoek in dit veld zijn uitgevoerd.

1 INLEIDING

1.1 ALGEMENE CONTEXT

Het programma Mooimakers heeft als doel een drastische vermindering van de hoeveelheid zwerfvuil in Vlaanderen. Dit programma wordt aangestuurd door partijen vanuit de overheid (Openbare Vlaamse Afvalstoffenmaatschappij (OVAM), Vereniging van Vlaamse Steden en Gemeenten (VVSG)) en bedrijven FostPlus, Fevia en Comeos. Hiervan heeft de OVAM onder andere de verantwoordelijkheid om zwerfvuil in Vlaanderen continu te monitoren. Dit zodat de evolutie over tijd in kaart gebracht kan worden en beleidsevaluatie met het definiëren van concrete acties mogelijk wordt. In 2019 werd het '*vooronderzoek en steekproeftrekking voor fractietelling zwerfvuil*' (hierna: '*vooronderzoek fractietelling*') uitgevoerd dat zich focuste op hoe men zwerfvuil aan de hand van fracties kan monitoren. Het huidige onderzoek spitst zich daarentegen toe op de mogelijkheid tot automatisatie van de monitoring, meer bepaald door het gebruik van Artificiële Intelligentie (AI).

Het hoofddoel achter de opdracht is dus een degelijke monitoring bekomen wat zal leiden tot een beter beleid met gerichte acties. Ter illustratie: het is denkbaar dat in publieke zones zoals een bushalte meer zwerfvuil waargenomen wordt. In dat geval zouden extra vuilbakken in dit type zone geplaatst kunnen worden teneinde het zwerfvuil daar te doen afnemen. In een volgende meting kan gekeken worden naar de doeltreffendheid van dergelijke maatregelen.

1.2 SITUERING VAN DE OPDRACHT

Bij aanvang van de opdracht maakt de OVAM gebruik van manuele tellingen. Deze manuele tellingen worden gemiddeld vierjaarlijks uitgevoerd en beslaan alle vier seizoenen. Omdat het onmogelijk is om overal exacte tellingen uit te voeren, heeft de OVAM het Vlaamse grondgebied onderverdeeld in 16 categorieën of type-omgevingen. Deze 16 type-omgevingen worden op de website van de OVAM vermeld¹. Voor elk van deze categorieën worden willekeurig een aantal cellen van 10 bij 10 meter bepaald waarin metingen worden uitgevoerd. Extrapolatie van de gevonden meetresultaten geeft een totaalbeeld voor het Vlaamse openbaar domein. Voor details wordt verwezen naar het '*vooronderzoek fractietelling*'.

Een manuele telling komt tot stand in twee fasen. Een eerste fase waarbij terreinmeters met een vuilniszak de straat op gaan. Het gevonden zwerfvuil wordt per te meten cel in een vuilniszak van de respectievelijke cel gedaan. Op die manier wordt het zwerfvuil per cel van elkaar gescheiden. Deze vuilzakken worden vervolgens naar een centrale locatie gebracht waar een manuele telling, weging en volumebepaling van de verzamelde zwerfvuilitems gebeurt.

¹ <https://services.ovam.be/ovam-geoloketten/#/zwerfvuil?x=140410&y=198535&z=10>

Daar dit gaat om een zeer intensief proces, wordt in deze opdracht onderzoek gedaan naar de haalbaarheid van artificiële intelligentie (AI) voor zwerfvuilmonitoring. De veronderstelling is dat een beeldherkenningssysteem het zwerfvuil telt om inzicht te verwerven in de hoeveelheid zwerfvuil (aantal stuks), de samenstelling van het zwerfvuil en de vervuilingssnelheid van een bepaalde omgeving. Dit AI-systeem kan worden ingezet als aanvulling of vervanging van de huidige manuele tellingen. De kernvraag van deze opdracht luidt bijgevolg: *kan AI-beeldherkenning worden ingezet als **robuust monitoringinstrument** voor het zwerfvuil in Vlaanderen?*

Om een antwoord te formuleren op de haalbaarheid van een AI-oplossing formuleerde de OVAM een aantal onderzoeksvragen. Deze onderzoeksvragen zijn terug te vinden in paragraaf 7.1.

Naast de uitkomst van het onderzoek naar haalbaarheid, werd ook een Proof of Concept (POC) gerealiseerd die in de praktijk aantoont wat het resultaat van een AI-systeem kan zijn. Concreet worden een aantal bouwblokken uitgewerkt ter demonstratie en als basis voor een eventuele vervolgoopdracht. Het is niet de bedoeling dat alle bouwblokken reeds productiewaardig ontwikkeld worden.

1.3 AANPAK EN ONDERZOEKSVRAGEN

De haalbaarheid van een AI-systeem wordt onderzocht op basis van reeds gedefinieerde onderzoeksvragen (paragraaf 7.1) van de OVAM en omvatten drie grote invalshoeken, meer bepaald de technische, financiële en juridische haalbaarheid. Enkel door elk van deze aspecten grondig te doorzoeken kan de OVAM bepalen of een AI-systeem voor automatische telling van zwerfvuil een economisch verantwoorde keuze is.

In het technische luik zal het systeem en de ICT-kant onderzocht worden. Hier wordt onder andere gekeken naar wat de IT-infrastructuur van het AI-systeem kan zijn en op basis van functionele en technische vereisten een advies geformuleerd. Concreet zal de hardware, zoals camerakeuze, wijze van data-inzameling, dataopslag infrastructuur en de mogelijkheden van AI aan bod komen.

In het Juridisch deel worden mogelijke juridische consequenties en beperkingen besproken. Er wordt gekeken naar welk type agentschap de OVAM is, naar haar taken en bevoegdheden en de mate waarin dit in lijn ligt met het gewenste AI-systeem ter automatisatie van de fractietellingen. Tevens wordt nagegaan of de opgestelde vereisten vanuit de OVAM voor het AI-systeem voldoen aan nieuwe richtlijnen van de EU. Tot slot wordt gekeken naar de persoonlijke levenssfeer en mate waarin het te bouwen systeem voldoet aan de gegevensbeschermingsregelgeving.

Indien de haalbaarheidsstudie aantoont dat met voldoende zekerheid een systeem gebruikmakende van AI kan gebouwd worden, worden hiertegenover financiële inschattingen gezet. Deze worden gebundeld in het financiële luik. Alle geadviseerde keuzes brengen een bepaalde kost met zich mee, die de OVAM in staat stelt een finale keuze te formuleren en al dan niet een AI-toepassing te laten bouwen voor de automatische detectie van zwerfvuil.

1.4 RAPPORT STRUCTUUR

Het rapport bestaat uit drie grote onderdelen. De haalbaarheidsstudie, in eerste instantie technisch en juridisch, gaat op basis van kennis en ervaring uit de verschillende domeinen en reeds bestaande systemen een uitspraak doen over het gewenste ontwerp en de verwachte prestaties bij het toepassen van AI voor de detectie van zwerfvuil. In dit onderdeel worden de drie verschillende perspectieven en bijhorende onderzoeksvragen in detail uitgewerkt. Het tweede onderdeel omvat de uitgevoerde POC. Hier wordt de haalbaarheid en het verrichte onderzoek met de praktijk getoetst. Het doel van de POC is louter valideren en uitspraken uit de haalbaarheidsstudie staven en bewijzen. Het gebouwde systeem is geen productiewaardig systeem, maar is wel in staat aan te tonen hoe een algoritme zou werken en wat de mate van accuraatheid momenteel is en kan worden.

Het rapport doet uitschijnen dat men eerst de studie uitvoerde en vervolgens de POC. In realiteit werden tijdens het bouwen en testen van de POC regelmatig nieuwe inzichten verworven. Deze zijn uiteraard mee toegevoegd aan de technische en juridische studie en waar nodig werden conclusies aangepast.

Op basis van deze technische en juridische haalbaarheid en de resultaten van de POC werd de financiële studie uitgeschreven en kon in een finaal hoofdstuk tot een eindconclusie gekomen worden.

2 TECHNISCHE HAALBAARHEIDSSSTUDIE

De Technische studie omvat alle onderdelen die noodzakelijk zijn voor het realiseren van een systeem voor automatische zwerfvuildetectie. Volgende onderdelen werden gedefinieerd en onderzocht:

- In eerste instantie moet data opgevangen, opgeslagen en verwerkt kunnen worden. Hier zijn verschillend onderdelen die onderzocht dienen te worden zoals bijvoorbeeld de keuze van camera en of men videobeelden of foto's verkiest. Dit wordt bekeken in het onderdeel **datacaptatie**.
- Om de locatie van het met de camera gecapteerde zwerfvuil te achterhalen, is een locatiebepaling systeem nodig. De meest gekende is de Global Positioning System (GPS). Deze is echter inaccuraat, waardoor alternatieven onderzocht moeten worden. Deze problematiek wordt besproken in het onderdeel **accuraatheid GPS**.
- Vervolgens wil de OVAM artificiële intelligentie toepassen voor het **detecteren en classificeren van zwerfvuil**. Hier wordt gekeken naar soorten algoritmes en reeds bestaande zwerfvuil detectie modellen alsook mogelijke valkuilen voor dit project.
- Het detecteren van zwerfvuil alleen is niet genoeg, het dient ook geteld te worden. In het luik **zwerfvuil telling** wordt onderzocht wat hiervoor de meest robuuste manier is.
- In het onderdeel **infrastructuur** wordt nagegaan wat de huidige IT-infrastructuur van de OVAM is en of deze al dan niet voldoet aan de vereisten van een AI-systeem. Indien niet, wordt ook gekeken naar alternatieve oplossingen, rekening houdend met de wensen van de OVAM.

In dit hoofdstuk worden de mogelijkheden samengevat en keuzes gestaafd met de daarbij horende voor- en nadelen. De grootste risico's en uitdagingen worden telkens toegelicht.

2.1 DATAAPTATIE

Om AI succesvol toe te passen, is er voldoende data nodig waarvan gekend is welke zwerfvuil fracties zich waar bevinden in de data. Dit is noodzakelijk om een algoritme of model te leren detecteren. Er moet dus gekeken worden naar welke data nodig is en waar men deze data kan verzamelen. Het staat buiten kijf dat zonder data de haalbaarheid van een AI-systeem negatief is.

Welke data en hoe die gecapteerd wordt zal tevens als basis dienen voor het verder uittekenen van het proces en de infrastructuur.

2.1.1 In-situ/ex-situ

De eerste keuze die dient gemaakt te worden is op welke manier men het zwerfvuil wil monitoren en waar het systeem toegepast zal worden ten opzichte van de huidige manier van monitoring.

In-situ datacaptatie houdt in dat de data gecapteerd wordt op de plaats waar het zwerfvuil aangetroffen wordt. Concreet betekent dit dat het beeldmateriaal verzameld wordt op het openbaar domein. Bijgevolg kunnen

externe factoren zoals weersomstandigheden een invloed uitoefenen op de metingen en “ruis” veroorzaken in de gemaakte beelden.

Bij deze methodologie is men wel zeker dat alle zwerfvuil fracties in kaart worden gebracht daar men ter plekke de captatie en analyse gaat uitvoeren. Een extra voordeel kan de noodzakelijke GPS integratie zijn die het gevonden zwerfvuil zal toekennen aan één van de type omgevingen. Via de GPS-data wordt immers de locatie bepaald van het beeldmateriaal. Dit zorgt automatisch voor een matching van het zwerfvuil en zijn locatie, iets wat in het huidige proces manueel bijgehouden wordt en dus open staat voor menselijke fouten.

Ex-situ houdt in dat de analyse gebeurt op een centrale verzamelplaats en wordt toegepast op de huidige manuele manier van monitoring. Het betekent concreet dat het zwerfvuil opgehaald wordt op vooraf (willekeurig) bepaalde plaatsen, vervolgens verzameld wordt in een centrale verzamelplaats en dan pas geanalyseerd. Bij deze methodologie moet de ophaler per geografische cel van 10 bij 10 meter het zwerfvuil in een aparte vuilzak verzamelen. Dit om het achteraf, wanneer het zwerfvuil op de transportband/vloer wordt geleegd, aan de juiste geografische zone toe te wijzen. Hierdoor ontstaat het risico op menselijke fouten. Er moet nauwkeurig worden bijgehouden welke vuilzak wanneer op de band/vloer gelegd wordt en van welke geografische cel die vuilzak afkomstig is om foute conclusies te vermijden. De ervaring leert ons echter dat deze huidige methode in de praktijk vlot en efficiënt werkt. Er wordt zelden een manuele fout ontdekt.

Bij Ex-situ wordt boven de transportband/vloer een camera geplaatst en zo de data gecapteerd. Door op deze manier te werken worden externe factoren – zoals weersomstandigheden, terreinondergrond en lichtinval – die een uitdaging vormen voor AI tot een minimum beperkt. Bovendien bevatten de beelden alleen de objecten (zwerfvuil fracties) en geen randinformatie. Dit vormt een beduidend voordeel ten opzichte van In-situ. Voorgaande onderzoeken die van een ex-situ strategie gebruik maakten, haalden goede resultaten. Zo behaalde onderzoek van Kulkarni et al. (2019) een *mean Average Precision* (mAP) van 85% op zes klassen. Theorie en berekeningen achter mAP wordt later in het rapport verder toegelicht.

Een ander uitdaging is het garanderen dat alle fracties in de monitoring worden meegenomen. Neem bv. kauwgom, een fractie dat niet altijd eenvoudig van zijn locatie mee te nemen is. Echter behoort kauwgom tot één van de meest voorkomende vuilnisfracties (in aantal stuks). Als deze fractie niet meegenomen wordt in de metingen, ontstaat er een fout beeld over de netheid in Vlaanderen.

Tot slot is het belangrijk op te merken dat bij deze methodologie het proces gelijkaardig is aan de huidige manuele tellingen. Er wordt nog steeds een manuele verzameling van zwerfvuil uitgevoerd, waardoor enkel het tellen van het aantal fracties geautomatiseerd zal worden en de winst die OVAM met deze aanpak maakt zeer beperkt wordt.

Bovenstaande voor- en nadelen kunnen in onderstaande tabel worden samengevat.

Tabel 1: In-situ/ex-situ

	In-situ	Ex-situ
Voordelen	• Alle zwerfvuil fracties worden in kaart gebracht in de monitoring daar deze op locatie gebeurt • Automatische matching zwerfvuil en de locatie door GPS component • Zwerfvuil detectie/monitoring gebeurt gelijktijdig met opruiming waardoor veel manuele handelingen wegvallen	• “Ruis” op de data wordt geminimaliseerd (bv. lichtinval, weer) • Enkel object worden gecapteerd en geen randinformatie
Nadelen	• Gevoelig aan “ruis” op de data. Omgevingsfactoren zijn moeilijk controleerbaar (bv. lichtinval, weer)	• Mogelijks worden niet alle fracties opgenomen (bv. kauwgom kan moeilijk te verzamelen zijn) • Nauwelijks verschil met het huidige manuele proces van zwerfvuil monitoring

In **conclusie**, vereenvoudigt een Ex-situ aanpak de AI-component van het systeem door de “ruis” in de data te minimaliseren en zo hoogstwaarschijnlijk de nauwkeurigheid van monitoring te verhogen. Echter wegen de resterende operationele nadelen van Ex-situ harder door dan de voordelen. Er wordt dermate weinig tijd en kosten bespaard ten opzichte van het oude manuele Ex-situ proces, waardoor het economisch niet verantwoord is voor de OVAM om voor een Ex-situ aanpak te kiezen. In samenspraak met de OVAM werd dan ook besloten om te kiezen voor een In-situ aanpak.

Tot slot is het noemenswaardig om een bijkomende complexiteit die geldt voor beide aanpakken te vermelden, nl. de grens tussen **publiek en privaat domein**. Per definitie is zwerfvuil afval dat zich op openbaar terrein bevindt. Zwerfvuil gevonden op privaat domein dient dus niet in kaart te worden gebracht. In een Ex-situ aanpak kan hier nog enigszins rekening mee gehouden worden door bijvoorbeeld afval in tuinen niet mee te nemen, desondanks is deze afbakening zelfs voor een persoon soms moeilijk waarneembaar. Bij In-situ dient het systeem deze grens automatisch te detecteren, wat nog complexer is dan Ex-situ. Een mogelijke oplossing voor de In-situ aanpak is het integreren van de GPS met de camera’s en kadastrale informatie. Hierdoor is door middel van data gekend wat de grenzen zijn en kan hierop theoretisch gezien gefilterd worden. Om deze filtering te kunnen doen wordt een zekere nauwkeurigheid vereist ten opzichte van de geografische lokalisatie van de geïdentificeerde zwerfvuilobjecten. Hoe nauwkeuriger de GPS, hoe beter het systeem uiteindelijk zal presteren. Een harde grens naar nauwkeurigheid is dus moeilijk te definiëren want is afhankelijk van de beoogde accuraatheid. Vermoedelijk zal een nauwkeurigheid van enkele tientallen cm volstaan om een acceptabele nauwkeurigheid van zwerfvuiltelling te bekomen. Uit paragraaf 2.4 zal echter blijken dat indien gewenst wordt gebruik te maken van meerdere modi, een probleem van dubbeltelling ontstaat. Om dit probleem te kunnen

verhelpen en toch de verschillende modi te kunnen gebruiken zal de GPS nauwkeurigheid een pak beter moet zijn en op cm niveau nauwkeurigheid moeten zijn.

In elk geval is het duidelijk dat de nauwkeurigheid van de GPS cruciaal zal zijn voor de oplossing. De GPS nauwkeurigheid en dus de haalbaarheid van deze oplossing, wordt in paragraaf 0 verder uiteengezet.

2.1.2 Transport modus

Door de keuze van In-situ, zal de camera naar de locatie moeten gebracht worden en ontstaat een nieuwe keuze die gemaakt dient te worden, nl. de transport modi die gebruikt wordt tijdens de data captatie. Er worden vier opties naar voor geschoven door de OVAM: te voet, al fietsend, met een wagen en via een drone. De keuze van modus heeft een grote invloed op de uiteindelijke oplossing.

Dronebeelden verschillen significant van beelden genomen via de andere modi. Drones vliegen typisch op een hoogte van zeven meter, waar camera's bevestigd op wagens, fietsen of bevestigd aan een persoon opereren op een hoogte van één tot twee meter. Dit veroorzaakt een extra complexiteit voor het toepassen van AI. Wanneer gekozen wordt voor een combinatie van drones en andere modi zijn de beelden te verschillend om ze door eenzelfde classificatie mechanisme te laten verwerken en kan een dergelijk model, getraind op dronedata, niet veralgemeend worden naar beelden van de andere modi. De andere modi opereren op vergelijkbare hoogten waardoor het haalbaar is om de camera met eenzelfde oriëntatie te laten filmen. Ter vereenvoudiging worden de vier modi herleid tot twee groepen, nl. de lucht-modi (drones) en grond-modi (bodycam, fiets, wagen). Voor beide groepen worden de voor- en nadelen toegelicht.

Het gebruik van **grond-modi** heeft het voordeel dat moeilijk bereikbare plaatsen met de wagen kunnen opgevangen worden door te kiezen voor een andere modus (fiets of bodycam). Denk hierbij bijvoorbeeld aan straten waar geparkeerde auto's het zicht van de camera op het voetpad belemmeren. Een belangrijke kanttekening hierbij is dat wanneer men te vaak van modus moet veranderen, de business case sterk gereduceerd wordt. In het slechtste geval moet de bestuurder in de wagen door blokkering van het zicht voor iedere 10 bij 10 meter cel uitstappen om voetpaden en dergelijke in kaart te brengen. In dit geval is het huidige systeem van manuele tellingen hoogstwaarschijnlijk even efficiënt. Het risico op de blokkering van het zicht van de camera is het grootste risico van de grond-modi door de hoogte waarop de camera's zich bevinden.

Een ander voordeel is de mogelijkheid om *transfer learning* toe te passen bij het ontwikkelen van het AI model. Bij deze techniek kan men verder bouwen op reeds bestaande en nauwkeurige modellen, die gebaseerd zijn op beelden met dezelfde oriëntatie als onze grond-modi. Hierdoor kan men beduidend sneller tot zeer goede resultaten komen. Het concept van *transfer learning* wordt later in dit hoofdstuk onder het luik Artificiële Intelligentie toegelicht. Ook dit voordeel kan genuanceerd worden daar de eerste stappen in de reeds bestaande modellen zich focussen op details in afbeeldingen zoals randen en vormen, en niet zozeer op objecten. Uit dit perspectief is het dus, net zoals voor de grond-modi, mogelijk om deze specifieke modellen te gebruiken voor de lucht-modi. Het voordeel dat enkel geldt voor de grond-modi ligt vooral in de stappen waarbij men object detectie zal doen.

Een meer uitgesproken voordeel voor grond-modi is de eenvoud waarmee nieuw beeldmateriaal kan bekomen worden, wat de nauwkeurigheid van de AI-module kan bevorderen. Het inzetten van drones zal steeds een zekere planning vereisen en kan voor logistieke uitdagingen zorgen. Denk maar aan het inplannen van medewerkers met een geldend attest van afstandspiloot, de garantie op veiligheid door te filmen op dagen met weinig wind en de korte batterijduur. Daarnaast moet deze drone ingeschakeld worden specifiek voor deze opdracht, terwijl grond-modi meerdere doelen kunnen hebben en dus praktisch automatisch nieuw beeldmateriaal verwerven. Denk hierbij aan het voorbeeld van afvalophaling waar de wagens uitgerust worden met camera's. Telkens de wagen uitrijdt kan de camera nieuw beeldmateriaal maken.

Het gebruik van **lucht-modi** of drones wordt op Europees niveau gereguleerd. Hieruit volgt het volgende:

1. De Europese regelgeving bestaat uit Verordening (EU) 2019/945 en Uitvoeringsverordening (EU) 2019/947². Op Belgisch niveau zijn deze Europese eisen uitgevoerd in het Koninklijk besluit van 8 november 2020 tot uitvoering van uitvoeringsverordening (EU) 2019/947 van de Commissie van 24 mei 2019 inzake de regels en procedures voor de exploitatie van onbemande luchtvaartuigen en de Ministeriële uitvoeringsbesluiten van 21 en 29 december 2020.
2. Betreffende het gewicht van de drone (belangrijk, vermits een zekere autonomie vereist wordt alsook het gebruik van camera's): een drone van het model C2, overeenkomstig Verordening (EU) 2019/945 bijlage deel iii, mag een maximumstartmassa (MTOM) hebben van 4kg, met lading inbegrepen; bij het model C3 (volgens deel IV) is dit dan weer 25 kg en een afmeting van minder dan 3 meter en bij het model C4 (volgens deel V) is dit minder dan 25 kg zonder begrenzing in afmeting.
3. Betreffende de toegelaten vluchttuitvoeringen: De vereiste dat men zowel woongebieden als niet-betrokken personen kan overvliegen beperkt het aantal mogelijke opties. Hier lijkt het aangewezen te kiezen voor de UAS-vluchttuitvoeringen in de categorie "specifiek". Voor een eenmalige geplande vluchttuitvoering kan men gebruikmaken van een operationele risicobeoordeling (conform Uitvoeringsverordening EU 2019/947 bijlage Deel B UAS.SPEC.010), maar dit geeft problemen betreffende het overvliegen van niet-betrokken personen (want beperkt door de standaardscenario's). Ook administratief is het niet interessant. Voor meerdere vluchten lijkt het dus aangewezen te opteren voor een exploitatieverklaring (UAS.SPEC.020) of een vergunning (UAS.SPEC.030 en UAS.SPEC.040). Een alternatief is dat OVAM als rechtspersoon opteert voor het bekomen van een certificaat van exploitant van lichte UAS (of LUC), overeenkomstig artikel 40 Verordening (EU) 2019/945 en Uitvoeringsverordening EU 2019/947 bijlage Deel C UAS.LUC.010 e.v..
4. De maximale vlieghoogte bedraagt in beginsel 120 meter.
5. De exploitant -en pilootverantwoordelijkheden zijn opgesomd in de artikelen 7, 8 en 9 Uitvoeringsverordening (EU) 2019/947 en – voor wat betreft onze toepassing – verder gepreciseerd in bijlage Deel B UAS.SPEC.050 en UAS.SPEC.060. Het Koninklijk besluit van 8 november 2020 voert de Europese eisen op nationaal niveau uit. Specifiek verlangt de Belgische wetgever van piloten op afstand dat deze een attest van bestuurder van een RPA of bewijs van bevoegdheid als bestuurder van een RPA in bezit hebben (artikel 13). Men verwijst hier in dit artikel wel naar het voormalige (inmiddels opgeheven) Koninklijk besluit van 10 april 2016 voor wat betreft de te volgen procedure om dit attest of bewijs te bemachtigen, wat toch enige rechtsonzekerheid met zich meebrengt.

² <https://www.easa.europa.eu/sites/default/files/dfu/Easy%20Access%20Rules%20for%20Unmanned%20Aircraft%20Systems.pdf>

6. Er dient rekening gehouden te worden met de door de Belgische wetgever vastgelegde geografische zones waar vliegen met een drone, hetzij verboden, hetzij enkel toegelaten is mits toestemming (artikel 15 Uitvoeringsverordening (EU) 2019/947 juncto Ministerieel besluit van 21 december 2020 tot vaststelling van vaste geografische UAS-zones en toegangsvoorwaarden voor vaste geografische UAS-zones). In overeenstemming met artikel 15.3 van voormelde verordening heeft de Belgische wetgever deze geografische zones digitaal³ beschikbaar gemaakt.

Eenzijds zorgt het feit dat meermaals door de Belgische wetgeving verwezen wordt naar een opgeheven KB voor rechtsonzekerheid. Onder andere voor de procedure voor het behalen van de nodige attesten. Er gelden momenteel overgangsmaatregelen die gelden tot ten laatste 2023. Indien gekozen wordt voor een uitrol is de kans groot dat een AI-systeem, niet vóór 2023 in gebruik zal zijn. Daarom zijn deze maatregelen mogelijk niet van belang. Anderzijds is de regelgeving op Europees niveau strikt vastgelegd, maar heerst er nog steeds onduidelijkheid over hoe de Cel Drones dit in de praktijk zal implementeren. De Cel Drones zit de Belgian Civilian Drone Council (BCDC) voor, die recent een roadmap⁴ gepubliceerd heeft. Hieruit valt af te leiden dat de vooruitgang op vlak van implementatie eerder beperkt is. Om deze redenen adviseert Ordina om bij een keuze voor het gebruik van lucht-modi contact op te nemen met de Cel Drones om na te gaan wat de mogelijkheden en beperkingen zijn.

Het voordeel van de **lucht-modi** of drones is dat de afbeeldingen een overzicht geven van de omgeving. Het risico blokkering van zicht door obstakels op en langs de weg is beduidend kleiner daar de drone een flexibelere beweegruimte heeft in vergelijking met de grond modi, enkel overhangende obstakels zoals terrassen, overdekkingen en kruinen van bomen kunnen een probleem vormen. Dit probleem kan eenvoudig weg opgelost worden door de drone lager te laten vliegen.

Een volgend voordeel is de bewegingsvrijheid van de lucht-modi. Drones hoeven immers in theorie geen rekening te houden met straten doordat ze over gebouwen kunnen vliegen. In de praktijk zorgt dit voordeel echter voor een nadeel voor de kwaliteit van het beeldmateriaal en of deze voldoende scherp zijn om een nauwkeurig AI model te bouwen. Hierdoor vervalt het voordeel, daar het zorgt dat een degelijke automatische detectie niet haalbaar is.

Een eerste nadeel van drones is de batterijduur, die in de meeste gevallen één uur vliegen toelaat. Deze tijd daalt wanneer de drone extra gewicht moet dragen zoals die van de camera's in onze situatie. Daarnaast is er een hoog risico dat zelfs met precieze camera's kleinere of vertrappelde zwerfvuilobjecten niet waarneembaar zullen zijn, zelfs niet met het blote oog. Het niet opnemen van fracties met kleine zwerfvuilobjecten zou een grote limitatie voor de oplossing betekenen en is daarom niet wenselijk. Ten derde is het gebruik van drones gevoelig aan omgevingsfactoren zoals wind, die aanleiding kunnen geven tot botsingen.

³ <https://map.droneguide.be/>

⁴ https://mobilit.belgium.be/sites/default/files/resources/files/roadmap_1.pdf

Naast de technische nadelen heerst er een ander, eerder maatschappelijk risico. Het inzetten van drones zal niet onopgemerkt blijven. Het ontstaan van een ‘big brother’ gevoel bij de brede bevolking is bij deze aanpak niet onrealistisch.

In onderstaande tabel worden de voor- en nadelen van beide groepen overzichtelijk samengevat.

Tabel 2: Voor- en nadelen van de verschillende modi

	Lucht-modi (drones)	Grond-modi (auto, fiets, voet)
Voordelen	• Bewegingsvrijheid (let op: rekening houdend met rode zones) • Obstakels die het beeld versperren beperkt tot overhangende obstakels.	• Combineren van modi mogelijk • Combinatie vuilnis ophaling en zwerfvuilmonitoring mogelijk • <i>Transfer learning</i> mogelijk
Nadelen	• Modellen moeilijk te veralgemenen naar andere modi. • Strikte wetgeving drones en rechtsonzekerheid • Zwerfvuilmonitoring is niet combineerbaar met ander proces • Beperkte mogelijkheid <i>transfer learning</i> • Blokkering van zicht door overhangende obstakels • Beperkte vliegtijd door batterij • Limitatie detectie kleinere fracties • Omgevingsfactoren zoals wind	• Beperkte bewegingsvrijheid: Gebonden aan infrastructuur en verkeersregels • Blokkering van het zicht door obstakels

Op basis van bovenstaande voor- en nadelen werden **conclusies** getrokken en in overleg met de OVAM gekozen voor de grond-modi. Een combinatie systeem van auto, fiets en bodycam geeft de nodige flexibiliteit bij een In-situ aanpak en verlaagt het risico op niet voldoende kwalitatieve data die AI voor automatische detectie mogelijk maakt. In wat volgt wordt uitgegaan van een grond-modi systeem. Drones worden buiten beschouwing gelaten.

2.1.3 Media formaat

Een aspect dat onderzocht dient te worden is het formaat van de media. Enerzijds kan rechtstreeks gewerkt worden met foto's en een zekere frequentie waarmee die foto's gemaakt worden. Anderzijds is het mogelijk om het beeldmateriaal te verzamelen in de vorm van video's. Bij deze laatste is het belangrijk op te merken dat een video een aaneenschakeling is van afbeeldingen en het relatief eenvoudig is een video om te zetten naar *frames* die apart opgeslagen kunnen worden. Met *frames* wordt specifiek bedoeld op foto's geëxtraheerd uit een video. Bij video's kan een *sample rate* gedefinieerd worden waarmee bepaald wordt hoeveel en welke *frames* uit de video worden gehaald om zo het datavolume te beperken. Deze manier van aanpak geeft flexibiliteit en ondersteunt de mogelijkheid tot doorgedreven onderzoek op gelijkaardige afbeeldingen indien dit noodzakelijk moest zijn.

Een belangrijk risico dat hoger ligt bij video's is dubbeltellingen. Indien er twee seconden een blikje wordt gefilmd met een fps (*frames per second*) van 30, dan resulteert dit in 60 foto's van hetzelfde blikje. De bekomen *frames* worden sequentieel aan het AI model gevoed. AI modellen zonder geheugen hebben geen notie van wat ze op een vorige frame hebben gedetecteerd en bekijken dus elke frame zonder rekening te houden met de video stream. In ons voorbeeld zou het zwerfvuilobject op elk van de 60 *frames* voorkomen, 60 keer gedetecteerd worden en dus 60 keer geteld worden. De te overwegen AI modellen voor de probleemstelling van de OVAM worden in meer detail besproken in hoofdstuk 2.3.1.2. Uit deze paragraaf zal blijken dat object detectie algoritmen het meest geschikt zijn binnen de context van deze opdracht. Dit soort algoritmen hebben echter geen geheugen waardoor deze probleemstelling reëel wordt.

Werken met foto's kan dit probleem oplossen. Dit door het tijdsinterval waarin foto's getrokken worden af te stellen op de snelheid van het voertuig. Zodoende kan overlap in sequentiële foto's vermeden worden, waardoor een object telkens op slechts één afbeelding zichtbaar zal zijn. Deze methode werd toegepast door Rad et al. (2017): door een constante snelheid van 12km/u aan te houden en om de halve seconde een foto te trekken werd elk zwerfvuilobject éénmaal waargenomen.

In een gecontroleerde omgeving, zoals in een POC, is deze aanpak van constante snelheid en frequentie veelbelovend en eenvoudig. Echter zijn er in praktijk verscheidene oncontroleerbare factoren die het rijden aan een constante snelheid onmogelijk maken. Denk maar aan verkeerslichten of overstekende voetgangers. Daarbij komt dat je door het gebruik maken van foto's maar één kans hebt om het zwerfvuil te detecteren. Eénzelfde omgeving wordt maar één keer vastgelegd door de camera. Wordt het zicht op het zwerfvuilobject geblokkeerd, is er geen 2^e kans (uit een ander perspectief) op een succesvolle foto.

Om samen te vatten resulteren video's in een groot aantal *frames* met telkens een grote overlap. Dubbeltelling is in dit scenario moeilijker te vermijden. Foto's daarentegen leiden tot slechts een eenmalige captatie van zwerfvuilobjecten. Hierdoor is de kans reëel dat sommige zwerfvuil fracties niet geregistreerd worden. Daarnaast geeft de methodologie met foto's aanleiding tot een aantal operationele problemen. Daar de risico's verbonden aan het nemen van foto's praktisch zeer moeilijk oplosbaar zijn, wordt gekozen voor een oplossing met video's. Het probleem met dubbeltellingen gebonden aan video-opnames, wordt in hoofdstuk 2.4.1 verder in detail besproken.

2.1.4 Keuze camera

Tot slot moet naast de keuze van manier van telling, de transport modi en media formaat, ook een keuze gemaakt worden omtrent de camera. Deze dient aan volgende eisen te voldoen:

- Camera specificaties:
 - Kwaliteit van minstens 2k. In eerder onderzoek uitgevoerd door Rad et al. (2017), waar de focus vooral lag op het detecteren van kleine objecten zoals sigaretten en bladeren, werden de kleine objecten met een 2k camera succesvol geïdentificeerd met een *precisie* en *sensitiviteit* tussen 65% en 80%. Voor dit onderzoek was weinig data voor handen om toch tot een degelijk resultaat te komen⁵. Bijgevolg lijkt een 2k camera voldoende om deze opdracht succesvol uit te voeren.
 - Geïntegreerde GPS om succesvol de gevonden zwerfvuilobjecten te mappen met het juiste 10 bij 10 meter cel (zie paragraaf 2.1.1).
- Operationele vereisten:
 - Een stabilisator die schokken tijdens captatie kan opvangen. Denk maar aan een wagen die over een oneffenheid in de weg rijdt.
 - Beperkt in omvang en gewicht zodoende het comfort van de bodycam-oplossing te garanderen.
 - De mogelijkheid om de camera aan een harnas (bevestiging nodig voor bodycam) en zuignap (bevestiging nodig voor wagen) te monteren.
 - Waterdichtheid om in verschillende weersomstandigheden te opereren.

Er zijn een beperkt aantal camera's op de markt die aan al deze eisen voldoen. De GPS, draagbaarheid en waterdichtheid zijn specificaties die vooral gevonden kunnen worden in het *actioncam* gamma. In deze categorie is het merk GoPro de bekendste met hun Hero series. De voornaamste concurrent is DJI, maar deze voldoen niet aan de GPS vereisten waardoor in het verdere onderzoek geopteerd werd voor een GoPro camera.

Binnen de Hero reeks is de GoPro Hero 9 de nieuwste en meest geavanceerde camera. De verbeteringen ten opzichte van zijn voorganger zijn naast de hogere beeldkwaliteit (4k-kwaliteit), een betere stabilisator en een langere batterijduur. Dit zijn belangrijke functionaliteiten die grote implicaties kunnen hebben voor het uiteindelijk gekozen systeem en waarvan de minimale vereisten getoetst dienen te worden tijdens de POC. Dit wil niet zeggen dat in de eind architectuur zal gekozen worden voor deze camera. Het geeft louter de flexibiliteit om te toetsen wat de minimale resolutie moet zijn en welke andere camera's, mogelijks goedkoper, dus ook mogelijk zijn. Uit paragraaf 2.4.2.1.2 zal bijvoorbeeld blijken dat het tevens nuttig is om de ZED2 aan te schaffen en te testen in de POC.

⁵ 69 sigaretten, 958 bladeren en 394 hopen van bladeren.

2.1.5 Besluit datacaptatie

In het hoofdstuk omtrent datacaptatie werden vier belangrijke beslissingen genomen, namelijk:

1. Er zal in-situ beeldmateriaal opgenomen worden.
2. De gekozen modus voor datacaptatie is de combinatie van auto, fiets en bodycam.
3. Men kiest voor video's in plaats van foto's, ongeacht de problematiek omtrent dubbeltellingen.
4. De GoPro Hero 9 wordt gebruikt als multifunctionele camera die mogelijks overgekwalificeerd zal blijken tijdens de POC.

Bij alle keuzes werden belangrijke aannames omtrent de accuraatheid van de GPS gemaakt. Theoretisch gezien zijn deze tot op een meter nauwkeurig. Echter bleek tijdens het onderzoek en de toetsing dit een grote uitdaging te zijn, waardoor gekozen werd om een apart hoofdstuk te wijden aan de accuraatheid van de GPS. Ondanks dat dit cruciaal is bij een In-situ benadering, blijft Ex-situ economisch niet verantwoord en werd gekozen om de probleemstelling van de GPS verder uit te diepen. In volgend hoofdstuk wordt deze uitdaging in meer detail bekeken.

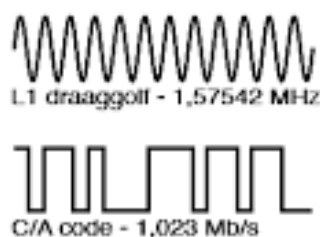
2.2 ACCURAAATHEID GPS

Zoals hierboven aangegeven, moeten de objecten op gecapteerde beelden worden toegekend aan een type omgeving op basis van de 10 bij 10 meter cellen. Daarom is het cruciaal om een voldoende nauwkeurige geografische locatie aan een genomen beeld te kunnen koppelen. Theoretisch gezien kunnen GPS signalen tot op de meter nauwkeurig zijn. Onderzoek wijst echter uit dat dit niet het geval is. Om de problematiek van de GPS beter te begrijpen wordt eerst de werking van een GPS en de oorzaken van het probleem toegelicht. Vervolgens worden andere alternatieven GPS-systemen bekeken.

2.2.1 GPS werking en probleemstelling

Eenvoudig gesteld bestaat een GPS signaal uit twee gesuperponeerde⁶ signalen. Een C/A code ("Civilian Acquisition") die gemoduleerd is op een draaggolf. Naar deze C/A code wordt daarom vaak verwezen als de modulatie. Beide signalen worden voorgesteld op Figuur 1.

Figuur 1: Onderdelen van een GPS signaal⁷

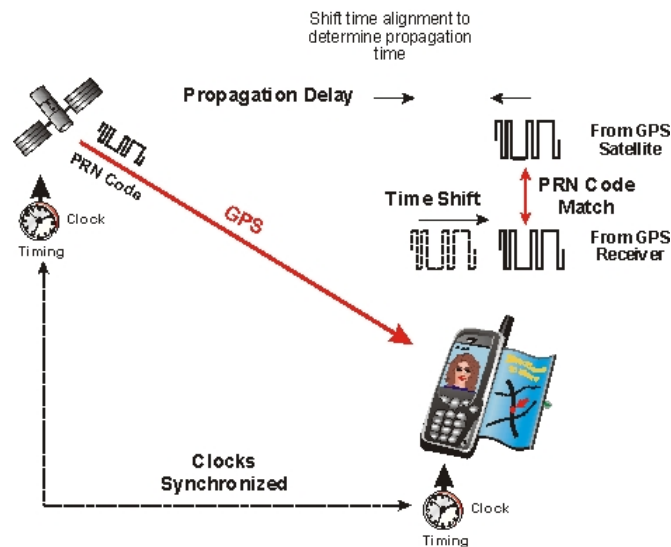


De C/A code is gekend door zowel GPS zenders (satellieten) als door de GPS ontvangers. Het is dus de C/A code die gebruikt wordt om te bepalen wat de tijdsverschuiving is tussen het uitgezonden signaal en het verder geëvolueerde signaal in de ontvanger. Door deze tijdsverschuiving te bepalen en te vermenigvuldigen met de snelheid waarmee het GPS signaal door de ruimte beweegt kan de afstand bepaald worden tussen de GPS zender en de GPS ontvanger. Dit wordt schematisch voorgesteld in Figuur 2. Van zodra een ontvanger zijn positie kent ten aanzien van minstens drie satellieten kan mathematisch zijn locatie berekend worden. In de praktijk wordt echter gebruik gemaakt van meerdere satellieten om tot een nauwkeurigere locatiebepaling te komen.

⁶ [https://nl.wikipedia.org/wiki/Superpositie_\(natuurkunde\)](https://nl.wikipedia.org/wiki/Superpositie_(natuurkunde))

⁷ http://www.fi.uu.nl/wiskrant/artikelen/282/282december_bos.pdf

Figuur 2: GPS tijdverschuiving⁸



Uit meervoudig onderzoek blijkt echter dat de nauwkeurigheid van GPS systemen ver afwijken van één meter. Verschillende onderzoeksinstellingen hebben de oorzaken hiervoor onderzocht. In Tabel 3 is een samenvatting terug te vinden van de grootste oorzaken beschreven door Trimble Inc., een Amerikaans bedrijf dat onder andere navigatie oplossingen bouwt voor verschillende sectoren.

Tabel 3: Fout-oorzaken van GPS signalen⁹

	Standaard GPS [m]	DGPS [m]
Satelliet klokken	1.5	0
Orbit fouten	2.5	0
Ionosfeer	5	0.4
Troposfeer	0.5	0.2
Ontvanger ruis	0.3	0.3
Meervoudig pad	0.6	0.6
Opzettelijke fouten	30	0

Oorspronkelijk werd de grootste fout in GPS systemen opzettelijk geïntroduceerd. De voornaamste reden was het voorkomen dat terroristische organisaties GPS zouden gebruiken om accurate wapens te produceren. In 2006 werd deze opzettelijke fout in opdracht van Bill Clinton verwijderd waardoor burgersignalen van GPS satellieten een veel hogere nauwkeurigheid hadden.

⁸ http://www.wirelessdictionary.com/wireless_dictionary_GPS_Propagation_Time_Measurement_Definition.html

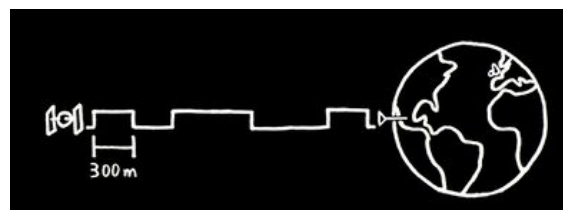
⁹ https://www.trimble.com/gps_tutorial/howgps-error2.aspx

Met de opzettelijke fout verwijderd uit het systeem, is de grootste afwijking het gevolg van de ionosfeer. Deze problematiek is te wijten aan de vrije elektronen in deze luchtlag. In eerste instantie werd dit voor militaire toepassingen opgelost door C/A code uit te zenden via twee draaggolven (L1 en L2). Door het verschil te bepalen in ontvangsttijd tussen L1 en L2 kan de invloed van de ionosfeer geneutraliseerd worden. Burgertoepassingen kunnen dit echter niet gebruiken. Het tweede signaal vereist namelijk een militaire decoder.

Omdat sommige burgertoepassingen toch een grotere nauwkeurigheid vereisten zijn meer geavanceerde systemen ontwikkeld om alsnog de invloed van de ionosfeer te neutraliseren. Deze systemen maken gebruik van het idee van twee draaggolven waarbij gebruik gemaakt wordt van twee ontvangers van hetzelfde GPS signaal. Eén van de ontvangers wordt vast geplaatst op een gekende locatie, terwijl een tweede ontvanger op de locatie staat die men wil bepalen. Door data van beide systemen te combineren en het verschil tussen beide signalen te berekenen kan de invloed van de ionosfeer berekend worden. Deze systemen worden in de literatuur "*Differential GPS*" of kortweg DGPS genoemd.

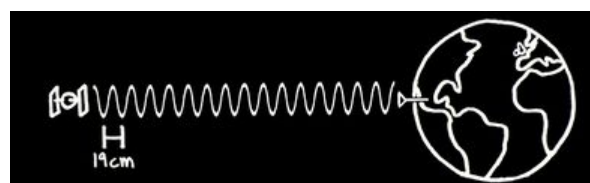
DGPS systemen maken gebruik van de modulatie van het GPS signaal om het verschil tussen beide ontvangers te berekenen. Elke eenheid binnen deze modulatie (=periode) dekt een afstand van ongeveer 300 m, zoals zichtbaar op Figuur 3. Deze grote afstand geeft aanleiding tot een beperking van nauwkeurigheid.

Figuur 3: Afstand van een periode de modulatie van het GPS signaal ¹⁰



Tot slot kan men om dit probleem aan te pakken en de nauwkeurigheid verder te verbeteren gebruik maken van RTK toestellen. Deze toestellen maken niet enkel gebruik van de informatie gegeven door de modulatie, maar ook van de informatie gegeven door de draaggolf. De afstand van een periode in de draaggolf is slechts 19 cm wat aanleiding geeft tot een hogere accuraatheid.

Figuur 4: Afstand van een periode in de draaggolf van het GPS signaal ¹⁰



¹⁰ <https://support.swiftnav.com/support/solutions/articles/44001850808-understanding-gps-gnss-rtk-technology>

2.2.2 Alternatieven

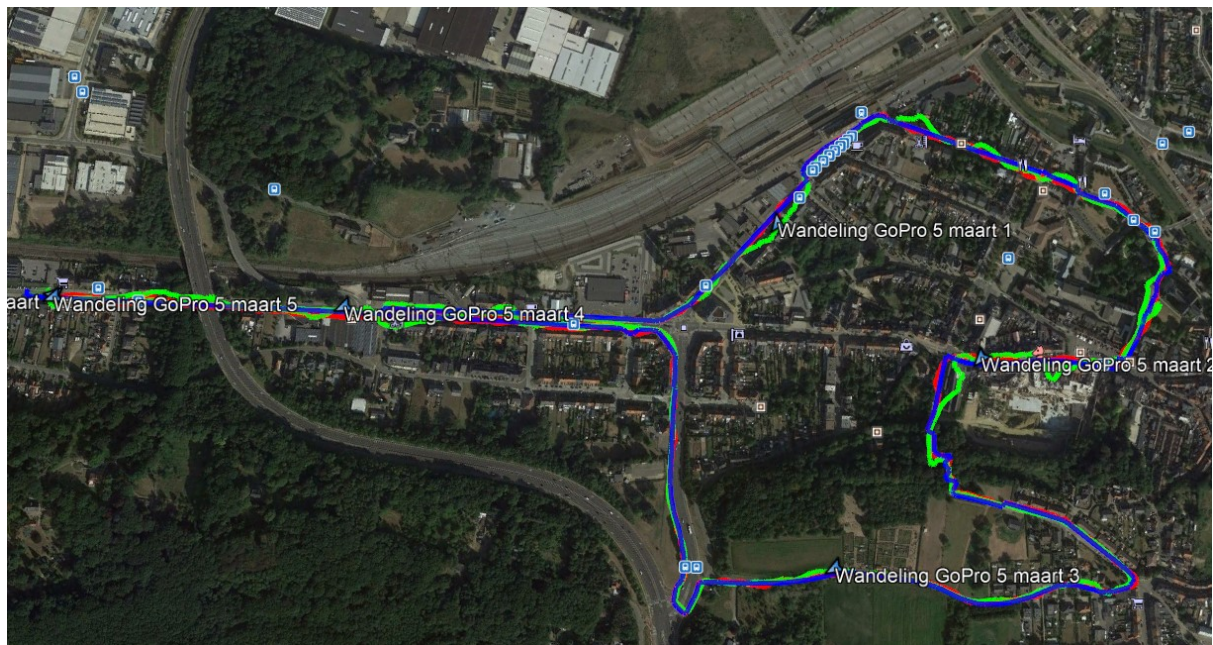
Door de onnauwkeurigheid gevonden in de GPS systemen van onze GoPro wordt gekeken naar andere mogelijke systemen. Enerzijds wordt de nauwkeurigheid van het GPS systeem van een iPhone bekeken. Hier worden betere resultaten verwacht daar Apple GPS combineert met andere informatie om zo betere precisie te bekomen. Daarnaast wordt gekeken naar de ArduSimple simpleRTK2B, een RTK toestel specifiek gebouwd voor een betere nauwkeurigheid. Tot slot wordt het toekomstig Europees Galileo project toegelicht.

2.2.2.1 GoPro versus iPhone

Theoretisch gezien wordt verwacht dat een iPhone GPS nauwkeuriger zal presteren dan de GPS van de camera gebruikt in de POC. Om de accuraatheid in de praktijk te testen, werd een route gevolgd door Aarschot waarbij bewust werd gekozen voor een combinatie van verschillende soorten gebieden, zoals dichtbebouwd stedelijk gebied, weiland en een open steenweg. Dit om ook hiervan de impact op de GPS resultaten te onderzoeken. Enerzijds werd de route belopen met de Go Pro Hero 9, anderzijds met een iPhone X. Beide systemen volgen dezelfde route waardoor vergelijking en dus prestatie achteraf vergeleken kunnen worden.

Op Figuur 5 wordt de effectieve wandeling in het blauw weergegeven, de iPhone in het rood en de GoPro in het groen. De weersomstandigheden bij deze wandeling waren goed, een open hemel met slechts enkele wolken.

Figuur 5:Wandeling iPhone vs. GoPro



Figuur 6 en Figuur 7 zoomen in op twee interessante locaties. De eerste betreft een vrij open omgeving: een straat met voortuinen voor de huizen. De tweede is een dichtbebouwde stadskern waar het voetpad vlak naast de huizen ligt. We zien in deze twee afbeeldingen duidelijk dat de GPS signalen in de dichtbebouwde omgeving verder van de effectieve wandeling ligt en het signaal meer verstoord wordt. Bij de open omgeving is de accuraatheid veel sterker.

Figuur 6 Wandeling - Open omgeving met bebouwing



Figuur 7 Wandeling - Dichtbebouwd



Een volledige samenvatting van de verschillende soorten omgevingen staat weergegeven in Tabel 4. Hierin wordt de maximale afwijking van de effectieve route weergegeven die in elk type gebied werd gemeten.

Tabel 4: Maximale afwijking GPS signaal per gebied

Maximale afwijking GPS signaal per gebied			
	GoPro Hero 9	iPhone X	Consistentie
Open weide	6 m	5 m	iPhone beter, zeer klein verschil
Brede open weg	4 m	3 m	iPhone beter, zeer klein verschil
Smalle weg, open langs één kant	3 m	4 m	GoPro beter, zeer klein verschil
Zeer smalle straat (één richting, bebouwing aan beide kanten)	30 m	8 m	GoPro zeer slecht, iPhone afwijkend
Druk dichtbebouwd centrum, veel verkeer	28 m	15 m	GoPro zeer slecht, iPhone ook zeer afwijkend
Minder dicht centrum, minder verkeer	20 m	5 m	GoPro zeer slecht, iPhone beter
Steenweg + lintbebouwing, soms dicht bij huizen, soms meer open	22 m	7 m	GoPro zeer slecht, iPhone beter (op open delen even goed)
Overdekkingen	14 m	10 m	iPhone beter

Uit dit onderzoek blijkt dat het GPS systeem van de GoPro niet voldoende nauwkeurig kan zijn voor automatische zwerfvuildetectie waarbij met 10 bij 10 meter cellen wilt hanteren. En Apple GPS presteert beter, maar zal ook in bepaalde type omgevingen niet volstaan, zoals bv. in een stadscentrum. Er wordt verder gekeken naar de RTK toestellen om na te gaan of daar wel voldaan kan worden aan alle omgevingen.

2.2.2.2 Ardusimple simpleRTK2B

Eerder viel reeds te lezen dat een RTK GPS systeem de nauwkeurigheid drastisch zou moeten verbeteren. Een mogelijks interessante implementatie van dit systeem is de Ardusimple simpleRTK2B. Dit is een Arduino applicatie bord met een veronderstelde centimeter-level precisie in open gebied. Onderzoek van Hamza et al. (2020) geeft aan dat deze hoge precisie ook geldt in stedelijk gebied. Hierbij zijn verschillende configuraties mogelijk die voor variërende nauwkeurigheid zorgen. Over de hele lijn zou dit tot een hogere precisie moeten leiden dan de eerdere, reeds geteste, GPS-oplossingen. Uit paragraaf 2.2.2 weten we immers dat er mogelijks een verschil bestaat tussen theoretische en praktische accuraatheid.

Ordina heeft eerdere ervaringen met de Ardusimple simpleRTK2K, waardoor er verder gebouwd kan worden op eerdere bevindingen en ervaringen. Eerdere testen toonden voor de Ardusimple simpleRTK2B een accuraatheid van twee cm in stilstand. Bij beweging daalt de accuraatheid tot tussen de twee tot vijf meter, ook in sterk bebouwde gebieden. Dit betekent een duidelijke verbetering ten opzichte van de accuraatheid van de GoPro en de iPhone.

De opzettijd van de Ardusimple simpleRTK2B bedraagt ongeveer zes weken. Onder deze tijd valt de integratie met een Arduino of Raspberry pi, het bouwen van een behuizing om de hardware te beschermen tegen slechte weersomstandigheden en het synchroniseren van de interne klokken in de verschillende hardwarecomponenten (camera en Ardusimple simpleRTK2B).

De interne klokken dienen tot op milliseconde gelijk te lopen om succesvol de GPS-data met de juiste *frames* van een video te koppelen. De belangrijkheid van dit laatste kan als volgt aangetoond worden: Indien in een mogelijke uitrol een snelheid van 20 km/u wordt gereden (het voertuig verplaatst zich elke seconde 5.5 meter) dan verhoogt de in-accuraatheid van de locatie met 5.5 meter per seconde dat de interne klokken van de GPS en de camera verschillen ten opzichte van elkaar. Tel hierbij de in-accuraatheid die inherent is aan de GPS, waardoor matching met de juiste 10 bij 10 meter cel praktisch niet meer mogelijk is.

Gezien de lang durende opzettijd, is er besloten de Ardusimple simpleRTK2B niet zelf aan te kopen en niet te testen voor de POC. Er wordt uitgegaan van de cijfers uit eerder opgedane ervaring. Twee tot vijf meter accuraatheid wordt beschouwd als de maximale precisie die met moderne, beschikbare technologie behaald kan worden.

2.2.2.3 Europees Galileoproject

Voor huidige civiele toepassingen is de accuraatheid van GPS dus beperkt tot twee à vijf meter, zelfs mits gebruik van meer geavanceerde systemen zoals RTK GPS.

De European Space Agency (ESA) begon in 2016 met de uitrol van Galileo, een wereldwijd satelliet navigatiesysteem (Global Navigation Satellite System, GNSS) dat gecreëerd werd om Europese landen onafhankelijk te maken van het Amerikaanse GPS systeem of het Russische GLONASS systeem. Met Galileo wil Europa niet enkel onafhankelijkheid nastreven, maar ook de bestaande systemen verbeteren. De kwaliteit en betrouwbaarheid van Galileo toont op heden een duidelijke verbetering in accuraatheid wanneer vergeleken met GPS. EUSPA (EU Agency for the space programme) belooft een accuraatheid van 10cm en een weerstand tegen interferentie¹¹. Galileo zal net als de GPS, uitzenden op twee draaggolven met als verschil dat ze beschikbaar zullen zijn voor civiele toepassingen. Hoewel Galileo is uitgerold in 2016, is het systeem nog niet volledig operationeel. Normaliter was dit gepland in 2020 met de lancering van de resterende satellieten, maar deze is uitgesteld naar 2021¹². Op het moment van schrijven (juni 2021) is dit nog niet gebeurd.

Voor de OVAM kan het interessant zijn om, zodra beschikbaar, gebruik te maken van Galileo in plaats van GPS om zo een nog nauwkeuriger resultaat te bekomen.

2.2.3 Besluit

Uit bovenstaande is duidelijk geworden dat de GPS van de GoPro ontoereikend is om een genomen afbeelding te koppelen aan een 10 bij 10 meter cel. Dit geldt voornamelijk in bebouwde gebieden, waar de fout van de GPS kan oplopen tot 30 meter. De ArduSimple simpleRTK2B kan deze fout op het signaal reduceren naar twee tot vijf meter. Deze nauwkeurigheid kan beschouwd worden als de meest precieze met huidige, beschikbare technologieën en zou voldoende moeten zijn om zwerfvuil te linken aan de 10 bij 10 meter cellen, al zal hier een standaardfout op zitten. De OVAM kan mogelijks ook afstappen van het gebruik van 10 bij 10 meter cellen. Deze mogelijkheid wordt intern door de OVAM verder bekeken. Het implementeren van RTK-toestellen in de POC valt buiten de scope van deze opdracht. Dit zou ons immers afleiden van de essentie: "Kan een AI-systeem ingezet worden om zwerfvuil te detecteren?"

Toekomstige technologieën zoals het Galileo project beloven nauwkeurige locatie bepalingen. Dit project zou in 2021 volledig operationeel zijn. Tijdens dit onderzoek bleek gebruik van deze technologie moeilijk door beperkte beschikbaarheid van hardware.

De grootste uitdaging van de GPS nauwkeurigheid ligt hem echter **niet** in de matching met de 10 bij 10 meter cellen, maar in het voorkomen van dubbeltellingen door het gebruik van video's (paragraaf 2.1.3). Deze oplossing vereist immers zeer precieze locatiebepaling, wat zelfs de ArduSimple simpleRTK2B niet aanbiedt. Waarom een accuraat GPS signaal het probleem van dubbeltelling in video's kan voorkomen wordt later besproken in luik 2.4.1. Bijgevolg is het aangeraden voor de OVAM om de nauwkeurigheid van de in theorie preciezere Galileo te toetsen zodra dit mogelijk is.

¹¹ <https://www.euspa.europa.eu/european-gnss/galileo/faq#value>

¹² https://www.spacedaily.com/reports/Soyuz_launch_from_Kourou_postponed_until_2021_2_others_to_proceed_999.html

2.3 ZWERFVUIL DETECTIE EN CLASSIFICATIE (AI)

Een belangrijk deel van de studie is het in kaart brengen van de huidige mogelijkheden van AI en de te verwachten moeilijkheden bij een uitrol over Vlaanderen. Dit luik bestudeert deze twee vraagstukken en tracht een antwoord te bieden op de vraag: is het mogelijk is om met de huidige technologie zwerfvuilobjecten te classificeren?

2.3.1 Computer vision

Voor het automatisch detecteren van objecten in foto's of video's bevindt men zich in het domein dat *computer vision* wordt genoemd. Binnen *computer vision* kunnen twee grote applicaties onderscheiden worden:

- **Afbeelding classificatie:** Het classificeren van afbeeldingen.
- **Object detectie:** Het detecteren van objecten op afbeeldingen.

2.3.1.1 Afbeelding classificatie

Men bevindt zich in het domein van het classificeren van afbeeldingen als er één typerend object op de afbeelding staat waarvoor het model een waarschijnlijkheid zal meegeven van wat het object kan zijn, bv. hond of kat. De uitkomst van het model beperkt zich dus louter tot een classificatie. Het AI model zal voor alle categorieën waar het op getraind is, een waarschijnlijkheid als uitkomst geven.

Een simpel voorbeeld kan dit verduidelijken. Een AI model wordt getraind om afbeeldingen te classificeren als hond of kat. Om dit te bekomen verwacht het model als invoer afbeeldingen met hun classificatie. Dit zou bijvoorbeeld een lijst kunnen zijn van foto's met volgende namen: 001_hond.jpg, 002_hond.jpg, 003_kat.jpg, Als uitkomst zal het model voor elke nieuwe afbeelding voor elke categorie een waarschijnlijkheid geven. Voor Figuur 8 zou het resultaat van het model, dat enkel getraind is op hond en kat, de volgende kunnen zijn: [0.95, 0.03]. Hierbij is het eerste getal de waarschijnlijkheid dat het om een hond gaat en de tweede de waarschijnlijkheid dat de afbeelding een kat bevat. De hoogste waarschijnlijkheid bepaalt de classificatie. Hierbij is het belangrijk te melden dat in de meeste gevallen wordt gewerkt met een drempelwaarde waarboven de waarschijnlijkheid ten minste moet liggen. De datawetenschapper kan deze parameter zelf bepalen. Indien de drempelwaarde in dit voorbeeld op 0.8 zou liggen, geeft het model als uitkomst aan dat afbeelding 100.jpg een hond afbeelding is.

Figuur 8: Voorbeeld van classificatie, één typerend object op de foto¹³



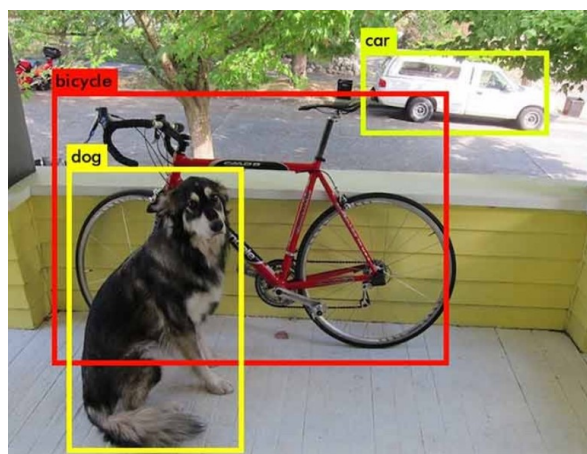
¹³ https://www.flickr.com/photos/giant_schnauzer/20081037129/sizes/l/

2.3.1.2 Object detectie

Zodra er meerdere objecten zijn in één afbeelding kan classificatie niet meer gebruikt worden. Een voorbeeld hiervan kan teruggevonden worden in Figuur 9. Het AI model kan in dit geval de afbeelding onmogelijk classificeren als ofwel “een hond”, ofwel “een fiets”, ofwel “een auto”. Al deze klassen zijn immers tegelijk aanwezig op de afbeelding. In deze situatie moet men een stap verder gaan dan de classificatie alleen. Men dient niet alleen objecten te detecteren, maar deze ook te lokaliseren op de afbeelding. Naar dit applicatiedomein wordt vaak verwezen als object detectie.

Object detectie gaat dus in eerste instantie verschillende objecten op de afbeelding detecteren om zo tot slechts één object te komen en vervolgens een classificatie uit te voeren. Indien het gevonden object ook geassocieerd kan worden zal het model zowel de locatie als de classificatie als uitkomst van het model teruggeven. Het beschrijven van de locatie van een object in een afbeelding gebeurt aan de hand van een *bounding box*. Een *bounding box* is niet meer dan een rechthoekige omkadering die rond een typerend object wordt getekend en aangeeft waar het object zich bevindt op de afbeelding. Een voorbeeld van een afbeelding verwerkt via een object detectie algoritme kan worden teruggevonden in Figuur 9. De kaders stellen de *bounding boxes* voor, terwijl de classificatie van elke *bounding box* terug te vinden is in de linkerbovenhoek van een *bounding box*.

Figuur 9: Voorbeeld Object detectie, meerdere typerende objecten op foto¹⁴



Het is belangrijk om te weten dat alvorens een object detectie model objecten succesvol kan aanduiden, het model reeds getraind moet zijn (net zoals bij de afbeelding classificatie). Training gebeurt door afbeeldingen met daarop de ware locatie van de objecten, aangegeven door *bounding boxes*, aan het model te voeden. Op deze manier kan het AI model leren hoe de objecten eruitzien om ze dan later op nieuwe afbeeldingen te kunnen detecteren. Afbeeldingen met *bounding boxes* rond de ware objecten worden ook wel *geannoteerde data* genoemd. Het vergaren van geannoteerde data is kostelijk en tijdsintensief. De data moet manueel “geannoteerd” worden, dwz. *bounding boxes* tekenen rond het object en aangeven bij welke klasse het object

¹⁴ <https://medium.com/@enriqueav/object-detection-with-yolo-implementations-and-how-to-use-them-5da928356035>

hoort. In dit onderzoek werd ook aan annotatie gedaan door Ordina en meer bepaald in de annotatie service tool die geïntegreerd is met azureML (paragraaf 2.5.3.3).

Object detectie laat ook toe objecten te tellen op afbeeldingen. Veronderstel even dat op Figuur 8 twee honden zouden staan. Bij afbeelding classificatie zou het AI model enkel ‘hond’ voorspellen zonder aan te geven hoeveel honden er op de afbeelding staan. Wordt er daarentegen een AI model gekozen uit het domein van de object detectie, dan zou het AI model eerst twee objecten detecteren en deze twee classificeren als elk een hond. De uitkomst van object detectie model ziet er dan ook anders uit dan deze van afbeelding classificatie modellen. Waar afbeelding classificatie enkel een kans teruggeeft voor elke klasse, zal *object detectie* naast deze waarschijnlijkheid voor elke klasse ook een *bounding box* als uitkomst geven. Deze *bounding box* wordt dan afhankelijk van het model gedefinieerd als twee hoekpunten, of als het middelpunt en een lengte en breedte.

Het is duidelijk dat het te ontwikkelen zwerfvuil detectiesysteem zich bevindt in het object detectie domein aangezien er meerdere vuilnisobjecten op een frame kunnen staan. In de rest van de studie worden zuivere classificatie algoritmen niet meer meegenomen.

2.3.2 Evaluatiemetrieken

Om te bepalen welke AI modellen de beste resultaten boeken, is het nuttig om eerst meer uitleg te geven over hoe AI-systemen worden geëvalueerd. Alleen zo kunnen de verschillende AI modellen onderling met elkaar vergeleken worden.

De bekendste metriek is accuraatheid. Deze geeft de graad van overeenstemming tussen een gemeten of berekende waarde en de daadwerkelijke waarde weer. Bv. duidt het AI-systeem twee afbeeldingen van honden aan als één afbeelding “hond” en één afbeelding “kat”, dan haalt het AI-systeem een accuraatheid van slechts 50%.

In het domein van object detectie voldoet accuraatheid alleen niet. Foute locaties van *bounding boxes* op afbeeldingen worden namelijk niet in rekening gebracht. Indien bijvoorbeeld op Figuur 9 het AI model een hond zou waarnemen op de locatie waar de auto staat, dan zou in de traditionele accuraatheid geen fout zijn gemaakt. Er werd immers nog steeds een hond gedetecteerd.

Een goede object detectie metriek penaliseert lokalisatie fouten. De meest gekende metriek bij object detectie is de *mean Average Precision* (mAP), die duidt op het gemiddelde van de *Average Precision* (AP; zie hieronder) over alle klassen heen. Dit betekent dat eerst voor elke klasse de AP wordt berekend. Bv. voor elke zwerfvuilfractie apart. Vervolgens wordt het gemiddelde genomen over alle klassen heen.

Om de AP beter te begrijpen moeten eerst een aantal basis metriekeken worden toegelicht waar de AP-berekening gebruik van maakt. Deze metriekeken worden aan de hand van een voorbeeld met de klasse ‘sigaret’ toegelicht:

- De *True Positives* (TP) worden bepaald door het aantal correct gedetecteerde objecten te tellen. In het voorbeeld met sigaretten gaat het dus om het totaal aantal sigaretten die correct als sigaret worden herkend door het AI model.
- De *False Positives* (FP) worden bekomen door het aantal fout geclassificeerde objecten te tellen. In het voorbeeld voor de klasse van sigaretten worden elementen die foutief als sigaret worden geclassificeerd geteld (bv. een takje wordt herkend als een sigaret).
- De *False Negatives* (FN) worden bekomen als het aantal werkelijke observaties van een bepaalde klasse die fout geclassificeerd worden. In het voorbeeld met sigaretten gaat het dus om alle sigaretten die anders geclassificeerd worden (bv. een sigaret wordt aanzien als een takje).
- De *True Negatives* (TN) geven tot slot het aantal voorwerpen die correct als ‘geen sigaret’ geïdentificeerd werden.

Het uitzetten van deze vier concepten geeft de *confusion matrix* weer (Figuur 10).

Figuur 10: *Confusion matrix*¹⁵

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Op basis van deze metriekeken wordt vervolgens de *Precision*¹⁶, de *Recall*¹⁷ en de *Intersection over Union* (IoU) berekend. De formules kunnen respectievelijk worden teruggevonden in Vergelijking 1,

Vergelijking 2 en

Vergelijking 3.

Vergelijking 1: *Precision*

¹⁵ <https://yeseulle0311.medium.com/pytorch-performance-evaluation-of-a-classification-model-confusion-matrix-fbec6f4e8d0>

¹⁶ De precision meet hoe zeker een model is in zijn voorspellingen, vb. haalt een sigaretten model een precision van 70% dan wilt dit zeggen dat 70% van de gevonden sigaretten effectief sigaretten zijn.

¹⁷ De recall meet hoe goed een model is in het vinden van de ‘te zoeken’ objecten, vb. haalt een sigaretten model een recall van 70% dan wilt dit zeggen dat 70% van alle sigaretten zijn gevonden in de foto’s.

$$Precision = \frac{TP}{TP + FP} \quad \text{met: } \begin{array}{l} TP = \text{True Positives} \\ FP = \text{False Positives} \end{array}$$

Vergelijking 2: Recall

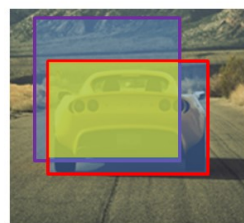
$$Recall = \frac{TP}{TP + FN} \quad \text{met: } \begin{array}{l} TP = \text{True Positives} \\ FN = \text{False Negatives} \end{array}$$

Vergelijking 3: IoU

$$IoU = \frac{\text{Area Of Overlap}}{\text{Area of Union}} \quad (\text{zie Figuur 11})$$

De IoU houdt geen rekening met de eigenlijke classificatie maar is een metriek die de nauwkeurigheid van de lokalisatie van het object vertegenwoordigt. De IoU wordt bekomen door de verhouding van de “Area of overlap” en de “Area of union”. Dit concept wordt verduidelijkt in Figuur 11. De Area of overlap stelt hier de oppervlakte voor die overlapt tussen de locatie van het object (de werkelijke *bounding box* aangeduid met een rode contour) en de gedetecteerde locatie van het object (de door het model voorspelde *bounding box* aangeduid met een paarse contour). De Area of overlap is geel gearceerd. De Area of union is gearceerd in het blauw en stelt dus de oppervlakte voor van beide *bounding boxes*. De IoU zal dus steeds een getal zijn dat zich bevindt tussen 0 en 1. Hoe hoger de IoU, hoe groter de overlap tussen de voorspelde *bounding box* en de werkelijke *bounding box*, hoe beter de lokalisatie gebeurd is.

Figuur 11: IoU verduidelijking¹⁸



Intersection over union (IoU)

$$= \frac{\text{size of } \text{[yellow box]}}{\text{size of } \text{[blue box]}}$$

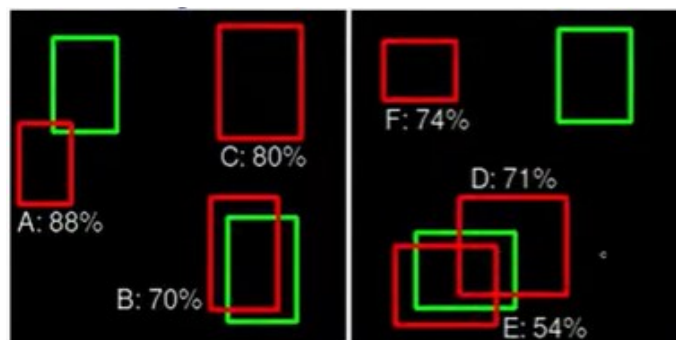
Om de berekening van *Average Precision* verder te verduidelijken wordt opnieuw gebruik gemaakt van het sigarettenvoorbeeld.

Beschouw Figuur 12. De rode *bounding boxes* zijn de plaatsen waar het model denkt een sigaret te hebben gevonden (lokalisatie). Elke box bevat tevens een percentage, welke de waarschijnlijkheid uitdrukt dat het gevonden object ook daadwerkelijk een sigaret is (classificatie). Naar deze percentages wordt verder gerefereerd als de “fractiekansen”. De groene boxes representeren de echte locaties van de sigaretten. De eerste stap in de berekening is het aantal TP’s en FP’s bepalen.

¹⁸ <http://datahacker.rs/deep-learning-intersection-over-union/>

Voor de bepaling van de metrieken in de context van object detectie wordt gebruik gemaakt van de IoU. Een *True Positive* moet immers een correcte classificatie hebben, maar het moet ook over de juiste locatie op de afbeelding gaan. Het classificatieprobleem is voor elke *bounding box* echter automatisch voldaan in dit voorbeeld, daar we enkel de *bounding boxes* gebruiken die overeenstemmen met een sigaret classificatie. Om te bepalen of de lokalisatie ook nauwkeurig genoeg is, wordt een IoU-drempelwaarde ("*cut-off*") bepaald. De meest populaire *cut-off* is 50%. Deze resulteert in de meeste TP's en dus de hoogste mAP (zoals zal blijken uit de uitleg hieronder). Wanneer de *cut-off* verhoogt, wordt het aantal TP's kleiner aangezien de overlap (IoU) tussen de voorspelde *bounding box* en de echte *bounding box* groter moet zijn. Dit resulteert in lagere mAP scores. Samengevat betekent dit dat bij een hogere *cut-off*, een betere lokalisatie verwacht wordt. In de praktijk worden vaak *cut-off* waarden gebruikt van 50% (AP50) en 75% (AP75).

Figuur 12: Grafisch voorbeeld van de berekening van de mAP-metrik¹⁹



In dit voorbeeld wordt een IoU-*cut-off* van 0.5 genomen. Visueel kan men uit Figuur 12 snel vaststellen dat B en E *True Positives* zijn. Stel er wordt een IoU-*cut-off* van 0.95 genomen dan zouden er geen TP's zijn, want de *bounding boxes* van B en E overlappen niet genoeg met de ware *bounding box*.

Merk op, B en E zijn enkel *True Positives* voor de klasse van sigaretten. Indien er meerdere klassen beschouwd worden, moet men deze oefening maken voor elke klasse apart. Voor elke klasse wordt dan enkel de *bounding boxes* aanschouwd waarvan de classificatie ook die klasse weergeeft.

In Tabel 5, wordt een voorbeeld uitgewerkt dat zich baseert op Figuur 12. Alle gedetecteerde *bounding boxes* worden op basis van hun fractiekans geordend van groot naar klein. Predictie A heeft de hoogste kans en wordt daarom eerst opgenomen in de tabel. De predictie A werd geclassificeerd als een FP. De geaccumuleerde TP in de tabel is daarom 0 en de geaccumuleerde FP komt op 1.

De geaccumuleerde *Precision*, of kortweg *Precision*, kan dan berekend worden op basis van Vergelijking 1. Ingevuld geeft dat: $\frac{0}{0+1} = 0$. Analoog kan de geaccumuleerde *Recall*, kortweg *Recall*, berekend worden volgens

¹⁹ <https://www.programmingsought.com/article/7948888875/>

Vergelijking 2. Ingevuld levert dit: $\frac{0}{4} = 0$. Merk op dat de noemer bij *Recall* een constante is. Door dezelfde berekeningen toe te passen voor alle predicties bekomt men de resultaten in Tabel 5.

Tabel 5: Berekening mAP

	Fractie kansen	Geaccumuleerde TP	Geaccumuleerde FP	<i>Precision</i>	<i>Recall</i>
A	88%	0	1	0 (=0/1)	0 (=0/4)
C	80%	0	2	0 (=0/2)	0
F	74%	0	3	0	0
D	71%	0	4	0	0
B	70%	1	4	0.20 (=1/5)	0.25 (=1/4)
E	54%	2	4	0.33 (=2/6)	0.5 (=2/4)

Uit bovenstaande tabel kan men afleiden dat het om een weinig getraind model gaat. De *Precision* uitzetten t.o.v. de *Recall* geeft immers een positief verband (als *Recall* stijgt, stijgt *Precision*). Voorspellingen met de hoogste fractie kansen blijken immers allen FP's te zijn. Bij een goed getraind model wordt veelal een negatief verband gevonden (als *Recall* stijgt, daalt *Precision*).

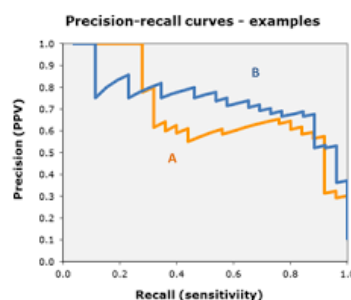
Gevoelsmatig kan dit aangetoond worden met een simpel voorbeeld gebruikmakende van verschillende merken frisdrank. Bij het opvragen van alle merken zullen immers eerst de meest evidente, algemeen gekende, frisdranken opgenoemd worden zoals een Coca Cola, Pepsi cola, Fanta, etc. De fractiescore van deze soorten zal dus hoog zijn, net zoals het aantal *True Positives*. De genoemde frisdranken zullen veelal daadwerkelijk merken frisdrank zijn waardoor het aantal *False Positives* laag zal zijn en de *Precision* hoog is. Naarmate de lijst met merken vordert, worden minder gekende merken frisdrank genoemd, zoals bv. Dr. Pepper in Europa. De zekerheid waarmee een merk genoemd wordt, neemt stelselmatig af. De kans op fouten neemt toe, waardoor de *False Positives* oplopen en de *Precision* daalt. De *Recall* zal gedurende de hele opsomming stijgen. De *Recall* wordt immers berekend als de verhouding van het aantal correct genoemde merken ten opzichte van het totaal aantal merken dat bestaat.

Uit Tabel 5 kan de AP berekend worden. Eerst worden de *Precision* waarden bepaald die overeenstemmen met de *Recall* waarden uit volgende lijst [0;0.1;0.2;0.3; ... ;1]. In dit voorbeeld moet dit gebeuren door een extrapolatie tussen de *Recall* waarden. In realiteit zal dit probleem echter veel kleiner zijn. Er zullen immers veel meer waarden voor handen zijn, door een grote hoeveelheid aan afbeeldingen.

Als alternatief kunnen deze *Precision* waarden op grafische manier bekomen worden. Eerst wordt een scatterplot gemaakt waarin de beschikbare combinaties *Precision* en *Recall* worden uitgezet. De bekomen punten worden vervolgens verbonden om een lijngrafiek te bekomen. Op de grafiek kan nu de *Precision* worden afgelezen voor elke 0.1-*Recall* waarde. Deze methodologie wordt geïllustreerd in

Figuur 13. Op deze manier wordt hetzelfde resultaat bekomen als via de mathematische methode, maar dan visueel.

Figuur 13: Grafische voorstelling *Precision/Recall*²⁰



Door toepassing van Vergelijking 4 kan men vervolgens de AP berekenen voor deze klasse. Het is eigenlijk niets meer dan het gemiddelde van de 11 bekomen *Precision* waarden.

Vergelijking 4: AP

$$AP = \frac{1}{11} \sum_{recall_i} precision(recall_i)$$

Idealiter wil men een model dat voor elke *Recall* waarde goed presteert: zelfs bij minder zekere voorspellingen, juist zijn. Grafisch vertaalt zich dit in een grafiek die richting de rechterbovenhoek leunt in de uitzetting zoals in

Figuur 13.

We sluiten af met een samenvatting van de mAP. De mAP is een populaire metriek die de voorspelde plaatsen van een object vergelijkt met de echte plaatsen van een object in een foto aan de hand van een vooraf bepaalde IoU *cut-off*. Hierin wordt rekening gehouden met zowel de *Precision* en *Recall*, welke de uiteindelijke accuraatheid bepalen. Hoe hoger de mAP hoe beter het model de objecten lokaliseert en detecteert.

2.3.3 Algoritmes: Object detectie netwerken

Alvorens een antwoord kan geformuleerd worden op de onderzoeksvragen: “Zijn er voorgetrainde zwerfvuildetectie modellen aanwezig?” en “Is het met de huidige technologie mogelijk zwerfvuil te detecteren?”, moet er gekeken worden naar het type algoritmes die binnen object detectie bestaan. Het is belangrijk om deze te onderzoeken met hun voor- en nadelen alsook hun prestaties op bekende benchmarks.

Er zijn twee grote groepen binnen object detectie, met name **one-shot modellen** en **double-shot modellen**. Het volstaat om te weten dat double-shot modellen een extra architecturale stap doorlopen, namelijk het ‘*region*

²⁰ <https://acutecaretesting.org/en/articles/precision-recall-curves-what-are-they-and-how-are-they-used>

proposal network'. Deze extra stap maakt double-shot modellen trager, maar performanter dan de one-shot modellen. One-shot methoden vinden vooral hun toepassing in real-time image *processing*. Zo gebruikt BMW een single-shot model (Yolov4) voor hun AI-systemen. Er is dus een trade-off tussen de snelheid en de performantie.

In de literatuur worden methoden aangegeven die de twee types incorporeren door een voorspelling te maken over de complexiteit van een afbeelding (Soviany, 2018). 'Makkelijke' afbeeldingen worden gestuurd door een one-shot model en 'moeilijke' afbeeldingen door een double-shot model.

Naast one-shot en double-shot modellen zijn er ook **Siamese netwerken**. Deze netwerken vergelijken objecten op basis van standaard referentieklassen. Een gedetecteerd zwervuilobject, voortvloeiend uit een systeem dat louter zwervuil herkent ("zwervuil/geen zwervuil"), zou kunnen worden vergeleken met een database bestaande uit standaardfoto's van alle te detecteren zwervuilfracties. Dit model wordt niet verder onderzocht aangezien er weinig toepassingen beschikbaar zijn in de literatuur en deze optie beperkt toepasbaar lijkt door onder andere hoge computatie eisen: elk gedetecteerd zwervuilobject moet één op één vergeleken worden met een database vol referentie foto's. In een goed werkend systeem zou elke fractie bijgevolg honderden referentie foto's moeten hebben om elke mogelijke situatie in welke deze fractie zou kunnen voorkomen, voor te stellen. In onze situatie betekent dit 26 verschillende fracties met 100 referentie foto's per fractie die leiden tot een één op één vergelijking van een gedetecteerd zwervuilobject met 2600 referentiefoto's.

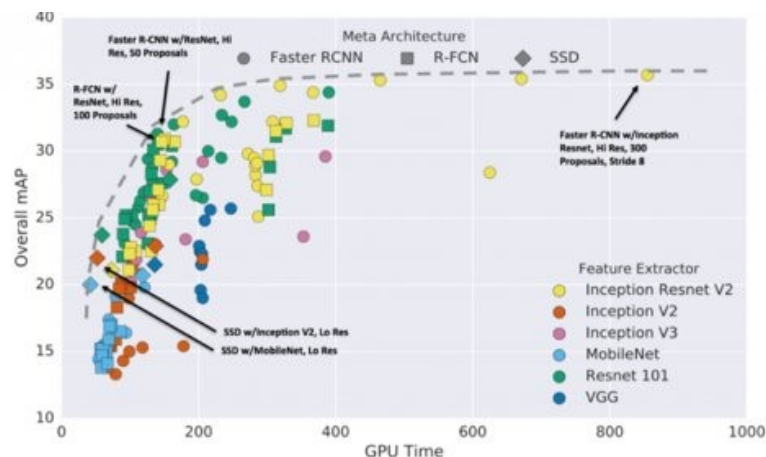
Enkel de prestaties van de one-shot en double-shot modellen en hun prestaties worden verder bekeken. De twee bekendste benchmarks zijn de COCO en PASCAL VOC die zowel de mAP als de snelheid van het model scoren.

De **COCO-benchmark** is een dataset met 81 verschillende klassen waaronder schapen, auto's, personen, etc. De bedoeling van de benchmark is om in een geringe trainingstijd een goede mAP score te behalen.

Figuur 14 toont de trade-off tussen de accuraatheid (two-shot) en de snelheid (one-shot) op de COCO-benchmark. Faster RCNN (bol), R-FCN (vierkant) zijn two-shot modellen. SSD's (ruit) zijn one-shot modellen.

Double-shot modellen scores beduidend beter dan one-shot modellen in termen van mAP. One-shot modellen doen het beter in termen van snelheid.

Figuur 14: COCO-benchmark resultaten²¹



Men mag zich niet laten misleiden door de lage scores op deze benchmark. Dit komt door het generieke karakter dat wordt gescoord door de benchmark en twee specifieke eigenschappen van de berekening van de mAP. De modellen dienen namelijk als startpunt voor een rij uiteenlopende taken. Ze zijn niet getraind om in een bepaalde taak zo goed mogelijk te worden (zoals detectie van zwerfvuil) maar vormen eerder de baseline voor andere modellen die dit trachten te bereiken.

De eerste eigenschap van de berekening die voortvloeit uit het generieke karakter is dat voor de berekening van de mAP op de COCO-benchmark een gemiddelde wordt genomen van de prestaties op 10 IoU waarden, waarbij heel hoge *cut-offs* zitten. Dit wil zeggen dat voor de meest intolerante IoU waarde (0,95) de voorspelde *bounding*

²¹ <https://jonathan-hui.medium.com/object-detection-speed-and-accuracy-comparison-faster-r-cnn-r-fcn-ssd-and-yolo-5425656ae359>

box 95% moet overlappen met de echte *bounding box*. Deze benchmark belooft dan ook modellen met heel goede lokalisatie. In de praktijk wordt er één, lagere IoU-waarde genomen en wordt hier het beste model op gemaakt. Ten tweede zijn er 81 ongebalanceerde fracties aanwezig in termen van hoeveelheid trainingsdata, bv. de meest gerepresenteerde klasse (persoon) heeft 10.000 training voorbeelden, terwijl de minste (toaster) er maar negen heeft. Het spreekt voor zich dat het model meer kan leren over een persoon dan over een toaster. Dit zorgt ervoor dat het model meer moeite heeft met het behalen van goede scores in het detecteren van toasters dan voor het detecteren van personen. De lage mAP scores op deze COCO-benchmark kunnen dus verklaard worden door het feit dat de onder-gerepresenteerde klassen evenveel meetellen in de uiteindelijke berekening van de globale mAP als de persoon klasse. Het opnemen van klassen in de COCO-benchmark die weinig data ter beschikking hebben, peilt naar het vermogen van een type model om met zo weinig mogelijk data toch goede scores te behalen.

Het verschil tussen de one-shot en double-shot modellen wordt minder duidelijk op de PASCAL VOC-benchmark die met slechts 20 klassen werkt en dus gelijkaardig is aan het aantal geïdentificeerde zwerfvuilfracties die de OVAM wenst te monitoren.

Zoals men kan zien in Figuur 15 zijn de prestaties van de one-shot modellen even goed als de double-shot waardoor one-shot modellen een aantrekkelijk alternatief worden door de snellere GPU-tijden. Het aantal fracties heeft duidelijk een significante invloed op de prestatie van object detectie algoritmen. Op basis hiervan zal er voor dit project meer gedoeld worden op mAP scores met een grootteorde van 70% - 80%.

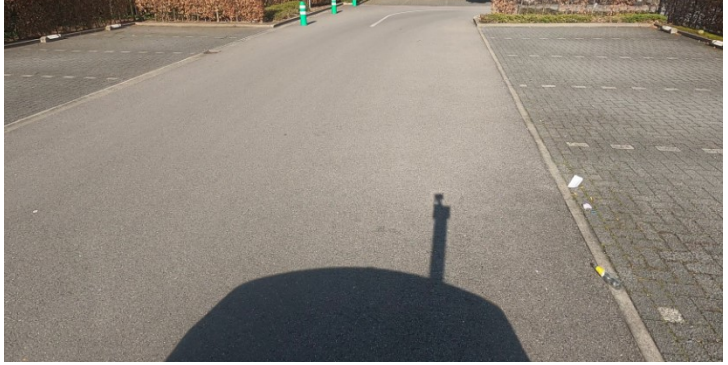
Figuur 15: PASCAL VOC benchmark resultaten²²

Method	Backbone	mAP (%)	
		VOC 2007 test	VOC 2012 test
<i>two-stage:</i>			
Faster R-CNN[36]	VGG-16	78.8	75.9
OHEM++[41]	VGG-16	-	80.1
R-FCN[5]	ResNet-101	83.6	82.0
<i>one-stage:</i>			
SSD300[30]	VGG-16	81.2	79.3
SSD512[30]	VGG-16	83.2	82.2
RON384++[24]	VGG-16	81.3	80.7

Er is echter nog een tweede factor met een grote invloed: de complexiteit van de foto's. De aangehaalde benchmarks werken namelijk met foto's waar de objecten duidelijk zichtbaar zijn. De objecten uit de foto's in de praktijk en specifiek voor onze probleemstelling zullen dat vaak niet zijn (Figuur 16). De linkse afbeelding is een voorbeeld uit de test set van de COCO-benchmark. De rechtse afbeelding komt uit een proefopstelling die een realistisch beeld geeft over het type foto's die later in de POC en uitrol zullen vergaard worden.

Figuur 16: Links een foto uit de COCO-benchmark en rechts een foto uit een proefopstelling

²² https://www.researchgate.net/figure/Detection-results-on-PASCAL-VOC-dataset-All-models-are-pre-trained-on-MS-COCO-and_tbl2_321180719



Bovenstaande maakt duidelijk dat er veel modellen bestaan die in aanmerking komen voor de detectie van zwerfvuil. Op basis van prestaties, ook wel **performantie** genoemd, en de **specificiteit**, wordt getracht enkel de beste modellen te identificeren. Met specificiteit wordt er gedoeld naar modellen die reeds specifiek getraind zijn voor afvalherkenning en classificatie. Deze worden besproken in volgende paragraaf.

De eerste beslissingsfactor, de performantie, werd hierboven belicht en kan als volgt worden samengevat. De literatuur leert ons dat double-shot modellen performanter zijn dan one-shot modellen in termen van accuraatheid. Het omgekeerde geldt voor de snelheid. Uit de PASCAL VOC-benchmark, die meer bij dit project aanleunt, was deze betere accuraatheid echter niet significant, waardoor in samenspraak met de OVAM beslist werd om slechts één model te selecteren per type model. Binnen de groep van de double-shot modellen scoren de Faster RCNN en de Mask RCNN het hoogst. Voor de one-shot modellen zijn dit YOLO en SSD.

Het is niet mogelijk om one-shot modellen te scoren op het specificiteitscriterium, daar er slechts één publiek beschikbaar zwerfvuildetectie model uit deze categorie bestaat. Deze werd alsook slecht bevonden. Het model lijdt namelijk aan *overfitting*, een uitdaging van AI die later nog besproken wordt in paragraaf 2.3.4.2. Dit AI model gebruikt een YOLO v3 implementatie dat naast sigaretten ook vuilniszakken kan detecteren²³. Voor de one-shot modellen zal dus enkel de beslissingsfactor performantie geëvalueerd worden. Hierdoor kan men onmiddellijk besluiten dat het YOLO algoritme de beste optie is van de one-shot modellen. In de POC zal gebruik worden gemaakt van het recentste **YOLO v5 framework**. Meer informatie over dit framework kan in de literatuur gevonden worden²⁴.

De keuze tussen de Mask RCNN en de Faster RCNN (double-shot modellen) zal worden gemaakt op basis van de specificiteit. Dit wordt de volgende paragraaf belicht.

2.3.3.1 Specificiteit: Bestaande zwerfvuildetectie modellen

Modellen die al afbeeldingen met zwerfvuil hebben gezien geven ontwikkelaars het voordeel minder tijd te moeten spenderen aan het aanleren van basis karakteristieken van zwerfvuil. Dit concept, het gebruik van modellen die reeds voorgetraind zijn, wordt ook wel *transfer learning* genoemd. In *computer vision* is het gebruiken van *transfer learning* de norm geworden. Verschillende voorgetrainde modellen worden hieronder toegelicht en beoordeeld.

Het eerste voorgetrainde model is het **TACO onderzoek**. Dit is een objectdetectie- en classificatiesysteem van zwerfvuil die gebruik maakt van een Mask R-CNN algoritme. In de paper over TACO (Proença, 2020) zijn twee modellen ontwikkeld. Een model dat zwerfvuil kan herkennen (TACO_1) en een model dat een stap verder gaat en zwerfvuil classificeert in tien klassen (TACO_10). Na verschillende tests blijken de resultaten teleurstellend. TACO_1 haalt een mAP van 26.1% en TACO_10 maar een mAP van 19.4%.

Ordina testte TACO_10 door een aantal afbeeldingen te voeden aan het zwerfvuilmiddel. In Figuur 17 staan drie voorbeelden. Fracties zoals flessen en sigaretten kan het model goed detecteren. Dit komt omdat de

²³ GitHub - maartensukel/yolov3-garbage-object-detection-training

²⁴ <https://github.com/ultralytics/YOLO-v5>

trainingsdata van het TACO model vooral bestaat uit deze fracties. Daarnaast werd duidelijk dat het model meermaals grote patronen in de foto onterecht classificeert als zwerfvuil. Een mogelijke verklaring is dat de trainingsdata van TACO voornamelijk bestaat uit close-up foto's van objecten waar de verhouding pixels van gedetecteerde objecten tot pixels van de gehele foto groot is. Het is mogelijk dat het model heeft geleerd om grote terugkerende patronen (zoals het terras in Figuur 17) te herkennen als zwerfvuil. Dit kan verklaard worden door een observatie die ook in de TACO paper wordt gemaakt, waarbij verklaard wordt dat TACO onvoldoende presteert bij kleine objecten. Samengevat heeft het TACO model zeker tekortkomingen, maar het is in staat reeds enkele fracties goed te detecteren en kan als een mogelijk startpunt dienen.

Figuur 17: Test afbeeldingen gestuurd door het TACO_10 model

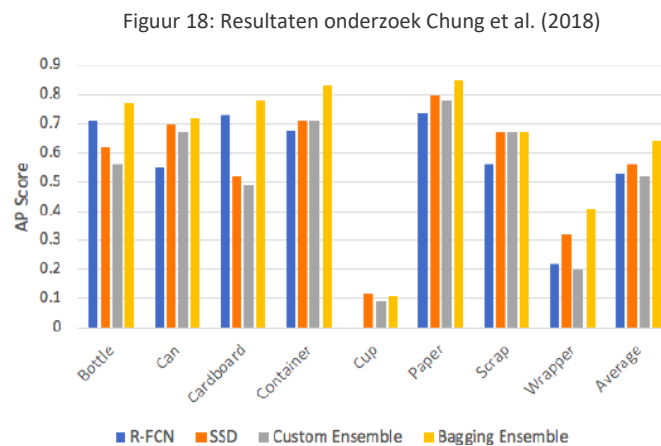


Een ander project (Rad et al., 2017) beperkt zich niet alleen tot de detectie van zwerfvuil maar ook tot de telling ervan. Hier wordt gebruik gemaakt van het Google's overfeat model. Zij vermijden dubbeltelling van bepaalde fracties door aan een constante snelheid foto's te nemen aan een vooraf ingestelde fps. Dit klinkt theoretisch veelbelovend maar is in de praktijk zoals eerder besproken moeilijk haalbaar. Denk maar aan voetgangers die oversteken of voertuigen die plots stoppen etc. In het onderzoek wordt beschreven dat er 25 verschillende fracties worden geclassificeerd, wat dicht aanleunt tegen de hoeveelheid fracties van dit project. Dit aantal hebben ze echter in hun analyse moeten reduceren door gebrek aan voldoende trainingsdata. Het uiteindelijke aantal werd teruggebracht naar drie fracties. De behaalde precisie hiervan varieert tussen de 60% en 75%. De focus van dit onderzoek lag voornamelijk in het detecteren van kleine voorwerpen zoals sigaretten en bladeren van op een hoogte van 3m met een fotoresolutie van 1920x1480. Dit doet vermoeden dat met een 2k camera de kleinere objecten ook waarneembaar zullen zijn voor dit project.

isaychris-litter²⁵ is een github project met een andere insteek. De bedoeling is om zwerfvuil te detecteren, maar niet te classificeren. In dit project wordt niets vermeld over de accuraatheid van het algoritme, wat doet vermoeden dat de prestatie niet aan de wensen voldeed.

²⁵ <https://github.com/isaychris/litter-detection-tensorflow>

Een ander onderzoek stelt een real-time zwerfvuildetectie framework voor dat gebruik maakt van ensemble-based machine learning models (Chung, 2018). Ze tonen onder andere aan dat Bagging (een ensemble methode) de performantie van het systeem significant verhoogt ten opzichte van stand-alone models. Een ensemble is een techniek in machine learning die meerdere modellen combineert om zo tot een beter resultaat te komen. Deze techniek is zeer populair om de accuraatheid te verhogen. Deze methode tast echter ook de snelheid aan aangezien de foto's door meerdere modellen moeten gaan. De resultaten van het onderzoek worden gevisualiseerd in Figuur 18. R-FCN en SSD zijn alleenstaande detectoren. De Bagging methode presteert gemiddeld duidelijk beter dan de andere methodes met een mAP van 63%. In de Bagging methode werden de *bounding boxes* van R-FCN en SSD vergeleken. Als de *bounding boxes* van verschillende modellen voor hetzelfde object een correlatie van 50% hebben ($IoU > 50\%$), wordt er een zwerfvuilobject geclassificeerd. Een zwerfvuilobject wordt niet meegerekend indien dit niet door alle modellen is gedetecteerd, tenzij het een zeer hoge kans toegewezen krijgt. De modellen zijn niet publiek en kunnen bijgevolg niet getest worden voor dit onderzoek.



Een mogelijks aangewezen aanpak is om het zwerfvuildetectie probleem op te splitsen in twee grote stappen: zwerfvuil detecteren, uitsnijden en dan classificeren. Deze methodologie werd ook toegepast door Chung met het Custom ensemble model (Figuur 18) (Chung, 2018). Na verschillende tests bleek deze werkwijze echter niet gunstig voor de prestaties, waardoor niet gekozen werd voor dit alternatief.

Tot slot bestaat het project van Let's Do It Foundation²⁶, dat gebruik maakt van hetzelfde algoritme als het TACO model, namelijk een MASK R-CNN. Dit model voor zwerfvuil detectie werd getest op dezelfde dataset als TACO (

²⁶ <https://github.com/letsdoitworld/wade-ai>

Figuur 19). Het Let's do it foundation-model presteerde goed ondanks het feit dat het model getraind is op hopen zwerfvuil in plaats van op individuele zwerfvuilobjecten. Wel moet vermeld worden dat dit model slechts één klasse in rekening brengt, zijnde zwerfvuil/geen zwerfvuil.

Figuur 19: Test afbeeldingen gestuurd door het Let's do it foundation model



In conclusie, zijn er twee openbare double-shot modellen die voldoen aan het vooropgestelde specificiteitscriterium, nl. het TACO-model en de Let's do it foundation-model.

Beide AI modellen hebben hun beperkingen. Echter in de POC zal het Let's do it foundation-model genomen worden als vertrekpunt voor het trainen aangezien dit empirisch gezien beter in staat was zwerfvuil te detecteren. Bijkomend heeft het TACO-model meer moeilijkheden met het detecteren van kleinere voorwerpen, een vereiste dat voor dit project uitermate belangrijk is. Meer info over de Mask R-CNN kan in de voetnoot gevonden worden²⁷.

2.3.3.2 Zwerfvuil datasets

Bovenstaande voorgetrainde zwerfvuildetectie modellen zijn getraind op gelabelde of geannoteerde data van zwerfvuil. Met oog op de POC is het goed deze verschillende datasets mee te nemen als data om op te trainen, wat voor een grote tijdsbesparing kan zorgen. Tabel 6 geeft een niet-exhaustieve overzicht van de datasets. In de POC zal beslist worden welke nuttig zijn (paragraaf 4.2.1).

Tabel 6: Geannoteerde open-source datasets

dataset	link
TACO	https://github.com/pedropro/TACO/tree/master/data
Let's do it foundation	https://drive.google.com/file/d/1X_ozEv5vF3bhg3FIU6_5suBC7UdVVtA/view
River plastic detection	https://github.com/colinvanlieshout/riverplasticdetection
isaychris	https://github.com/isaychris/litter-detection-tensorflow/tree/master/dataset

²⁷ https://github.com/matterport/Mask_RCNN

2.3.4 AI obstakels

Binnen het domein van AI, en zo ook in ons onderzoek, zijn er een aantal valkuilen die vermeden kunnen worden en uitdagingen die overwonnen moeten worden. De meest voorkomende, die tevens toegelicht zullen worden hieronder, zijn de nood aan annotatie, het risico op *overfitting* en de tijd nodig om te trainen.

2.3.4.1 De nood aan annotatie

Het concept annotatie werd reeds geïntroduceerd in paragraaf 2.3.1.2. Het annoteren van afbeeldingen d.m.v. het manueel aanduiden van *bounding boxes* is noodzakelijk voor de training van object detectie modellen. Deze trainingstechniek wordt ook wel *supervised learning* genoemd. Echter is er een alternatief genaamd *unsupervised learning*. Deze vereist geen geannoteerde trainingsdata (afbeeldingen zonder *bounding boxes*) en proberen door middel van bv. clustering groepen of patronen in de data te vinden. In onze toepassing zou bij input van een afbeelding van een sigaret, het systeem op zoek gaan naar gelijkaardige afbeeldingen. Indien men de resultaten vertrouwt, kan men de meest gelijkaardige objecten annoteren als zijnde de klasse 'sigaret'. Er is dus geen manuele annotatie nodig.

Ter verduidelijking lichten we de algemene werking van *unsupervised object detectie* algoritmen toe²⁸:

1. Een *query*²⁹ wordt afgevuurd naar een database. De *query* is in dit geval het game personage Mario.

Figuur 20: Eerste stap unsupervised object detectie²⁸

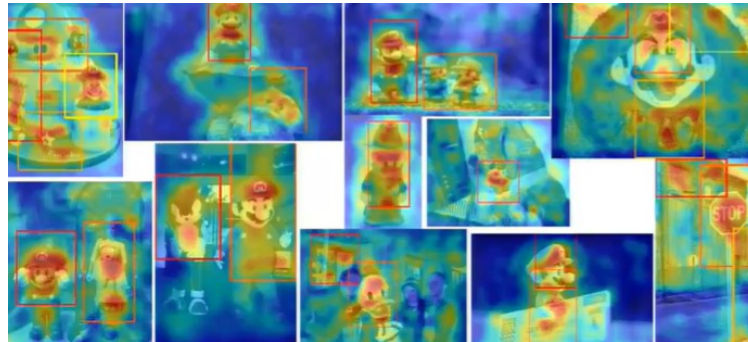


2. Een *discriminator* gaat sequentieel door de database. De *discriminator* zoekt de verzameling pixels die gelijkaardig zijn aan de *query*. De rode pixels tonen visueel de plaatsen aan waar er gelijkenissen zijn. Deze dienen dan als *bounding boxes*.

²⁸ <https://www.youtube.com/watch?v=BkmOHwKUCTA&t=149s>

²⁹ Een query is een opdracht die aan een database wordt gegeven om een bepaalde actie uit te voeren.

Figuur 21: Tweede stap unsupervised object detectie²⁸



3. Voor elke *bounding box* wordt de centraliteit berekend ten opzichte van het hele netwerk. Hoe hoger de score, hoe meer gelijkaardige *bounding boxes* in het netwerk zitten. Op basis daarvan kunnen *bounding boxes* uitgesloten worden die niet toepasselijk zijn voor deze *query*, zoals bijvoorbeeld de Donkey Kong met centraliteit van 0.2.

Figuur 22: Laatste stap unsupervised object detectie²⁸



Tot op heden wordt deze methodologie niet of zeer beperkt toegepast voor beeldherkenning daar het enkele risico's en nadelen met zich meebrengt. Ten eerste is deze oplossing computationeel intensief. De *query*-afbeeldingen moeten sequentieel vergeleken worden met elke afbeelding in de database en nadien moeten de gegenereerde *n* *bounding boxes* onderling vergeleken worden. Dit betekent $n \times (n-1)$ berekeningen. Wanneer dit voor 26 fracties moet gebeuren en er meerdere *query*'s per fractie (bv blikje fanta en cola) nodig kunnen zijn, dan loopt het aantal berekeningen heel snel op. Ten tweede kan de achtergrond het AI model negatief beïnvloeden. In bovenstaande uitleg en voorbeelden zouden de rode achtergronden kunnen meegenomen worden als Mario. Deze stijging van het aantal mogelijke Mario's verhoogt het aantal berekeningen nog meer. Tenslotte zijn de resultaten van dergelijke systemen minder goed dan die van supervised systemen.

Het antwoord op de vraag “**Is het mogelijk om unsupervised learning te gebruiken?**” luidt: nog niet. Deze technologie staat nog niet op punt. Voor eenvoudigere taken zoals in het voorbeeld met Mario, zou *unsupervised learning* wel kunnen werken. Voor dit project, waar de complexiteit een pak hoger ligt, is het niet haalbaar. Het researchgebied is echter wel in volle ontwikkeling.

2.3.4.2 Overfitting

Bij het maken van het model moet er rekening gehouden worden met het risico op *overfitting*. *Overfitting* vindt plaats als het model als het ware te hard geleerd heeft op zijn trainingsdata en niet goed generaliseert waardoor het slecht presteert op nieuwe data.

Er zijn verschillende manieren om *overfitting* tegen te gaan, waarvan twee toegepast worden in het kader van dit project. Eén techniek is de train/validatie/test-split waarbij men vooral de mate van *overfitting* gaat meten dan het echt tegen te gaan. Hierbij wordt de dataset opgedeeld in drie delen waarvan enkel het eerste deel (training set) gebruikt wordt om het model te leren of trainen. De validatie en test set zijn er om respectievelijk *hyperparameters* te *tunen* en voor model testing. *Hyperparameters* zijn aanpasbare parameters waarmee het trainingsproces van het model wordt beheerd. Een voorbeeld van zo’n *hyperparameter* die bij elk neurale netwerk gedefinieerd moet worden is het aantal verschillende lagen in het AI model. Model prestaties zijn sterk afhankelijk van *Hyperparameters*. In de test set zit vervolgens data waarop het getrainde model niet geleerd heeft. Er is *overfitting* als de resultaten van deze test en validatie set slechter zijn dan de resultaten op de trainingsset. Meestal wordt een 70/15/15-split gehanteerd, maar hier kan van afgeweken worden (de trainingsset zal altijd groter zijn dan validatie en test samen).

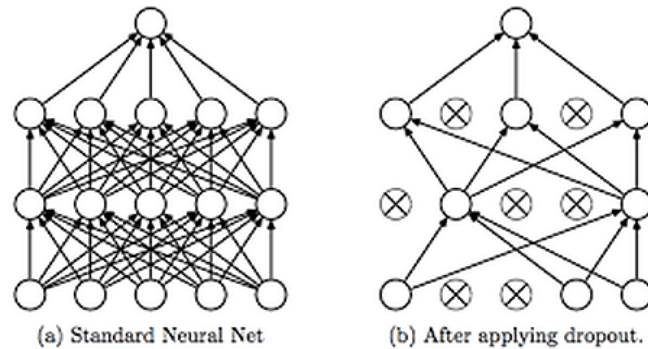
Reeds getrainde modellen, die niet overfit zijn, kunnen een mechanisme zijn tegen *overfitting*. Hierbij worden de gewichten van de eerste lagen van een voorgetraind model vastgezet. Zo moeten enkel de detail lagen getraind worden, wat heel vaak sneller tot goede resultaten leidt en voor betere generalisatie zorgt.

Indien het model daadwerkelijk overfit is, kan er gekeken worden naar de complexiteit van het model. Hoe hoger de complexiteit, hoe hoger de kans op *overfitting*. Een eerste methode is het verlagen van het aantal lagen in het model. Een andere methode is *dropout*, waarbij er willekeurige *variabelen* uitgeschakeld worden gedurende een bepaalde termijn in het trainingsproces om zo minder afhankelijkheid van dominante feature door te voeren en het ‘overleren’ tegen te gaan. In

Figuur 23, staat een representatie van het concept *dropout*.

Er zijn nog andere methodes zoals early stopping, L1 en L2 regularisatie. In de meeste gevallen zullen de train/validatie/test-split en het gebruik van voorgetrainde modellen voldoende zijn. De gepaste oplossingsmethoden zullen worden gekozen aan de hand van de problemen die zich voordoen tijdens de PoC.

Figuur 23: Een voorbeeld van overfitting³⁰



2.3.4.2.1 Trainingstijd

Een ander probleem binnen AI is de trainingstijd. Zware AI modellen hebben typisch meerdere dagen of zelfs weken tijd nodig voor hun berekeningen en om tot een model te komen. Voor de POC is dit door de beperkte beschikbare tijd een groter probleem dan tijdens de uitrol. Bij de uitrol kan een eerdere modelversie gebruikt worden tot het nieuwe model operationeel is. In de POC fase moet nog een eerste model gecreëerd en gevalideerd worden. Dit maakt dat deze variabele een cruciale invloed heeft op de POC fase en moet gecontroleerd worden.

Er zijn vier factoren die een invloed hebben op de trainingstijd:

1. het type model
2. de gebruikte hardware
3. het aantal cycli
4. hoeveelheid trainingsdata

Het type model en de hardware waarop getraind wordt, kunnen gecontroleerd worden door andere keuzes te maken om zo de trainingstijd beperken. Hierbij werd de keuze voor het type model reeds in een eerder hoofdstuk gemaakt. De trainingstijd per cyclus van het YOLO v5-model (one-shot) zou theoretisch gezien sneller moeten zijn dan het Mask R-CNN model (two-shot). Doordat gebruik wordt gemaakt van een voorgetraind model op zwerfvuil heeft de Mask R-CNN al meerdere cycli doorlopen en is het dus mogelijk dat de totale trainingstijd voor dit model lager gaat liggen.

Voor het aantal cycli en de hoeveelheid (kwalitatieve) trainingsdata geldt de regel “hoe meer van deze parameter, hoe beter de resultaten”. Een beperking in deze factoren toepassen, zorgt voor een verslechtering van de resultaten. Het aantal cycli wordt in het vakjargon uitgedrukt als *epochs*. Eén *epoch* is het proces waarbij de gehele trainingsset door het model gaat en de gewichten worden geüpdatet.

³⁰ <https://www.kdnuggets.com/2019/12/5-techniques-prevent-overfitting-neural-networks.html>

Tenslotte rest ons de hardware. Er zijn verschillende opties om het model op te trainen: op de eigen laptop CPU of GPU, op een Cloud CPU of GPU of een apart aangekochte GPU. Trainen op een GPU heeft als voordeel dat meerdere afbeeldingen in parallel kunnen verwerkt worden in tegenstelling tot de CPU waar dit sequentieel gebeurt. Echter is de aankoop van een losstaande GPU die gekoppeld kan worden aan een computer relatief duur en kan men dit niet opschalen, wat in de Cloud wel kan. De GPU's die standaard in computers zitten, voldoen in de meeste gevallen niet bij deze algoritmes. Ze zijn eerder bedoeld voor de grafische noden bij bv. games, CAD toepassingen, etc. Om deze reden werden bovenstaande opties uitgesloten en enkel de trainingstijden van drie opties getest op een publiek beschikbare dataset³¹ met 131 afbeeldingen van kangoeroes in 2K resolutie. Het gebruikte model is een Mask R-CNN. De resultaten staan op Figuur 24.

Figuur 24: Resultaten trainingstijden op kangoeroe dataset

	LOKAAL	CLOUD	
Processor	CPU	CPU	GPU
Trainings tijd per epoch (min)	45	20	4
Kosten (euro/uur)	gratis	0,15	1,15

Deze resultaten worden geëxtrapoleerd naar een realistisch voorbeeld zodoende een inschatting te maken over de totale trainingstijd die een model zal nodig hebben. Alexey, de maker van het veelgebruikte YOLO model³², beschrijft de nood aan minstens 2.000 trainingsvoorbeelden (*bounding boxes*) per fractie. Voor de eenvoud wordt er uitgegaan dat er één zwerfvuilitem (één *bounding box*) op één foto staat. Dit brengt het totaal trainingsafbeeldingen op 54.000. Indien er 200 *epochs* wordt getraind op 54.000 2k afbeeldingen dan duurt de training op een Cloud GPU +/- 8,48 dagen. Op een CPU in de Cloud duurt dit proces +/- 42,41 dagen. Dit proces zal op een eigen laptops CPU +/- 93,3 dagen duren. Het gebruik van een GPU bij de POC is essentieel om voldoende ruimte te laten voor modelvalidatie en eventueel nog een aantal trainingssessies op nieuwe data. Voor de uitrol kan, rekening houdend met de kosten, gebruik gemaakt worden van een CPU. De trainingstijd is hier namelijk minder relevant is.

2.3.5 3D-modellen

Tot nu toe is er uitgegaan van 2D-beelden genomen door camera's. Dit is voldoende om de zwerfvuilfracties te tellen. In het verloop van de haalbaarheidsstudie werd door de OVAM een extra vraag gesteld, nl. "Is het mogelijk een goede inschatting te kunnen maken over het volume en bijgevolg het gewicht van een gevonden zwerfvuilobject?". Om dit mogelijk te maken zijn 3D-beelden vereist.

Ordina heeft reeds ervaring met het bepalen van volumes van gedetecteerde objecten. Een accurate bepaling van het volume van een object is echter zeer complex en vergt bijgevolg veel tijd. Zo duurde het bijvoorbeeld drie maanden om het volume van één object correct te kunnen inschatten.

³¹ <https://github.com/experiencor/kangaroo>

³² <https://github.com/AlexeyAB/darknet>

Een eenvoudigere benadering dan het gebruik van een 3D-model is om de totale hoeveelheid van een zwerfvuilfractie (output van het 2D AI model) te vermenigvuldigen met het gemiddelde standaardgewicht en standaardvolume van die bepaalde fractie. Samen met de OVAM is beslist om het volume en gewicht via deze eenvoudigere benaderingen in te schatten. Een systeem die gebruik maakt van 3D-beelden zou accurater zijn, maar beduidend meer kosten (in termen van ontwikkelingskosten en rekenkracht) die niet opwegen tegen de extra baten die het zou opleveren.

Een andere complexiteit binnen deze studie die opgelost zou kunnen worden met 3D-modellen is de reconstructie van gebroken voorwerpen. Gebroken zwerfvuilobjecten, denk bv. aan een plasticen lepel, verliezen hun vorm en structuur waardoor de kans groot is dat een 2D AI model het zwerfvuilobject niet zou detecteren of hetzelfde object meerdere malen zou detecteren. 3D-modellen zouden aan de hand van 3D-beelden de originele zwerfvuilobjecten kunnen reconstrueren alvorens door het 2D AI model te sturen.

Ook hier zal een kosten/baten analyse licht doen schijnen op de vraag of het economisch verantwoord is om dit te bouwen. Online zijn geen publieke modellen te vinden die dit probleem aanpakken. Bijgevolg moet het systeem zelf uitgevonden worden van a tot z. Dit doet vermoeden dat de ontwikkeling van een dergelijk systeem een hoge kost zal hebben. Ook valt te betwijfelen of dit systeem alle gebroken zwerfvuilobjecten kan reconstrueren. Stukken van een object kunnen namelijk over de gehele straat verspreid liggen en mogelijks niet compleet gecapteerd worden. In samenspraak met de OVAM is beslist de reconstructie van gebroken zwerfvuilobjecten buiten beschouwing te laten in het verdere verloop van de studie en de POC.

2.3.6 Besluit

Het detecteren van zwerfvuil bevindt zich in het applicatie domein **object detectie** (paragraaf 2.3.1.2). Om een beslissing te maken over het type model werden twee criteria opgesteld: het performantiecriterium en het selectiviteitscriterium. Het performantiecriterium selecteerde de best presterende modellen in termen van mAP. Analyse van de modellen gescoord op deze twee benchmarks (COCO-benchmark en de PASCAL VOC-benchmark) maakte duidelijk dat one-shot modellen in realiteit, bij een gelimiteerd aantal klassen, equivalent scoren als double-shot modellen. De inclusie van een one-shot model is daardoor aantrekkelijk gezien zijn significant snellere trainingstijd. De modellen werden gereduceerd tot een YOLO (one-shot), Mask R-CNN (double-shot) en Faster R-CNN (double-shot) (paragraaf 2.3.3).

In paragraaf 2.3.3.1, werd het selectiviteitscriterium toegepast door te onderzoeken of er modellen publiek zijn gemaakt die al kennis hebben over zwerfvuil. Online bestaan er geen one-shot zwerfvuildetectie modellen, waardoor dit criterium enkel kan toegepast worden om de keuze te maken tussen de twee overgebleven double-shot modellen. In totaal werden zes voorgaande zwerfvuildetectie onderzoeken besproken, waarvan twee het model gepubliceerd hadden. Deze twee modellen hanteren een Mask R-CNN architectuur en werden getest op een zelf samengestelde dataset. Het model van Let's do it foundation³³ werd empirisch de beste bevonden.

³³ <https://github.com/letsdoitworld/wade-ai>

Samengevat zal in de POC gebruik gemaakt worden van twee modellen: **het YOLO v5-model (one-shot)** en **het let's do it foundation-model (double-shot)**.

Het trainen van deze modellen introduceert echter een aantal generieke AI-problemen welke belicht werden in Paragraaf 2.3.4. Ten eerste zal voor een goede werking van de object detectie modellen veel gelabelde data nodig zijn, iets dat duur en tijdsintensief is. *Unsupervised learning* lost dit probleem op maar staat in het domein object detectie nog in kinderschoenen. *Overfitting* is een tweede uitdaging dat in elk AI-trainingsproces voorkomt. Er werden meerdere methodes toegelicht om dit te voorkomen. Tot slot zijn er de lange trainingstijden bij het trainen van een object detectie model. Het gebruik van een **GPU** reduceert de trainingstijd aanzienlijk ten opzichte van een CPU, waarbij de **Cloud** extra flexibiliteit biedt op vlak van opschaling.

De getrainde AI modellen zijn tweedimensionaal waardoor twee toepassingen (volume bepaling en gebroken zwerfvuilobject herkenning) technisch onhaalbaar zijn tenzij er specifieke 3D-modellen worden ontwikkelt. De kosten van dergelijke ontwikkelingen zijn groter dan de baten die het oplevert (paragraaf 2.3.5).

Bovenstaand onderzoek in acht nemend, stelt ons in staat één van de basis onderzoeksvragen te beantwoorden: **“Is het met de huidige technologie mogelijk om zwerfvuilitems te classificeren?”**. Deze vraag wordt beantwoord binnen de context van AI en houdt geen rekening met bijkomende operationele problemen zoals “wat met obstakels die het zicht op zwerfvuilobjecten blokkeren?”. Met andere woorden kan een object detectie model een zwerfvuilobject detecteren en classificeren op een afbeelding?

Mits voldoende trainingsmateriaal en op basis van de resultaten van reeds bestaande object detectie modellen, is Ordina ervan overtuigd dat dit mogelijk is. Paragraaf 2.3.3.1 bevestigde dat het ook mogelijk moet zijn voor het detecteren van zwerfvuil. Door deze voorgetrainde modellen verder te trainen op een grote hoeveelheid, kwalitatief trainingsmateriaal, kan theoretisch gezien een goede accuraatheid bekomen worden. De hoeveelheid trainingsdata die nodig is, hangt af van de beoogde accuraatheid door de OVAM. Hoe hoger de eisen van de OVAM, temeer trainingsdata nodig is. In de POC zal dit onderzocht worden.

Belangrijk te melden is dat de performantie van automatische zwerfvuildetectie af zal hangen van het te detecteren object. Het is lastiger, en bijgevolg tijdrovend, om een werkend detectie model te verkrijgen voor fracties waarvan objecten verschillen in vorm, grootte en kleur dan voor objecten die “gestandaardiseerd” zijn, vb. blikjes versus sigarettendoosjes. Ook kleinere zwerfvuilobjecten zullen typisch een uitdaging vormen voor het object detectie model.

2.4 ZWERFVUILTELLING

Enkel het capteren van afbeeldingen en het bouwen van een AI model zal niet volstaan om een zwerfvuiltelling te kunnen uitvoeren. De uitkomst van het AI model zal nog verder verwerkt moeten worden om tot een effectieve zwerfvuiltelling te kunnen komen. De gedetecteerde objecten door het AI model moeten enerzijds gelinkt worden aan een geografische coördinaat, ook wel object lokalisatie genoemd. Anderzijds moeten alle gedetecteerde objecten op een correcte manier kunnen geteld worden, zonder objecten dubbel te tellen.

Object lokalisatie en het probleem van dubbeltellingen moeten niet noodzakelijk los van elkaar gezien worden. Het is immers denkbaar dat het toekennen van nauwkeurige coördinaten aan de gedetecteerde objecten tevens een oplossing kan bieden voor dubbeltelling. Indien een object van dezelfde klasse, of in dit geval van hetzelfde zwerfvuiltype, teruggevonden wordt op eenzelfde geografische locatie is het immers denkbaar dat het hier over hetzelfde object gaat. Vooraleer verder in te gaan op object lokalisatie in paragraaf 2.4.2 wordt eerst het probleem van dubbeltelling in meer detail besproken in paragraaf 2.4.1.

2.4.1 Dubbeltellingen

Het probleem van dubbeltellingen kwam een eerste keer reeds aan bod in paragraaf 2.1.3. De manier van dubbeltelling die daar als probleem naar voor kwam is slechts één van de twee manieren waarop dubbeltelling kan ontstaan. Beide manieren van dubbeltelling worden hieronder toegelicht.

Ten eerste kan dubbeltelling ontstaan als gevolg van een sequentie aan *frames* zoals besproken in paragraaf 2.1.3. AI modellen binnen het applicatiedomein van object detectie hebben immers geen geheugen. Deze modellen behandelen elk frame uit een video volledig alleenstaand en onthouden de output van voorgaande *frames* niet. Indien een object dus op 60 opeenvolgende *frames* te zien is, zal dit object 60 keer geteld worden.

Een tweede vorm van dubbeltelling ontstaat bij meervoudige passage van eenzelfde gebied. Het is immers denkbaar dat meermaals door dezelfde straat wordt gereden. Objecten die dan in beide passages gedetecteerd worden zullen opnieuw geteld worden.

Beide vormen van dubbeltelling kunnen losgekoppeld worden van het AI model. Het object detectie AI model zal enkel in staan voor de detectie van objecten, maar geen rekening houden met of het dit object reeds eerder gezien heeft. Concreet betekent dit dat de nood voor een bijkomend, *Post-processing* algoritme ontstaat. Dit algoritme zal de output van het AI model gebruiken om het probleem van dubbeltelling op te vangen. Dit kan afhankelijk van de oplossing gebeuren met de toevoeging van extra informatie zoals GPS data.

Er kunnen twee mogelijke trajecten beschouwd worden om het probleem dubbeltellingen op te vangen:

- **Object Lokalisatie:** Door de combinatie te maken van de videostream met GPS data, gyroscoop data en fotogrammetrische principes kan in theorie een nauwkeurige geografische locatie worden bepaald voor elk gedetecteerd zwerfvuilobject. Dubbeltelling kan dus vermeden worden door zwerfvuilobjecten met éénzelfde, nauwkeurige geografische locatie als dezelfde objecten te beschouwen. Deze oplossing is volledig afhankelijk van de GPS accuraatheid, welke besproken werd in paragraaf 0. Zowel de theoretische uiteenzetting van de methodologie als de principes van fotogrammetrie kan in meer detail worden teruggevonden in paragraaf 2.4.2.
- **Object tracing:** Het AI model maakt geen gebruik van de informatie die gebonden is aan de sequentie waarin de *frames* aan het model worden aangeboden. Echter, er bestaan algoritmen die deze sequentiële informatie wel gebruiken. Deze informatie wordt gebruikt om waarschijnlijkheden te bepalen dat objecten in opeenvolgende *frames* hetzelfde zijn. Deze methodologie kan dus – met een bepaalde waarschijnlijkheid – dubbeltellingen als gevolg van meerdere opeenvolgende *frames* die hetzelfde object bevatten voorkomen. Er wordt verder ingegaan op object tracing in paragraaf 2.4.3.

Object lokalisatie draagt de voorkeur als oplossingsmethodologie. Deze methode stelt ons immers in staat om niet alleen het probleem van dubbeltelling op te lossen, maar zal elk gedetecteerd object ook voorzien van een geografische coördinaat. Het kennen van de geografische locatie van een object maakt het dan mogelijk om dit object toe te kennen aan een 10 bij 10 meter cel. Paragraaf 0 leert echter dat zelfs mits gebruik van geavanceerde hardware, nog steeds rekening gehouden moet worden met een fout van twee tot vijf meter op de GPS-coördinaten. Dit betekent dat dubbeltelling oplossen via object lokalisatie vereist dat er geen zwerfvuilitems van dezelfde klasse binnen een afstand van minimaal vijf meter van elkaar liggen. Een assumptie die praktisch onhaalbaar is. Bijgevolg kan object lokalisatie het dubbeltelling probleem niet oplossen.

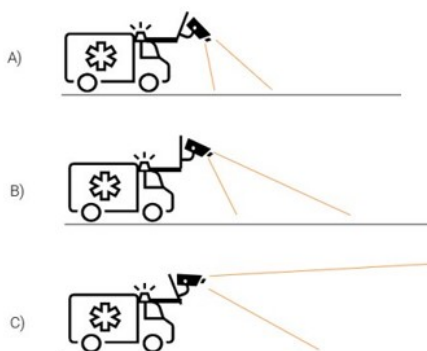
Aangenomen dat omgevingstypes niet elke vijf meter wijzigen is object lokalisatie nog steeds een valide optie om benaderende geografische locaties aan objecten toe te kennen en deze zodoende aan een type-omgeving te koppelen. De methodologie voor object lokalisatie wordt in meer detail besproken in paragraaf 2.4.2.

Het probleem van dubbeltellingen zal desalniettemin moeten opgevangen worden via object tracing. Object tracing heeft echter het nadeel dat het enkel gebruikt kan worden voor dubbeltellingen als gevolg van sequentie van *frames*. Dubbeltellingen die ontstaan door meervoudige passage van hetzelfde gebied zullen daardoor op een andere manier moeten aangepakt worden. De meest voor de hand liggende oplossing hiervoor is het zodanig bepalen van routes zodat meervoudige passages niet voor komen. Object tracing wordt in meer detail besproken in paragraaf 2.4.3.

2.4.2 Object lokalisatie

Omdat meervoudige passages dubbeltellingen zullen veroorzaken, zal gestreefd worden naar het behalen van een zo groot mogelijk bereik met één passage. Het bereik van de camera is afhankelijk van de positie waarop de camera gemonteerd staat op zijn drager. Voor de eenvoud wordt, op basis van paragraaf 2.1.2, verondersteld dat de camera op een wagen gemonteerd wordt. Door de hoek die de camera maakt ten opzichte van de wagen, wordt het bereik groter of kleiner. Dit concept is geïllustreerd in Figuur 25.

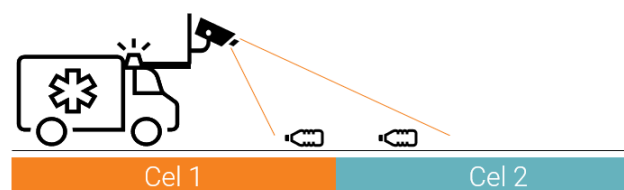
Figuur 25: Camera positie



Een kleiner bereik heeft als voordeel dat gecapteerde beelden over een hogere resolutie zullen beschikken, maar heeft als nadeel dat we minder oppervlakte dekken met een passage. Een groter camera bereik draagt daarom de voorkeur zolang de beeldkwaliteit voldoende hoog blijft om vuil te detecteren.

Een groter bereik brengt een extra probleem met zich mee: De geografische locatie van gedetecteerde objecten wijkt verder af van de geografische positie van de camera. Bovendien kunnen meerdere gedetecteerde objecten in één frame, zich in verschillende cellen van 10 bij 10 meter bevinden, zoals zichtbaar in Figuur 26.

Figuur 26: Eén frame capteert meerdere cellen



Indien men wil vasthouden aan de pre-gedefinieerde cellen is het noodzakelijk om de *frames* geografisch te oriënteren. Praktisch betekent dit dat op basis van de afbeelding, de locatie van het zwerfvuil relatief bepaald moet kunnen worden ten opzichte van de camera positie. De positie van de camera wordt immers verwacht gekend te zijn. De relatieve positie van het zwerfvuil ten opzichte van de camera en de werkelijke coördinaten van de camera stellen de gebruiker in staat om de geografische locatie van het zwerfvuil te bepalen.

Object lokalisatie is daardoor te ontbinden in twee stappen:

- De locatiebepaling van de camera.
- De relatieve locatie van een gedetecteerd object ten opzichte van de camera.

Stap 1: de locatiebepaling van de camera

De locatie bepaling van de camera berust op het gebruik van GPS. Uit paragraaf 2.2.2 blijkt dat deze eerste stap maar met een beperkte nauwkeurigheid kan gerealiseerd worden. De afweging moet dus gemaakt worden of het zinvol is om een relatieve locatie ten opzichte van de camera te bepalen gegeven de onnauwkeurigheid van de locatie van de camera zelf. In de veronderstelling dat toekomstige technologie het mogelijk maakt om een nauwkeurigere locatie bepaling te doen wordt tevens stap twee in meer detail bekeken.

Stap 2: relatieve locaties van gedetecteerde objecten ten opzichte van de camera

Om een relatieve locatie van objecten ten opzichte van een camera te bekomen moet gekeken worden naar de wetenschap van de fotogrammetrie. Er wordt immers gezocht naar een methode om *bounding boxes* op een afbeelding te koppelen aan een relatieve geografische locatie. Daar *bounding boxes* kunnen voorgesteld worden als een middelpunt met een lengte en een breedte, komt dit er praktisch op neer om de relatieve locatie te bepalen van het middelpunt van de *bounding box* ten opzichte van de camera.

Om de relatieve locatie van een object volledig te bepalen ten opzichte van de camera is er nood aan volgende grootheden:

- De afstand van het object tot de camera
- De windrichting waarin de camera gepositioneerd is.

Het bekomen van de eerste grootheid wordt verder besproken in paragraaf 0. Hoe de tweede grootheid bepaald kan worden, wordt besproken in paragraaf 2.4.2.2.

2.4.2.1 **Fotogrammetrie**

Fotogrammetrie is de wetenschap die zich bezighoudt met het omzetten van afbeeldingen tot een 3D beeld. Door het bekomen van een 3D beeld kan telkens de afstand bepaald worden tussen de camera en het object. Het is logisch om het antwoord voor dit probleem in deze wetenschap te zoeken. Binnen fotogrammetrie wordt een onderscheid gemaakt op basis van het aantal afbeeldingen dat nodig is om een locatie inschatting te maken en de manier waarop de foto's gemaakt worden.

De indeling op basis van het aantal noodzakelijke afbeeldingen:

- Mono-fotogrammetrie (één afbeelding nodig)
- Stereo-fotogrammetrie (minimaal twee afbeeldingen nodig)

De indeling op basis waarvandaan de foto's gemaakt zijn:

- Terrestrische fotogrammetrie ("close range" of afbeeldingen gemaakt van op de vaste grond)
- Luchtfotogrammetrie (afbeeldingen gemaakt van uit de lucht)

Daar in paragraaf 2.1.2 beslist werd om te werken met grond-modi kadert dit onderzoek binnen de terrestrische fotogrammetrie. Luchtfotogrammetrie zal niet verder onderzocht worden. Voor bepaling van de relatieve locaties op basis van afbeeldingen blijven dus twee opties voor handen: mono-fotogrammetrie en stereo-fotogrammetrie. Beiden worden besproken in respectievelijk paragraaf 2.4.2.1.1 en 2.4.2.1.2.

2.4.2.1.1 Mono-fotogrammetrie

Mono-fotogrammetrie wil op basis van een enkel beeld een vertaling doen zodanig de verhoudingen van een bepaald vlak gewaarborgd blijft en dus kan opgemeten worden. Dit proces wordt fotorectificatie genoemd. Dit proces wordt vooral gebruikt bij terrestrische fotogrammetrie, waar de POC gebruik van zal maken.

Om aan mono-fotogrammetrie te doen is het noodzakelijk om eerst een geografische kalibratie te bekomen. Deze kalibratie kan gebeuren door verschillende punten in een stilstaand frame te koppelen aan echte geografische locaties. Praktisch moet deze kalibratie gebeuren na montage van de camera. Eenmaal de camera vast gemonteerd is, is immers het beeld bereik van de camera vastgelegd. Vervolgens wordt de camera aangezet en kan een persoon naar verschillende punten op het camera beeld gaan. Op elk van die punten moet deze persoon dan de geografische coördinaten vastleggen, bijvoorbeeld door gebruik van GPS. Dit concept wordt geïllustreerd op Figuur 27. Er van uitgaand dat dit een stilstaand beeld is, kan dus voor elk van de punten de longitude en latitude bekomen worden.

Figuur 27: Mono-fotogrammetrische rectificatie



Vervolgens kunnen de schaalfactoren berekend worden:

Vergelijking 5: Schaalfactoren berekening

$$\begin{aligned} \text{Schaal } (POS1, POS2) &= \frac{\text{Pixel Afstand } (POS1, POS2)}{\text{Fysieke Afstand } (POS1, POS2)} \\ \text{Schaal } (POS3, POS4) &= \frac{\text{Pixel Afstand } (POS3, POS4)}{\text{Fysieke Afstand } (POS3, POS4)} \end{aligned}$$

Deze schaalfactoren worden gebruikt om het perspectief te berekenen:

Vergelijking 6: Perspectief berekening

$$\text{Perspectief} = \frac{\text{Schaal } (POS1, POS2)}{\text{Schaal } (POS3, POS4)}$$

Dit stelt ons in staat om posities op de foto te bepalen. Om een willekeurig punt P op de afbeelding te interpoleren tussen POS1 en POS3 volgens de x-as kan volgende formule gebruikt worden:

Vergelijking 7: Pixel coördinaat berekening

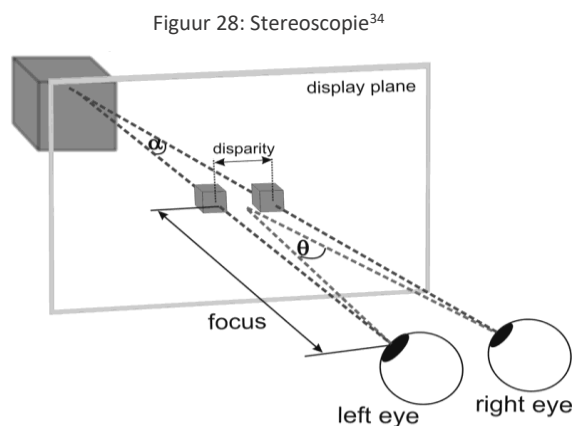
$$P_{x,image} = POS1_{x,image} + \frac{\left[(POS3_{x,image} - POS1_{x,image}) * \left(\frac{P_{x,realty}}{P_{y,realty}} - \frac{POS1_{x,realty}}{POS1_{y,realty}} \right) \right]}{\frac{POS3_{x,realty}}{POS3_{y,realty}} - \frac{POS1_{x,realty}}{POS1_{y,realty}}}$$

Deze formule kan uiteraard aangepast worden om de omgekeerde berekening te maken en de coördinaten van een gevonden object op het beeld om te zetten naar de werkelijke coördinaten.

Het bepalen van het perspectief en de schaal dient eenmaal te gebeuren per opstelling. Dit in de veronderstelling dat de camera positie vast blijft en niet wijzigt, de weg altijd vlak is (dus op viaducten, bruggen kan dit voor problemen zorgen) en er geen zicht belemmerende objecten of andere hoogteverschillen in beeld zijn. Hoogteverschillen zorgen immers voor vervorming van het perspectief. Als gevolg daarvan is deze methode dus beperkt bruikbaar.

2.4.2.1.2 Stereo-fotogrammetrie

Stereo-fotogrammetrie maakt gebruik van minimaal twee afbeeldingen. Die afbeeldingen dienen uit verschillende posities te zijn genomen, maar hetzelfde gebied te bevatten. Deze methodologie is gebaseerd op stereoscopie, wat omschreven kan worden als de manier waarop onze ogen reliëfverschillen waarnemen. Onze twee ogen maken als het ware twee foto's van hetzelfde beeld, elk vanuit hun eigen positie, waardoor een inschatting kan gemaakt worden van de diepte. Dit concept wordt geïllustreerd in Figuur 28.

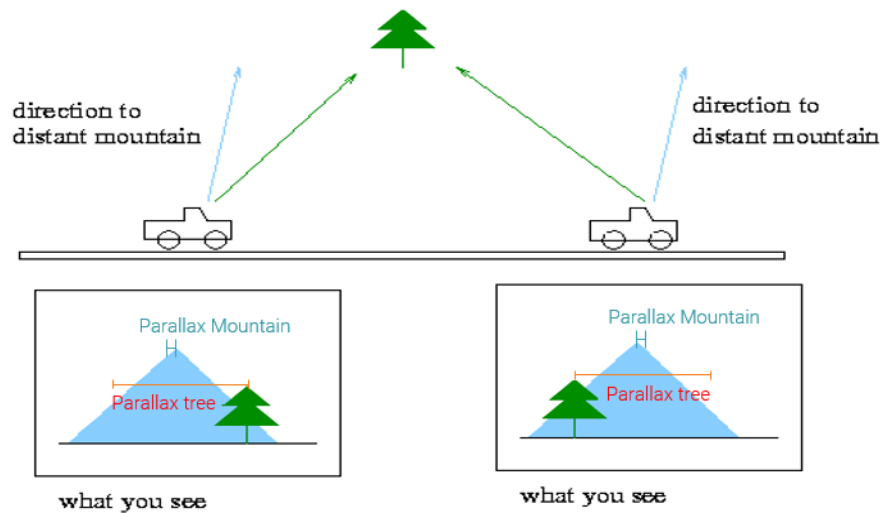


De diepte bepaling die onze ogen maken is op basis van de parallax in combinatie met een gekende baseline. De term "Parallax" wordt hieronder toegelicht. De baseline is in dit geval niets anders dan de afstand tussen onze ogen.

Het begrip parallax kan best omschreven worden als "de schijnbare verandering in positie van een object door de beweging van de waarnemer". Dit begrip kan het best worden uitgelegd aan de hand van een eenvoudig voorbeeld: Bij het rijden door een landschap wordt op een bepaald tijdstip door een waarnemer een foto genomen van het landschap. Dit landschap bevat een boom dichtbij en een bergtop veraf. De waarnemer beweegt zich enkele meters verder en neemt opnieuw een foto van hetzelfde landschap. De opstelling samen met de genomen afbeeldingen wordt voorgesteld in Figuur 29. Daar de boom veel dichterbij de waarnemer was in vergelijking met de bergtop, zal de boom een veel grotere verplaatsing op de afbeelding ondergaan hebben dan de bergtop. Die verschuiving wordt de parallax genoemd en is veel kleiner voor objecten die verder van de waarnemer staan, dan objecten die dichterbij de waarnemer staan.

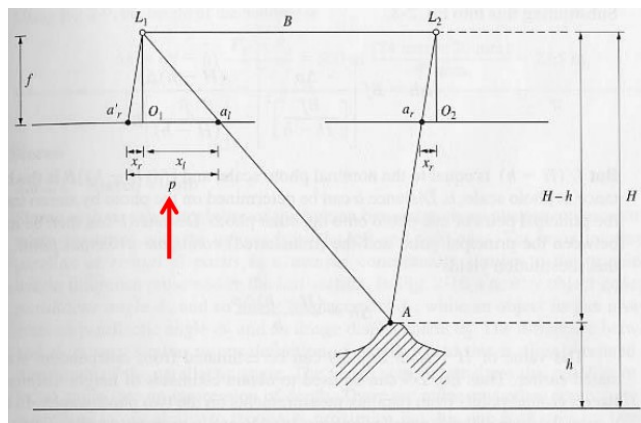
³⁴ https://www.researchgate.net/figure/Concept-of-stereoscopic-viewing-By-fusing-left-and-right-images-the-brain-perceives_fig7_257922664

Figuur 29: Parallax voorbeeld ³⁵



Indien men van twee genomen afbeeldingen de exacte camerapositie kent, kan men mathematisch bepalen wat de afstand tot een zeker punt is. Dit principe wordt geïllustreerd in Figuur 30.

Figuur 30: Parallax mathematisch ³⁶



Op

Figuur 30 stelt B de afstand voor tussen twee lenzen L_1 en L_2 . Deze lenzen kunnen in realiteit ogen zijn, maar ook camera's. De afstand tussen beide lenzen wordt geacht gekend te zijn. f is de brandpuntsafstand van de lens. In cameraterminologie wordt deze ook wel de cameraconstante of focusafstand genoemd. Ook deze parameter is

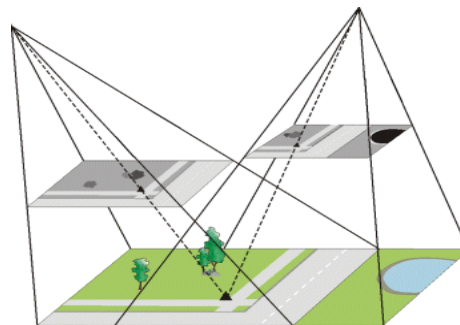
³⁵ <https://www.eg.bucknell.edu/physics/astronomy/astr101/specials/parallax.html>

³⁶ <https://seos-project.eu/3d-models/3d-models-c02-p03-s01.nl.html>

gekend. De twee bekomen afbeeldingen bevatten beiden een projectie van punt A. De projecties van punt A op de afbeelding worden respectievelijk a_l (A op afbeelding links) en a_r (A op afbeelding rechts) genoemd. Het punt a_r van Lens 2 kan geplaatst worden op de afbeelding genomen door L_1 . Dit punt noemen we dan a'_r . De afstand tussen a_l en a'_r is de Parallax P. De parallax is vooral interessant indien men hoogteverschillen wil detecteren tussen twee punten op de afbeelding. Dit is voor deze case minder interessant en wordt daarom niet verder uitgewerkt.

Wat echter wel interessant is aan deze uitwerking, is de mogelijkheid om op basis van de rechten L_1 , a_l en L_2 , a_r , de locatie te bepalen van het punt A ten opzichte van beide lenzen. Dit is immers exact wat we nodig hebben: een afstandsbepaling relatief ten opzichte van de camera. Het zoeken van het snijpunt tussen deze rechten is wat men noemt “voorwaartse insnijding”. Het principe wordt visueel voorgesteld in Figuur 31.

Figuur 31: Voorwaartse insnijding ³⁷



Uit prospectie rond fotogrammetrie blijkt dat er hardware op de markt bestaat met beschikbare “Software Development Kits” (SDK’s) die toelaten om deze principes te combineren met object detectie modellen. Naar deze hardware wordt verwezen als stereocamera’s of depth camera’s. Ze bevatten twee lenzen waardoor ze in staat zijn om een drie dimensionale puntenwolk te creëren. Deze puntenwolk bevat de coördinaten van elk punt in het beeld van de camera in een assenstelsel waarvan de oorsprong samenvalt met de camera lens. Deze puntenwolk maakt het mogelijk om voor elk punt in het beeld van de camera een inschatting te maken van de afstand van dat punt tot de camera alsook de hoek die het object maakt met de kijkrichting van de camera. Kortom, op basis van deze puntenwolk kan voor elke pixel een relatieve positie ten opzichte van de camera berekend worden. Twee vaak gebruikte stereocamera’s zijn de ZED en Intel RealSense.

Het is duidelijk dat zelf een dergelijk algoritme bouwen complex is. Indien een dergelijk algoritme ontwikkeld moet worden op eigen hardware moet rekening gehouden worden met een aantal praktische problemen:

- De noodzaak aan twee afbeeldingen met voldoende overlap vanuit twee verschillende posities.
- De ruimtelijke positie en oriëntatie van de camera(s) op moment dat beide afbeeldingen genomen werden.
- Bepaling van de projecties op twee afbeeldingen van hetzelfde punt.

³⁷ <https://seos-project.eu/3d-models/3d-models-c02-p04.nl.html>

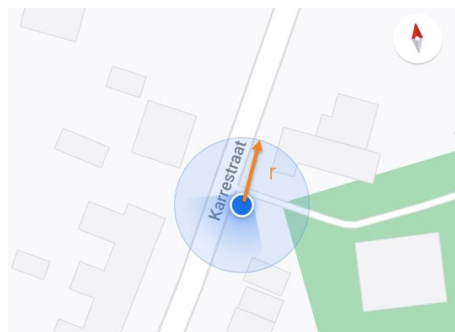
- Geen voorwaartse insnijding mogelijk. Kruisende rechten in plaats van snijdende rechten.

Voor de volledigheid worden deze praktische problemen verder uitgewerkt in bijlage, respectievelijke paragrafen 7.2.1.1, 7.2.1.2 en 7.2.1.3. Na het lezen van de bijlagen zal duidelijk zijn waarom het veel efficiënter is om gebruik te maken van de voorhanden zijnde oplossingen.

2.4.2.2 De Richting van de camera en IMU

Op basis van de principes uit de stereo-fotogrammetrie kan de relatieve locatie van een gedetecteerd object door het AI model ten opzichte van de camera bepaald worden zoals besproken in paragraaf 2.4.2.1.2. Deze relatieve coördinaten kunnen echter niet zomaar opgeteld worden bij de positie van de camera. Deze wordt immers bepaald door de coördinaten bekomen van een GPS ontvanger. De coördinaten stellen een puntlocatie voor, maar geven geen informatie over de oriëntatie van de camera. Daarom is enkel geweten dat het zwerfvuilobject zich ergens op de cirkel rond de camera bevindt met straal (r). Deze straal kan berekend worden uit de puntenwolk gevormd door stereo-fotogrammetrie. Dit probleem wordt geïllustreerd op Figuur 32.

Figuur 32: Camera oriëntatie probleem



Het is duidelijk dat naast de relatieve positie van een gedetecteerd object ten opzichte van de camera, ook de camera oriëntatie gekend moet zijn. De oriëntatie van het gedetecteerde object kan zo worden samengesteld uit de (absolute) oriëntatie van de camera en de relatieve oriëntatie van het object ten opzichte van de camera.

De **relatieve oriëntatie** van het object ten opzichte van de camera kijkrichting kan, zoals reeds aangegeven in paragraaf 2.4.2.1.2, berekend worden op basis van stereo-fotogrammetrische principes. Deze principes leveren immers een 3D puntenwolk op. Deze puntenwolk geeft voor elke pixel van de afbeelding, relatieve coördinaten ten opzichte van de camera. Uit deze coördinaten kan zowel de relatieve afstand, maar ook de relatieve hoek ten opzichte van de camera berekend worden.

De **absolute oriëntatie** van de camera wordt voorgesteld als een nauwkeurige windrichting. Eigenlijk gaat het hier dus opnieuw om een relatieve oriëntatie ten opzichte van het noorden. De oriëntatie van de camera kan bekomen worden op twee methoden:

- Op basis van GPS sequentie data
- Op basis van sensor data

2.4.2.2.1 Camera oriëntatie op basis van GPS sequentiële data

Op basis van opeenvolgende geografische coördinaten kan een bewegingsvector bepaald worden. Deze vector wordt bekomen door enerzijds te kijken naar een frame en zijn bijhorende GPS-coördinaat en anderzijds naar een meteen volgende frame en zijn bijhorende GPS-coördinaat. Hoe meer *frames*, hoe korter op elkaar de GPS-coördinaten volgen, hoe nauwkeuriger deze verplaatsingsvector zal zijn.

Deze methode heeft als voordeel dat er geen nood is aan extra hardware. Het nadeel van deze methode is ongetwijfeld de nauwkeurigheid. Uit paragraaf 0 werd immers aangetoond dat GPS-toestellen in het beste geval nog steeds een onnauwkeurigheid van twee meter hebben. De oriëntatie bepalen op basis van sequentiële GPS-coördinaten zal dus ook een redelijk grote onnauwkeurigheid tot gevolg hebben.

Bovendien moet nog opgemerkt worden dat deze methode enkel werkt indien het voertuig zich ook beweegt in de richting van de camera. Als het voertuig zich in de tegenovergestelde richting zou verplaatsen, zou het lijken alsof de camera 180° gedraaid is. In conclusie, is het duidelijk dat deze oplossing niet ideaal is.

2.4.2.2.2 Camera oriëntatie op basis van sensor data

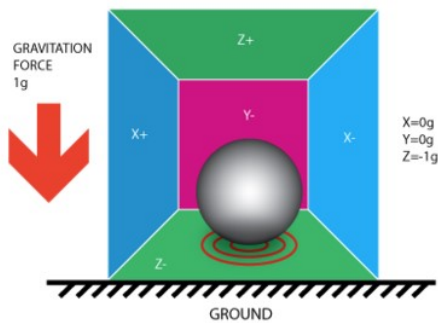
Bij deze tweede methode kijkt men naar sensor data. Drie typen sensoren laten toe de oriëntatie te bepalen:

- acceleratiemeter
- gyroscoop
- magnetometer

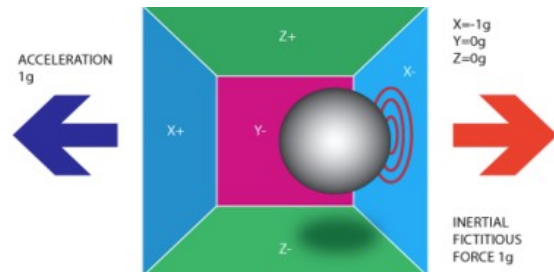
In sommige gevallen wordt ook gebruikt gemaakt van een barometer. Een barometer heeft in dit geval voornamelijk als doel de hoogte waarop het toestel zich bevindt te bepalen. Hoe hoger men zich bevindt, hoe lager de luchtdruk zal zijn. In paragraaf 2.1.2 werd gekozen om te werken met grond-modi. Daarnaast is het Vlaamse landschap relatief vlak, waardoor een barometer niet meer interessant is voor het onderzoek en dus ook niet verder besproken zal worden.

De **acceleratiemeter** kan beschouwd worden als een bal die zich in een kubus bevindt. In rust zal de bal niet bewegen en druk uitoefenen op het ondervlak van de kubus. Deze druk komt overeen met de zwaartekracht of 1g. Een grafische weergave kan worden teruggevonden in Figuur 33.

Figuur 33: Acceleratiemeter in stilstand³⁸



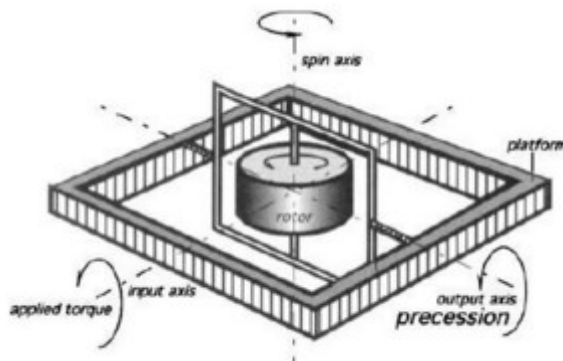
Figuur 34: Acceleratiemeter in beweging³⁸



Indien een versnelling in een bepaalde richting gebeurt, zal deze bal een druk uitoefenen in de tegenovergestelde richting (wetten van Newton). Een object in rust wil immers in rust blijven. Een voorbeeld hiervan is afgebeeld op Figuur 34. De druk op de wand wordt gemeten waardoor men een beeld krijgt van de versnelling in een bepaalde richting.

De tweede sensor is een **gyroscop**. Dit is een sensor die hoekveranderingen en dus rotaties rond drie assen opmeet, zoals getoond in Figuur 35. De gyroscop meet dus geen absolute waarde, maar veranderingen ten opzichte van een eerdere positie. Theoretisch zal men een beginpositie vastleggen en de rotaties opmeten ten opzichte van die startposities. De praktijk wijst echter uit dat de hoekmetingen van een gyroscop zeer nauwkeurig zijn indien het tijdsinterval waarover ze beschouwd worden klein is. Indien het tijdsinterval te groot wordt, treedt drift op. Drift is het fenomeen waarbij de “opgetelde” of eigenlijk geïntegreerde hoekrotaties over de tijd heen niet meer kloppen met de werkelijke rotaties. De gyroscop is daarom niet geschikt om hoekmetingen te doen over een langere tijdsperiode.

Figuur 35: Gyroscop³⁹



³⁸ <https://www.programmersought.com/article/49873613504/>

³⁹ <https://sensorwiki.org/sensors/gyroscope>

Om een richting te kennen over een langer tijdsinterval wordt de derde sensor, de **magnetometer**, geïntroduceerd. De magnetometer maakt gebruik van het magnetisch veld van de aarde om het noorden te bepalen. De magnetometer stelt ons in staat om elke positie te iken aan het werkelijke noorden. Hierdoor kunnen de gyroscooprotaties telkens bekeken worden ten opzichte van een vorige geijkte positie. Als gevolg hiervan blijven de tijdsintervallen klein en de accuraatheid groot. Drift wordt er op deze manier uitgefilterd.

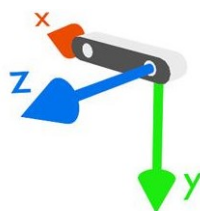
Door deze drie sensoren (acceleratiemeter, gyroscoop en magnetometer) te combineren kan de oriëntatie van een object accuraat bepaald worden ten opzichte van een vorige positie. Indien manueel samengesteld brengt dit typisch bijkomende uitdagingen zoals omzetting van de coördinatensystemen, het afstemmen van de sensor klokken, etc. Het is daarom aangewezen om de markt te verkennen en op zoek te gaan naar bestaande integratie oplossingen. Naar de integratie van deze sensoren wordt in de literatuur verwezen als "*Inertial Measurement Unit*" (IMU).

IMU's zijn de hardwarecomponenten die ervoor zorgen dat een object zijn ruimtelijke positie kent ten opzichte van de aarde. Ze worden onder andere ook gebruikt in vliegtuigen en drones. Daarnaast worden ze vandaag de dag toegepast om GPS data aan te vullen. Op plaatsen waar de GPS geen bereik heeft, zoals in tunnels, kan een IMU tijdelijk berekenen waar het object zich moet bevinden uitgaande van de laatst gekend GPS locatie.

De GoPro beschikt over een gyroscoop en acceleratiemeter, maar beschikt niet over een magnetometer. Ook de Intel RealSense bevat geen interne IMU. ZED beschikt daarentegen wel over een IMU. Intel biedt dan weer wel de mogelijkheid om de stereocamera te combineren met een tracking camera zoals de T265⁴⁰. Daar de ZED beide systemen combineert in één toestel lijkt het eenvoudiger om verder te werken met de ZED. In kader van het onderzoek en de POC werd besloten een ZED aan te kopen.

De oriëntatie van de camera moet gebeuren ten opzichte van een vast referentiepunt. IMU's hebben deze mogelijkheid, gebaseerd op de magnetometer. De toegevoegde waarde van de andere twee sensoren in de IMU zijn voornamelijk voor relatieve positiebepaling ten opzichte van een vorige positie. Als gevolg daarvan is het voor de POC vooral interessant om de magnetometer in meer detail te bestuderen teneinde de camera oriëntatie te bepalen. Om het concept te illustreren wordt ervan uitgegaan dat de camera kijkrichting parallel loopt met de aardoppervlakte. Praktisch nemen we dus aan dat de aardoppervlakte evenwijdig loopt met het xz vlak, als het lokaal assensysteem van de camera is gedefinieerd zoals in Figuur 36.

Figuur 36: Lokaal referentiesysteem van camera⁴¹



⁴⁰ <https://www.intelrealsense.com/how-to-getting-imu-data-from-d435i-and-t265/>

⁴¹ <https://www.stereolabs.com/docs/positional-tracking/coordinate-frames/>

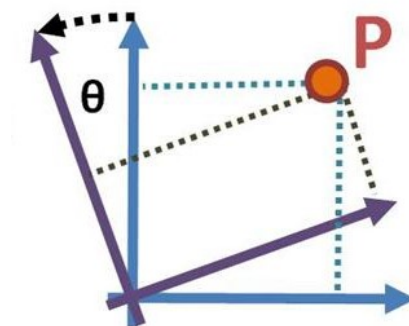
Deze vereenvoudiging zorgt ervoor dat we enkel de rotatiehoek moeten bepalen tussen de z-as en het noorden. Hierdoor wordt een drie dimensionaal probleem herleid tot een twee dimensionaal probleem. Dit probleem kan op analoge wijze drie dimensionaal worden opgelost zodoende de camera een bepaalde hoek te geven, gericht naar de grond. Dit is wenselijk in een eventuele uitrol.

Het vereenvoudigd probleem wordt schematisch weergegeven op Figuur 37. De gezochte hoek wordt op deze afbeelding voorgesteld als de hoek θ .

Figuur 37: Rotatiehoek voor oriëntatie camera



Figuur 38: Rotatie van een assenstelsel⁴²



Om de uiteindelijke geografische coördinaat van het gedetecteerd object te bepalen wordt dus eerst de relatieve positie van het gedetecteerde object ten opzichte van de camera bepaald. Vervolgens worden de bekomen coördinaten uit de puntenwolk herrekend naar het assenstelsel waarbij de Z richting uit Figuur 36 samenvalt met het Noorden. Deze herberekening kan wiskundig eenvoudig gebeuren aan de hand van rotatie formules tussen twee assenstelsels, zoals voorgesteld op Figuur 38.

2.4.2.3 Bepalen van de geografische locatie van het object

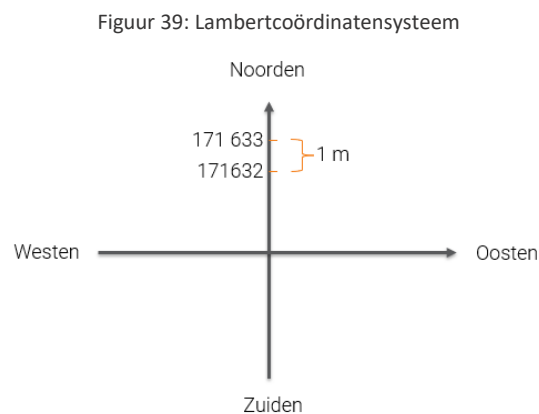
Op basis van voorgaande paragrafen 2.4.2.1 en 2.4.2.2 kan nu de werkelijke locatie van een object berekend worden. GPS coördinaten worden meestal uitgedrukt in het geodetisch systeem WGS 84. Geodetische systemen zijn systemen die gebruik maken van lengte- en breedtegraad. Deze systemen zijn eigenlijk gebouwd voor het beschrijven van een bol en geven dus bolcoördinaten weer. Voor de eenvoud worden deze bolcoördinaten vaak geprojecteerd op een vlakke kaart gebruik makend van ellipsoïden. Afhankelijk van de gebruikte ellipsoïde wordt een andere projectie en dus andere X,Y coördinaten bekomen. Zolang de ellipsoïde gekend is, is het steeds mogelijk om coördinaten te transformeren van een geodetisch systeem naar een vlak coördinaten systeem in beide richtingen.

In België wordt meestal gebruik gemaakt van de Lambert projectie en het Lambertcoördinatensysteem. Dit systeem wordt tevens gehanteerd door het Nationaal Geografisch Instituut (NGI). Een belangrijk voordeel van dit systeem is de afstelling van de coördinaten op de meter schaal. Indien de Y-coördinaat met één eenheid

⁴² https://nl.wikipedia.org/wiki/Passieve_en_actieve_rotatie

wijzig, betekent dit een verplaatsing van 1m. Bovendien loopt de Y-as van dit systeem van Zuid naar Noord en de X-as van West naar Oost. Dit concept wordt geïllustreerd in Figuur 39.

Dit betekent dat we de gekende GPS coördinaat eenvoudig kunnen omvormen tot een X, Y coördinaat in het lambertcoördinatensysteem. Daar de systeemassen gerefereerd zijn aan de windrichtingen, kan de link gemaakt worden met de oriëntatie van de camera zoals besproken in de coördinaten in het geroteerde assenstelsel uit paragraaf 2.4.2.2.2, waarbij de kijkrichting van de camera samenvalt met het Noorden. Vervolgens kan deze eenvoudig worden opgeteld bij de GPS puntcoördinaten in het lambertcoördinatensysteem.



2.4.2.3.1 Object lokalisatie conclusies

Om gedetecteerde objecten te lokaliseren en te linken aan gekende geografische coördinaten is er nood aan een absolute locatiebepaling van de camera en een relatieve locatiebepaling van objecten ten opzichte van de camera.

De relatieve locatiebepaling van gedetecteerde objecten op de afbeeldingen kan bekomen worden door gebruik te maken van fotogrammetrie. De beste methode binnen fotogrammetrie hiervoor is stereo-fotogrammetrie. Omwille van de complexiteit van stereo-fotogrammetrie is het aangeraden om gebruik te maken van bestaande hardware oplossingen, ook wel stereocamera's genoemd.

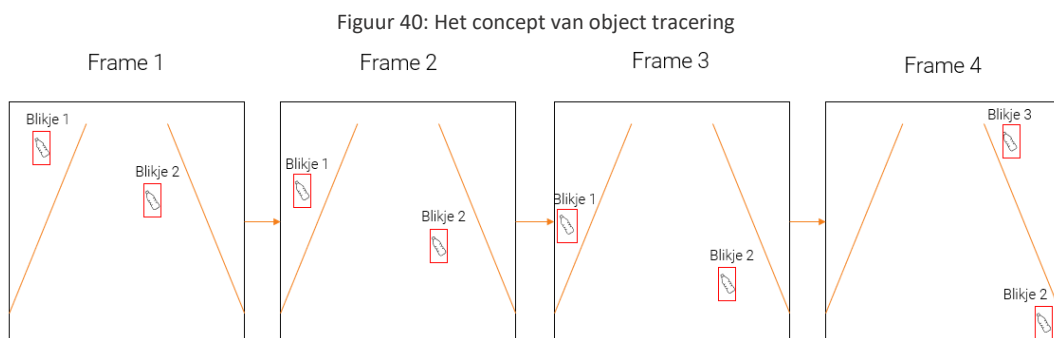
Voor de absolute locatiebepaling van de camera kan gebruik gemaakt worden van een GPS coördinaat in combinatie met IMU gegevens voor camera oriëntatie.

De Go Pro beschikt over een GPS, gyroscop en acceleratiemeter, maar beschikt niet over een magnetometer. De ZED beschikt niet over een GPS, maar wel over een IMU en kent dus op elk moment zijn oriëntatie ten opzichte van de aarde. Omdat de accuraatheid van de klassieke GPS-ontvangers niet voldoende zijn en toch overgestapt zal moeten worden op een meer nauwkeurige ontvanger, werd besloten de ZED-functionaliteiten mee te onderzoeken in de POC.

2.4.3 Object Tracering

Zoals aangegeven in paragraaf 2.4.1, is het probleem van dubbeltelling moeilijk op te lossen door gebruik te maken van object lokalisatie. Voornaamste reden hiervoor is de onnauwkeurigheid van de GPS. Zoals uitvoerig besproken in paragraaf 2.4.1 zijn er twee vormen van dubbeltelling die kunnen ontstaan. Enkel object tracering kan een antwoord bieden op de vorm die tot stand komt door object detectie op sequentiële *frames*.

Object tracering is het omvattende begrip dat algoritmen dekt die proberen om een object doorheen een video te volgen. Het komt erop neer om objecten te volgen over verschillende *frames* van de video heen. Het concept wordt geïllustreerd in Figuur 40. Op een eerste frame worden twee blikjes gedetecteerd die respectievelijk naamgeving 1 en 2 krijgen. Op een volgend frame blijven de twee zelfde blikjes in beeld. Het object tracering algoritme geeft beide blikjes dan ook dezelfde naamgeving. Hetzelfde gebeurt nogmaals in frame drie. Op frame vier is blikje 1 niet meer te detecteren, maar is er wel een nieuw blikje in beeld gekomen. Dit nieuwe blikje krijgt uiteraard een nieuwe naam en wordt blikje 3.



Dit concept kan op verschillende manieren geïmplementeerd worden. Mogelijke algoritmen zijn de volgende:

- CSRT
- KCF
- MOSSE
- SimpleSort
- DeepSort
- ...

Afhankelijk van de toepassing kan een bepaald algoritme beter presteren dan andere. Het aanbod algoritmen is te groot om allemaal in detail te bespreken. We beperken ons tot de meeste beloftevolle, doch eenvoudige algoritmen. Het gekozen algoritme moet voldoen aan een aantal functionele vereisten zoals meerdere objecten tegelijk kunnen traceren rekening houdend met de gedetecteerde objecten door het AI model. Op basis van deze criteria wordt gekozen om het SimpleSort algoritme in meer detail te bekijken. Het SimpleSort algoritme is eenvoudig te implementeren en zal een beeld geven over de mogelijkheden die dergelijke algoritmen bieden.

Het **SimpleSort algoritme** start gebruikmakend van de *bounding boxes* die het AI model als output geeft op het eerste frame. Voor elk van deze *bounding boxes* wordt een *tracerder* aangemaakt om de status van dit object

te volgen. Een traceerder kan beschouwd worden als een lijst van de geschiedenis van een bepaald object. Bij initialisatie zal de traceerder dus slechts één waarde bevatten: de initiële *bounding box*.

De traceerder zal op basis van de initiële *bounding box* een voorspelling maken over waar verwacht wordt de *bounding box* terug te vinden op een volgende frame. In deze situatie, waar de traceerder de *bounding box* slechts één keer gezien heeft, zal de voorspelling identiek zijn aan de locatie van de eerste *bounding box*.

De *bounding boxes* van het AI model van een tweede frame worden vervolgens aan de traceerder aangeboden. Voor elk van de *bounding boxes* van het tweede frame bepaald de traceerder vervolgens of deze *bounding box* al dan niet hetzelfde object bevat als het initiële object van die traceerder. Deze bepaling gebeurt op basis van de IoU-metrick, zoals besproken in paragraaf 2.3.2. De waarde van IoU, waarvoor bepaald wordt dat het om hetzelfde object gaat, is een parameter van het algoritme die aangepast kan worden.

Sommige traceerders zullen het object nu kennen op twee posities, een *bounding box* uit het eerste, originele frame en een *bounding box* uit het tweede frame. Op basis van deze twee posities zal opnieuw een voorspelling worden gedaan waar de *bounding box* zich zal bevinden op een derde frame. Om deze voorspelling te maken wordt gebruik gemaakt van een *Kalman-filter*. Dit kan het best omschreven worden als een rekenmethode waarmee storingen in reeksen meet- of andere gegevens gereduceerd worden. Kalman-filters worden veelvuldig gebruikt voor het bekomen van consistente tijdsreeksen. Zo worden ze bijvoorbeeld gebruikt in navigatiesystemen om uit de vele metingen die allemaal een zekere afwijking hebben, de meest waarschijnlijke positie of koers te bepalen. Voor meer details aangaande Kalman-filters verwijzen we graag naar de literatuur⁴³.

Bounding boxes die door het AI model worden aangeleverd op frame 3 worden vervolgens vergeleken met de voorspellingen van de Kalman-filter. Deze vergelijking berust opnieuw op de IoU-waarde tussen beide *bounding boxes*, nl. de verkregen *bounding box* van frame 3 en de voorspelde *bounding box* door de Kalman-filter. Het is duidelijk dat de Kalman-filter beter zal werken naarmate de beweging voorspelbaar is en een object meer gedetecteerd is.

Om onterechte detectie van objecten te vermijden, kan men instellen hoe vaak een traceerder een goede match moet vinden alvorens een unieke identiteit aan het object toe te kennen. Deze parameter kan tevens meegegeven worden op traceer niveau.

Omwille van computergeheugen moet tenslotte bepaald worden wanneer een traceerder mag stopgezet worden. Het is immers onmogelijk om elke geïnitieerde traceerder eeuwig te laten bestaan. Dit zou immers resulteren in een continu stijgende vraag naar geheugen van de computer. Om deze reden heeft het algoritme een parameter die aangeeft na hoeveel *frames* waarop geen match is gevonden met de traceerder, de traceerder wordt stopgezet.

⁴³ <https://www.intechopen.com/books/introduction-and-implementations-of-the-kalman-filter/introduction-to-kalman-filter-and-its-applications>

Het is duidelijk dat het SimpleSort algoritme enkel gebruik maakt van de sequentiële informatie van *frames* om een object te proberen volgen. Abrupte bewegingen zoals het plots stoppen van een wagen of het maken van een scherpe bocht kunnen een grote impact hebben op de accuraatheid van een dergelijk algoritme.

Het is opportuun en aangewezen om de voorspellingen van de nieuwe locatie van de *bounding boxes* zo accuraat mogelijk te maken. Dit kan gebeuren door de Kalman-filter voorspellingen te verrijken met informatie van de IMU-sensoren. Uit paragraaf 2.4.2.2 weten we immers dat gyroscoop en acceleratiemeters vooral goed presteren als ze worden bekeken op korte tijdsintervallen. In dit onderzoek is dit het geval, daar er louter gekeken wordt naar opeenvolgende *frames*. Omwille van de complexiteit en conceptuele duidelijkheid, zal dit in de POC niet uitgewerkt worden.

Tot slot raadt Ordina aan meer geavanceerde, robuuste algoritmen te gebruiken in een mogelijke uitrol. Ordina heeft reeds ervaring met het DeepSort algoritme. Er werd beslist dit algoritme niet te implementeren gezien de bouw zes tot acht weken vergt.

2.4.4 Besluit

Na object detectie van het AI model moeten nog twee problemen opgelost worden in wat we noemen zwerfvuiltelling. Enerzijds moeten gedetecteerde objecten gelinkt worden aan geografische coördinaat. Anderzijds moet het probleem van dubbeltellingen opgelost worden.

Object lokalisatie op basis van **stereo-fotogrammetrie** kan in combinatie met een **magnetometer** een antwoord bieden op het eerste probleem. Echter omwille van de beperkte GPS accuraatheid zal voor het probleem van dubbeltelling een andere aanpak moeten voorzien worden.

Dubbeltelling als gevolg van meervoudige passage kan vermeden worden door route optimalisatie, al dan niet manueel. Dubbeltelling als gevolg van sequentiële *frames* met zelfde objecten kan opgelost worden door gebruik te maken van object tracing algoritmen zoals SimpleSort. Om een hogere accuraatheid te bekomen van object tracing kunnen deze algoritmen verder ondersteund worden door IMU-data.

2.5 INFRASTRUCTUUR

Om een oplossing uit te bouwen, is er nood aan een IT-infrastructuur. In dit onderdeel wordt nagegaan of de OVAM zijn eigen bestaande infrastructuur kan gebruiken, die *on-premise* is, of naar de Cloud moet gaan voor het AI-systeem. Er wordt een vergelijking gemaakt tussen een *on-premise* en Cloud-strategie, waarna de verschillende Cloud opties verder worden toegelicht. Na elk onderdeel wordt ter conclusie een voorstel gegeven, afgestemd op de noden voor dit project.

2.5.1 Cloud vs. Non-Cloud aanpak

Er wordt een onderscheid gemaakt tussen een non-Cloud en Cloud benadering. In dit deel worden de voor- en nadelen van beide opties besproken.

2.5.1.1 Non-Cloudoplossing

Wanneer er geopteerd wordt voor een non-Cloudoplossing zal er gebruik gemaakt worden van de fysieke infrastructuur van de OVAM zelf, wat ook wel *on-premise* genoemd wordt. Dit brengt enkele voordelen, maar voornamelijk grote nadelen met zich mee.

Het voordeel van een non-Cloud aanpak is dat de OVAM fysieke controle over de server blijft behouden en zo ook de kritische en gevoelige data in-house blijft, aangezien de infrastructuur zelf aangekocht en beheerd zal worden. Tenslotte heeft *on-premise* ten opzichte van het Cloud type *SaaS* (term wordt toegelicht in paragraaf 2.5.2.3) grotere flexibiliteit aangezien *SaaS* een kant-en-klare softwareoplossing is waarbij er geen aanpasbaarheid mogelijk is en bij *on-premise* de infrastructuur zelf kan worden opgesteld. Tegenover een *IaaS*- of *PaaS*-oplossing (termen wordt respectievelijk toegelicht in de paragrafen 2.5.2.1 en 2.5.2.2) heeft een *on-premise* oplossing minder flexibiliteit. Bij deze Cloud types is het makkelijker om hardware snel aan te passen naar wens.

Het grootste nadeel van een non-Cloudoplossing is dat dit minder toekomstgericht is. Hier wordt verder op ingegaan bij onderdeel 2.5.1.2, waarin beschreven wordt waarom de Cloud dit wel is. Bovendien zijn bij een *on-premise* oplossing kapitaaluitgaven vereist. Dit houdt in dat men op voorhand moet investeren in hardware, infrastructuur, fysieke ruimtes, licenties en onderhoud, rekening houdend met de toekomstige noden en wensen. Dit wil zeggen dat de aangekochte infrastructuur op dit moment misschien niet nodig is, maar er toch op voorhand moet in geïnvesteerd worden. Bovendien is een goede planning vereist voor een eventuele opschaling.

Verder is voor installaties en onderhoud ook de nodige knowhow vereist. Wel moet vermeld worden dat er ook de nodige kennis en ervaring zal moeten opgebouwd worden om de Cloud services te kiezen en te gebruiken. Een bijkomend nadeel bij *on-premise* is dat de OVAM vatbaarder zal zijn voor dataverlies bij rampscenario's. De data staat namelijk op slechts één plek opgeslagen. Daarnaast zijn er geen uptime garanties en zal de service-level agreement (SLA) zelf te onderhouden zijn. Een voorbeeld: In de SLA's met Microsoft worden de toezeggingen van Microsoft voor uptime en connectiviteit beschreven. Zo garanderen zij bijvoorbeeld dat de Data Lake Store (opslagmogelijkheid) ten minste 99,9% van de tijd beschikbaar is. Wanneer de dienstniveaus, zoals beschreven in de SLA, niet bereikt worden dan zou de OVAM wellicht korting kunnen krijgen op een deel van de maandelijkse dienstkosten. Wanneer de OVAM voor *on-premise* kiest, zal de OVAM zelf verantwoordelijk zijn voor dit soort scenario's.

In Tabel 7 wordt er een overzicht gegeven van de voor-en nadelen van een non-Cloud of *on-premise* oplossing.

Tabel 7: Voor- en nadelen van een non-Cloud of *on-premise* oplossing

Non-Cloud / on-prem oplossing	
Voordelen	Nadelen
Directe toegang tot het lokale netwerk	Minder toekomstbestendig
Fysieke controle over de server	Minder flexibiliteit t.o.v. IaaS en PaaS
Kritische en gevoelige data blijft in-house	Vereist voorafgaande investeringen
Grotere flexibiliteit t.o.v. SaaS	Vereist meer knowhow dan bij een Cloudoplossing
	Vatbaarder voor data verlies
	Geen uptime-garanties
	Verantwoordelijk voor hoge beschikbaarheid
	Vereist goede planning bij eventuele opschaling

2.5.1.2 Cloudoplossing

Wanneer er gekozen wordt voor een Cloudoplossing, wordt de infrastructuur grotendeels uitbesteed. De term 'Cloud' wordt algemeen gebruikt om datacenters te beschrijven die beschikbaar zijn voor vele gebruikers via het internet. Bij Cloud computing worden op aanvraag hardware en software beschikbaar gesteld via een netwerk. De gebruiker hoeft dus geen eigenaar te zijn van de gebruikte hard- en software en is dus niet meer verantwoordelijk voor onderhoud, management, etc. Dit wordt verder in meer detail besproken in paragraaf 2.5.2. Enkele bekende Cloud providers zijn: Microsoft Azure, Amazon Web Services en Google Cloud.

Eén van de grootste voordelen van een Cloudoplossing is dat deze optie meer toekomstbestendig is in vergelijking met een non-Cloudoplossing. De Cloud biedt niet alleen de mogelijkheid om een op maat gemaakte IT-omgeving te creëren voor de huidige noden, het biedt ook de mogelijkheid om flexibel te schalen naar de noden in de toekomst. Bij een Cloudoplossing is het makkelijk om op of neer te schalen, zonder daar financieel voor afgestraft te worden. De Cloud hanteert immers het principe “you pay as you go”, wat betekent dat betaald wordt naar gebruik, vb. indien extra rekenkracht nodig is voor het trainen van een AI model kan in een cloudoplossing makkelijk worden opgeschaald met een aantal klikken, waar bij *on-premise* oplossingen nog steeds een nieuwe investering vereist is.

Afhankelijk van het niveau van uitbesteding (zie paragraaf 2.5.2) zal de OVAM niet meer verantwoordelijk zijn voor het voorzien en onderhouden van een fysiek datacenter, daarvoor moeten er dus geen extra mensen met de nodige knowhow ingeschakeld worden. De focus op kernactiviteiten kan hierdoor behouden worden.

De Cloud biedt hoge beschikbaarheid, bedrijfscontinuïteit en snel herstel. De data wordt bij opslag altijd drie keer gedupliceerd in een datacenter in een primaire regio. Alle Cloud providers bieden de mogelijkheid om de data op verschillende locaties op te slaan zodat bij een mogelijk rampscenario geen data verloren gaat. Daarnaast evolueert de Cloud continu en integreert het makkelijk met de meest recente technologieën, wat meer mogelijkheden creëert voor de toekomst. Tevens biedt de Cloud onbegrensde mogelijkheden op gebied van samenwerken, zowel intern als met partners. Tot slot is de Cloud overal toegankelijk mits de beschikbaarheid van een internetconnectie, alsook een connectie via VPN Access of ExpressRoute Access. Dit

laatste kan gezien worden als een nadeel. Via Azure VPN Gateway kan de OVAM zijn *on-premise* netwerk verbinden met Azure zonder dat deze via het openbare internet loopt. Dit zorgt voor een grotere betrouwbaarheid, kortere wachttijden en hogere snelheden. Een bijkomstig nadeel van een Cloudoplossing is dat data veiligheid niet meer intern beheerd wordt, maar wordt overgedragen naar de Cloud provider.

Een laatste voordeel is dat de Cloud providers voldoen aan de ISO-normen. ISO-normen zijn standaarden die internationaal overeengekomen zijn door experts (De Internationale Organisatie voor Standaardisatie) die de beste manier beschrijven om bepaalde zaken te doen. Dit kan gaan over het maken van een product, het managen van een proces, het leveren van een dienst, etc. Wanneer bv. Azure voldoet aan de ISO-normen, draagt Azure de belofte met zich mee om aan bepaalde standaarden te voldoen zoals bijvoorbeeld de IT-security standaarden om sensitieve informatie veilig te stellen.

In Tabel 8 wordt een overzicht gegeven van de voor- en nadelen van een Cloudoplossing.

Tabel 8: Voor- en nadelen van een Cloudoplossing

Cloudoplossing	
Voordelen	Nadelen
Toekomstbestendig	Connectie via VPN Access of ExpressRoute Access
Voordelen IaaS, PaaS en SaaS	Internetconnectie nodig
Uptime-garanties	Data veiligheid in handen van de Cloud provider
Pay as you go	
Integreert met meest recente technologieën	
Mogelijkheden tot samenwerken	
ISO-normen	

2.5.1.3 Voorstel Ordina

In paragrafen 2.5.1.1 en 2.5.1.2 werden de voor- en nadelen van zowel een *on-premise* oplossing als een Cloudoplossing opgesomd. In deze analyse werd duidelijk dat voor de doelstellingen van de OVAM, de Cloud een beter alternatief is. Door deze keuze kan de OVAM zich focussen op haar kernactiviteiten.

2.5.2 IaaS, PaaS of SaaS

Nu er gekozen is voor een Cloudoplossing, moet een beslissing gemaakt worden over de mate waarin IT-diensten uitbesteed gaan worden. In Cloud computing wordt een onderscheid gemaakt tussen drie niveaus: IaaS, PaaS, SaaS. Ieder niveau wordt toegelicht samen met zijn voor- en nadelen.

2.5.2.1 IaaS

Bij Infrastructure as a Service (IaaS) wordt het minst uitbesteed. De Cloud provider zorgt enkel voor de infrastructuur en laat de gebruiker de keuzes maken omtrent het aanschaffen, installeren, configureren en beheren van de software die de infrastructuur gebruikt. Het biedt dezelfde mogelijkheden als een traditioneel, *on-premise* datacenter zonder het fysiek te onderhouden of te managen. Bij IaaS wordt gebruik gemaakt van virtuele machines gehost bij de Cloud provider. We lijsten in onderstaande Tabel 9 nogmaals de voordelen en nadelen van een IaaS-oplossing tegenover een non-Cloudoplossing.

Tabel 9: Voor- en nadelen van IaaS t.o.v. een non-Cloudoplossing

IaaS t.o.v. non-Cloud	
Voordelen	Nadelen
Niet meer verantwoordelijk voor voorzien en onderhoud fysieke infrastructuur datacenter	Data veiligheid in handen van de Cloudprovider
Hoge beschikbaarheid, bedrijfscontinuïteit en snel herstel	Connectie met de Cloud via VPN Access of ExpressRoute Access
Schaalbaarheid: de gebruiker kan zeer snel op- en neerschalen naar gelang het gebruik (bv. opslag)	Internetconnectie nodig
Toegankelijkheid	

Een voordeel van IaaS tegenover de andere Cloud computing types is dat de OVAM dan het meeste controle heeft over de infrastructuur, doordat er minder extern wordt uitbesteed. Dit wordt verder uitgelegd in paragrafen 2.5.2.2 en 2.5.2.3. IaaS is dan ook het meest flexibele Cloud computing type omdat de Cloud provider enkel voor de infrastructuur zorgt en de rest openlaat naar de gebruiker. Een nadeel hiervan is dat dit meer verantwoordelijkheid en technische kennis vereist. Alsook is het vaak een duurdere oplossing omdat de gebruiker minder kan genieten van schaalvoordelen van de provider. Tabel 10 lijst de voor- en nadelen tegenover andere Cloud computing types op.

Tabel 10: Voor- en nadelen IaaS t.o.v. andere Cloud computing types

IaaS t.o.v. andere Cloud computing types	
Voordelen	Nadelen
Meeste controle	Vaak duurder
Meeste flexibiliteit	Meer verantwoordelijkheid
	Vereist meer technische kennis

2.5.2.2 PaaS

Bij het tweede type, Platform as a Service (PaaS) worden bestaande Cloud computing diensten gebruikt en op maat aangepast. Het configureren, beheren en monitoren van de onderliggende infrastructuur, software updates, bedrijfssystemen, etc. worden hier opgenomen door de Cloud provider. De gebruiker is zelf enkel verantwoordelijk voor de applicatie die gehost wordt op dit platform.

Omdat PaaS een verdere uitbreiding is van IaaS heeft PaaS, naast de voordelen van IaaS, extra voordelen. Bij PaaS zal de OVAM de kosten en complexiteit vermijden van het kopen en beheren van softwarelicenties. De OVAM zal zich enkel moeten focussen op het ontwikkelen van de applicatie wat gemakkelijk en kostenefficiënt is. Daarnaast is het vaker goedkoper omdat de OVAM meer kan genieten van de schaalvoordelen van de provider. Het grootste nadeel is dat er minder flexibiliteit is door de afhankelijkheid van de provider. Denk maar aan integraties met andere services en infrastructuren.

Tabel 11 somt deze voor- en nadelen op.

Tabel 11: Voor- en nadelen PaaS t.o.v. IaaS

PaaS t.o.v. IaaS	
Voordelen	Nadelen
Vermijdt kosten en complexiteit softwarelicenties	Minder flexibiliteit
Focus op applicatie	
Vaak goedkopere oplossing	

2.5.2.3 SaaS

Het derde type is Software as a Service (SaaS). Dit zijn volledige, kant-en-klare softwaretoepassingen die via het internet geleverd worden. Hierbij wordt alles uitbesteed aan de Cloud provider. De onderliggende infrastructuur, middleware en toepassingssoftware worden door de Cloud provider zelf gehost, beheerd en onderhouden. SaaS is meestal de goedkoopste oplossing omdat de gebruiker kan genieten van de schaalvoordelen van de provider. Een groot nadeel van een SaaS-oplossing is dat er geen flexibiliteit is.

Kant-en-klare SaaS-oplossingen moeten voldoen aan drie eigenschappen om de volledige scope van dit probleem te benaderen:

1. Zwerfvuildetectie/classificering
2. Telling van de fracties
3. Notie hebben van ruimtelijke plaats

SaaS zou de beste keuze voor de OVAM kunnen zijn, maar momenteel bestaan er geen kant-en-klare SaaS-oplossingen op de markt die de wensen van de OVAM volledig dekken. Dit betekent dat er door Ordina een applicatie ontwikkeld zal moeten worden en dat SaaS geen optie is.

2.5.2.4 Conclusie Cloud type

PaaS en IaaS zijn de twee Cloud computing mogelijkheden die beschikbaar zijn voor de OVAM. Aangezien er geen nood is aan veel flexibiliteit en andere voordelen van PaaS wenselijk zijn, is Platform as a Service in dit geval de beste optie voor de OVAM. Indien er tijdens de studie blijkt dat er geen passende PaaS-oplossing bestaat, blijft IaaS een mogelijkheid.

Daarnaast werd op basis van de wensen en toekomstvisie van de OVAM in overleg beslist om Microsoft Azure te gebruiken als Cloud provider.

2.5.3 Azure services voor POC en uitrolfase

Het doel van dit laatste luik is om de verschillende componenten van een PaaS-oplossing die nodig zijn voor de POC en uitrol samen te stellen. Eerst worden mogelijke platformen van dataopslag, datatransformaties en model training, testing en beheer voorgesteld, om in paragraaf 2.5.3.4 een uiteindelijk keuze te maken voor de POC en uitrol.

2.5.3.1 Dataopslag

Voor de POC en uitrol zullen er grote hoeveelheden ongestructureerde data worden opgeslagen. Voor het bijhouden van ongestructureerde data zoals video's en foto's zijn Azure Blob Storage en Azure Data Lake Storage de twee beste opties.

Azure Blob Storage is geoptimaliseerd voor het bewaren van enorme hoeveelheden ongestructureerde data. Blob staat voor “binary large object”. Blob Storage is geschikt om duizenden gelijktijdige uploads, grote hoeveelheden videodata en constant groeiende logfiles te verwerken. Er zijn geen restricties op vlak van toegelaten bestandstypes in Azure Blob Storage.

Azure Data Lake Storage Gen2 combineert de mogelijkheden van Azure Data Lake Storage Gen1, een opslagplaats geoptimaliseerd voor analyse van een grote hoeveelheid data, met Azure Blob Storage. Azure Data Lake heeft dus naast de voordelen van de Blob Storage ook hoge beschikbaarheid, sterke consistentie en mogelijkheden tot noodherstel na ramp. Dit zorgt voor verbetering in prestaties, management en beveiliging, wat voor een eventueel productie systeem ook wenselijk is.

Beide diensten behouden de data zolang de abonnementskosten betaald worden. De prijs van de opgeslagen data is afhankelijk van de grootte, de toegankelijkheid en de schrijf-/leeshoeveelheid. De data toegankelijkheid kan opgedeeld worden in drie lagen: hot, cool en archived. Elke opslag-laag heeft een andere prijsklasse waarbij hot de duurste is en archived de goedkoopste. Het verschil zit in het type van harde schijven waarop de data bewaard wordt en bijgevolg de snelheid waarop jouw data beschikbaar wordt. Voor de hot laag zijn dat premium SSD schijven, voor de cool laag zijn dit standaard SSD schijven en voor de archived laag zijn dit HDD schijven. De data blijft te allen tijde beschikbaar, onafhankelijk van de gekozen laag.

Voor één terabyte (TB) aan data betaal je (in februari 2021) op de hot laag 110€ per maand, voor cool laag 32 € per maand en voor de archived laag slechts 12€ per maand. De prijzen hangen sterk af van de lees- en schrijfoperaties in de verschillende lagen. Veel schrijf-/leesoperaties op een archived laag kosten bijvoorbeeld meer dan op een hot laag.

2.5.3.2 Data transformaties

Na het capteren van de data dient deze getransformeerd te worden. In Azure kunnen twee componenten gebruikt worden die dit ondersteunen.

Azure Databricks is gebouwd op Apache Spark en is geoptimaliseerd voor het uitvoeren van data transformaties en machine learning taken. Het grote voordeel aan Azure Databricks is de mogelijkheid om in parallel berekeningen uit te voeren waardoor de verwerkingstijd daalt. Voor meer details verwijzen we graag naar de officiële webpagina van Azure Databricks⁴⁴.

Azure Machine Learning (Azure ML) is voornamelijk een omgeving voor het trainen, implementeren, automatiseren, beheren en opvolgen van machine learning modellen. Hiernaast, biedt de omgeving de

⁴⁴ <https://azure.microsoft.com/nl-nl/services/databricks/>

mogelijkheid tot het gebruik van notebooks waarin de data transformaties gedaan kunnen worden met bv. Python code.

2.5.3.3 Model training, testing en beheer

Tot slot zijn er drie geschikte Azure platformen beschikbaar voor het trainen, testen en beheer van AI modellen: Azure ML, Azure Databricks en Azure Container Instance.

Azure ML is een geschikte optie voor het trainen, testen en beheren van AI modellen. Azure ML is ook compatibel met populaire machine learning opensource-programma's zoals TensorFlow, scikit-learn, etc. Verder biedt Azure ML ook toegang tot Azure ML Studio, een designprogramma dat een 'drag en drop'-dienst aanbiedt. Dit programma laat de gebruiker toe om op een snelle, gebruiksvriendelijke manier modellen te creëren zonder zelf te moeten programmeren.

In Azure ML is er de optie om een processor te kiezen die het best aansluit bij de taak met keuze uit verschillende types CPU's (een processor die de computer van rekenkracht voorziet), GPU's (de grafische kaart van een computer en uitermate efficiënt voor Deep Learning) en FPGA's (een gespecialiseerde processor voor versnelde Deep Learning).

Bijkomend is er een annotatie tool aanwezig in Azure ML. Het annotatieproces werd besproken in paragraaf 2.3.4.1.

Tenslotte kan MLOps geïntegreerd worden. Dit laat het data science team, dat in Azure ML modellen bouwt, toe samen te werken met andere IT-professionals en de implementatie en levenscyclus in coöperatie te beheren. MLOps kan beschouwd worden als DevOps⁴⁵ voor machine learning. Voordelen hiervan zijn reproduceerbaarheid, validatie (bv. goede praktijken) en automatisering.

Naast het transformeren van data biedt **Azure Databricks** ook de mogelijkheid tot het bouwen, trainen, testen, managen en *deployen* van machine learning modellen op grote schaal via Databricks Runtime for Machine Learning (Databricks Runtime ML). Azure Databricks biedt ondersteuning voor Python, Scala, R, Java en SQL. Net zoals Azure ML is Databricks compatibel met TensorFlow, PyTorch en scikit-learn.

Een nadeel van Azure Databricks is de afhankelijkheid tegenover de omgevingen die Databricks zelf gecreëerd heeft. Deze omgevingen bevatten de meest populaire machine learning pakketten en zijn moeilijk te wijzigen. Machine learning is echter een snel evoluerend veld en nieuwe pakketten worden regelmatig geïntroduceerd en geüpdatet. Een oplossing zou zijn om deze pré-geconfigureerde pakketten te kunnen upgraden/downgraden. Aangezien deze wijziging moet doorgevoerd worden naar alle nodes binnen een cluster, maakt het deze oplossing tijds- en arbeidsintensief.

Een derde optie is het gebruik van **Azure Container Instances**. Containers zijn uitvoerbare software units waarin de code is verpakt samen met de libraries en dependencies. Dit zorgt ervoor dat de code overal kan worden

⁴⁵ Dev Ops is een aanpak dat software development (Dev) en IT operations (Ops) combineert. Het doel is om de ontwikkelings levenscyclus van IT-producten in te korten en voorziet een framework van een continue aanlevering van hoogstaande software producten.

uitgevoerd op elke computer, alsook in de Cloud. Er is geen nood voor virtuele machines of andere infrastructuur.

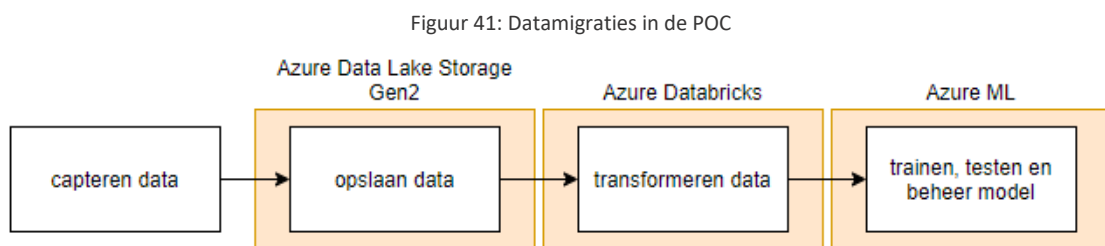
2.5.3.4 Conclusie datacaptatie en machine learning

Nu de mogelijke platformen van een PaaS-oplossing zijn besproken, kan een beslissing gemaakt worden over de platformen die gebruikt zullen worden in de POC en zal een blik geworpen worden op de vereisten waaraan de infrastructuur moet voldoen bij een toekomstige uitrol.

Op basis van de voor- en nadelen en de mogelijkheid voldoende in praktijk te testen tijdens de POC, stelt Ordina voor om volgende platformen te gebruiken tijdens de POC:

1. De Data Lake Storage Gen2 voor dataopslag
2. Azure Databricks voor data transformatie platform
3. Azure ML voor de model training, testing en beheer

Een grafische voorstelling van de datamigraties in de POC worden weergegeven in Figuur 41.



Bovenstaande platformen zijn ook geschikt voor een uitrolfase, maar niet optimaal daar het transformeren van data en het trainen van modellen nu op verschillende platformen plaats vindt. Deze twee stappen uitvoeren op eenzelfde platform zou efficiënter zijn om zo het aantal datamigraties en de bijhorende tijd te reduceren. Dit is ook wenselijk voor een efficiënter beheer van processen aangezien er minder componentenrekeningen te beheren zijn.

Tot slot is het belangrijk te vermelden dat de infrastructuur nodig voor de uitrolfase kan afwijken van het huidige voorstel vermits er met meerdere zaken rekening gehouden moet worden en het systeem aan bepaalde vereisten dient te voldoen. Zo dient bv. gedacht te worden of er automatisering vereist is voor nieuwe data en of een model slechts eenmalig of continu getraind moet worden.

2.6 CONCLUSIE TECHNISCHE HAALBAARHEID EN OPLOSSING

In het laatste deel van de technische studie, worden de voorgaande blokken samengebracht in een functionele workflow om vervolgens een architectuur op te bouwen.

2.6.1 Functionele flowchart

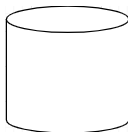
De architectuur wordt opgebouwd op basis van een functionele flowchart. Dit heeft als voordeel dat de relatie tussen de vereisten en de architectuur zichtbaar naar voor komt. De flowchart, zoals zichtbaar in Figuur 42 bevat een aantal type iconen, elk met hun eigen betekenis.



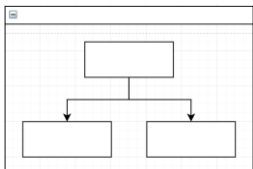
Dit icoon duidt op een trigger. Een trigger hoeft niet noodzakelijk manueel te zijn, maar voor de eenvoud wordt ervan uitgegaan dat dit wel zo is.



Dit icoon duidt een proces aan dat rekenkracht vereist. Hier zullen berekeningen gemaakt worden op basis van bestaande code. In dit specifieke geval zal dit veelal Python code zijn.



Dit icoon duidt op dataopslag. Het geeft echter geen duiding over het type van dataopslag. Welk type van dataopslag optimaal is zal later in de architectuur nauwer bekeken worden binnen de mogelijkheden van Azure.



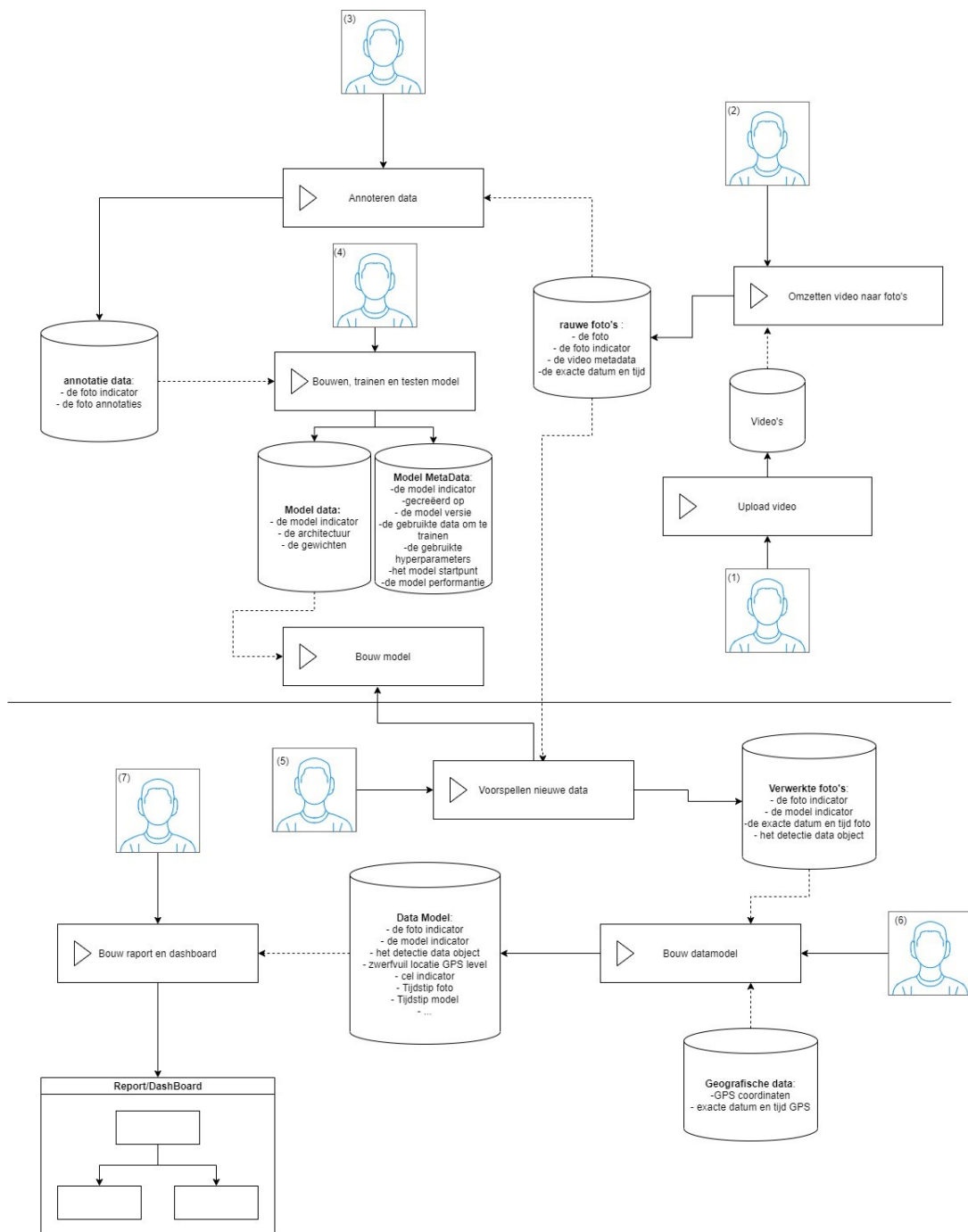
Het laatste icoon wijst op rapportering. De manier waarop gerapporteerd wordt is geheel vrij en kan gaan van visualisaties via een SaaS-oplossingen zoals Tableau of PowerBI tot simpele dataextracten in Excel of CSV formaat.

Het proces bestaat uit zes stappen:

1. Verzamelen en verwerken data
2. Annoteren data
3. Bouwen, trainen en testen model
4. Voorspellen nieuwe data
5. Bouw datamodel
6. Bouw rapport en dashboard.

Het is de bedoeling om de onderstaande functionele flowchart om te zetten naar een architectuur. Zo wordt de flowchart stap voor stap overlopen.

Figuur 42: Functionele flowchart



2.6.1.1 Verzameling en verwerking data

Als eerste stap zal er data in videoformaat verzameld worden. Er wordt gestart bij trigger 1 (rechts op de flowchart, boven de horizontale dikke lijn). Deze trigger zet een video upload in gang. Deze upload kan manueel gebeuren, maar zou ook volledig automatisch kunnen gebeuren door een directe connectie van de camera's met de Azure-omgeving. Als resultaat worden er video's opgeslagen in het systeem. Deze video's hebben geen unieke ID nodig, aangezien deze omwille van wettelijke beperkingen slechts voor een (zeer) beperkte tijd opgeslagen zullen worden (zie paragraaf 3).

Een tweede trigger, die zich in de flowchart in de rechterbovenhoek bevindt, kan men interpreteren als een actie die het verwerken van de rauwe videodata zal opstarten. Deze trigger kan eveneens een manuele of automatische (zodra nieuw videomateriaal beschikbaar) actie. In beide gevallen worden de originele video's omgezet naar individuele afbeeldingen en worden deze afbeeldingen door een *blurring* model gestuurd om gevoelige informatie zoals mensen en nummerplaten te maskeren. Na deze transformatie zal de video verwijderd worden. Deze individuele afbeeldingen moeten bij opslag minstens volgende data bevatten:

- *Foto ID*: Een uniek ID dat het systeem in staat stelt om doorheen het hele proces te achterhalen welke foto gebruikt werd.
- *Video metadata*: Alle data die van belang is voor het verwerken van de foto. Denk bv. aan het voertuig dat de video gemaakt heeft, de hoek die de camera vormde bij het maken van de video, etc.
- *Datum en tijd*: De exacte datum en tijdstip waarop de beelden zijn gemaakt om later de link met het GPS signaal van de ArduSimple simpleRTK2B (paragraaf 2.2.2.2) of een eventuele Galileo GPS (paragraaf 2.2.2.3) mogelijk te maken.

Het uiteindelijke resultaat van deze eerste stappen is het opbouwen van een database aan relevante afbeeldingen.

2.6.1.2 Annoteren data

Het proces om verworven afbeeldingen om te zetten naar trainingsdata wordt omschreven als het annoteren van afbeeldingen (paragraaf 2.3.1.2). Hierbij bekijkt een persoon de opgeslagen foto's en bepaalt hij/zij de *bounding boxes* van een bepaald type object op de foto (bv. het aanduiden van een sigaret). Deze data dient opgeslagen te worden volgens een bepaalde standaard zoals de COCO of PASCAL VOC standaard.

Het annoteren van de data wordt bovenaan in de flowchart gestart door trigger 3. Het resultaat hiervan is een nieuwe dataset waarbij de annotatie van de foto wordt opgeslagen samen met de "foto ID" van de verwerkte afbeelding. Op deze manier wordt de foto slechts op één plaats opgeslagen.

2.6.1.3 Bouwen, trainen en testen model

Met de trainingsdata beschikbaar kan een model gebouwd, getraind, gevalideerd en getest worden. Trigger 4 zet deze stap in werking. Twee mogelijke oorzaken kunnen leiden tot additionele training in een productie omgeving. Ten eerste is het denkbaar dat een nieuw model gereleased wordt op basis van recente ontwikkelingen binnen de computer visie wereld. Het is vanzelfsprekend dat een dergelijke release steeds manueel werk inhoudt, waarbij een data scientist moet tussenkomen. Daarnaast is het ook denkbaar dat een

nieuw model bekomen wordt als gevolg van doorgedreven training. Er kan immers veel nieuwe data beschikbaar zijn voor training, waardoor de accuraatheid van het model mogelijk verbeterd kan worden.

Een dergelijke train-validatie-test flow kan in theorie automatisch verlopen op basis van een aantal parameters zoals tijd, hoeveelheid nieuwe data, etc. In de praktijk blijft het echter nuttig om een manuele inschatting van de evaluatie te doen en testing te overschouwen.

De output van deze stap resulteert uiteindelijk in een model dat zich kenmerkt op basis van volgende parameters:

- *Model ID*: Uniek is voor elk model.
- *Model architectuur*: De architectuur van het betreffende model.
- *Model gewichten*: De getrainde gewichten die het model gebruikt.

Naast de zuivere model output, zijn er nog een hele reeks andere parameters die best worden bijgehouden. Dit is de model meta-data:

- *Gecreëerd op*: Wanneer het model gecreëerd is.
- *Model Versie*: De versie van dit model. De model versie wordt geüpdatet indien eenzelfde model verder getraind wordt.
- *Model afhankelijkheden*: De nodige software afhankelijkheden om dit model te kunnen gebruiken. In dit geval zal dit vooral betrekking hebben op gebruikte Python libraries.
- *Gebruikte data om te trainen*: Best wordt tevens bijgehouden op welke data precies getraind is om tot een (nieuwe) versie van het model te komen. Op deze manier kan nadien altijd achterhaald worden welke data het model gezien heeft en welke niet. Dit kan nuttig zijn in het kader van bepaalde zwerfvuilfracties. Indien men opmerkt dat een bepaalde fractie moeilijk detecteerbaar is voor een model, kan men kijken of er voldoende trainingsdata van die fractie gebruikt is. Om deze relatie te leggen wordt een verwijzing gemaakt naar de foto ID's van de gebruikte foto's.
- *Gebruikte hyperparameters*: Model *hyperparameters* zijn bepalend voor hoe het model traint. Het bijhouden van deze *hyperparameters* stelt data scientists later in staat na te gaan welke *hyperparameters* beter of slechter werkten en welke reeds gebruikt werden.
- *Model startpunt*: Als een model verder getraind wordt, is het zeer nuttig om een vertrekpunt te hebben. Het is bv. niet noodzakelijk dat een vierde versie van een model vertrekt van versie drie. Wanneer versie drie slechter bleek dan versie twee is het wenselijk voor het verder trainen van model versie vier te vertrekken van versie twee i.p.v. versie drie.
- *Model performantie parameters*: Prestatieparameters van het model zoals mAP, maar ook andere grootheden kunnen worden bijgehouden.
- *Testdataset referentie*: Wanneer er sprake is van een zekere prestatie, moet achterhaald worden op welke afbeeldingen de prestatie van toepassing is. Dit is vereist om de vergelijkbaarheid tussen modellen te kunnen verwezenlijken. Idealiter is er een testdataset, die enkel voor testing gebruikt wordt. Op die manier kan vergelijkbaarheid gegarandeerd worden.

2.6.1.4 Voorspellen nieuwe data

Nadat de data verzameld en getransformeerd is, het model gemaakt, getraind en gevalideerd is, zal het volgende proces plaatsvinden: “voorspellen nieuwe data”. Dit proces bestaat uit twee stappen: een *inference* stap en een *object tracing* stap.

Inference is een term die in de data science wereld wordt gebruikt om uit te drukken dat het model gebruikt wordt om nieuwe data te verwerken volgens wat het model eerder geleerd heeft aan de hand van de trainingsdata. De inference stap van het model wordt getriggerd door trigger 5 in de flowchart.

Om nieuwe data te verwerken is er nood aan twee zaken:

- De nieuwe data
- Een model

Bij gebruik van dit onderdeel dient een afbeelding of een selectie van afbeeldingen te worden aangeduid die verwerkt moet worden. Deze afbeeldingen worden geïdentificeerd op basis van verschillende eigenschappen zoals hun ID, video ID, tijdstip van de opname of andere beeldeigenschappen. Daarnaast dient men verplicht mee te geven welk model uit de model database men wenst te gebruiken. Deze referentie kan gebeuren op basis van de model ID. Idealiter wordt het model niet zelf gebouwd in de inference stap, maar wordt er via een *API* contact gelegd dat bij initialisatie van de connectie het model activeert. Daarna kunnen alle foto's via de actieve connectie verwerkt worden. Deze benadering heeft als voordeel dat de *API* geïntegreerd kan worden in andere applicaties en draagt bij aan de filosofie van micro-services. Dit zorgt voor een complete ontkoppeling tussen de training van een model en het gebruik ervan.

De output van de inference stap gaat rechtstreeks naar de tweede object tracing stap zodoende dubbeltelling te vermijden. Het object tracing algoritme neemt de sequentiële output van het object detectie model en geeft een ID aan elk uniek zwerfvuilobject. Elke foto in dit proces zal bijgevolg een model ID, foto ID, de exacte datum en tijd en een detectie data object hebben. Vb. bevat foto 971, 972 en 973 in hun “detectie data object” dezelfde zwerfvuilobject ID dan heeft het model in sequentiële beelden hetzelfde object gezien. Op deze manier wordt dubbeltelling vermeden.

Het detectie data object bevat een lijst van gedetecteerde zwerfvuilobjecten met volgende informatie:

- *Zwerfvuilobject ID*: Een unieke ID dat het stuk zwerfvuil identificeert.
- *Zwerfvuilobject fractie*: De zwerfvuilfractie waartoe dit stuk zwerfvuil behoort.
- *Zwerfvuilobject locatie pixel level*: De locatie waar het stuk zwerfvuil op de foto zichtbaar is. Dit zijn pixel coördinaten die overeenkomen met het centrum van de *bounding box*.

2.6.1.5 Bouw datamodel

Hoe het datamodel, niet te verwarren met het machine learning model, er exact moet uitzien is afhankelijk van hoe de OVAM de resultaten uiteindelijk wil verwerken en welke rapportering voor hen interessant is. Om deze reden valt de opbouw van een geschikt datamodel buiten scope van deze opdracht. Echter wordt dit wel voor architecturale redenen meegenomen.

Het bouwen van het datamodel wordt gestart door trigger 6. Deze trigger kan manueel of automatisch zijn. Bijvoorbeeld een dagelijkse verwerking van een nieuwe reeks beelden kan gepland worden waardoor dit automatisch zal getriggerd wordt. Naast de resultaten van het model zal in deze stap ook de geografische data, gecapteerd met ArduSimple of Galileo GPS, meegenomen worden. Op deze manier kan het gedetecteerde zwerfvuil gelinkt worden aan een bepaalde 10 bij 10 meter cel. Herinner dat de desbetreffende GPS-toestellen geen centimeterniveau accuraatheid halen. Toch zou de link met een 10 bij 10 meter cel mogelijk moeten zijn (paragraaf 0). Het linken van beide databronnen (verwerkte foto's en geografische data) is mogelijk door de exacte datum en tijd die zijn bijgehouden. Hoe de uitkomst van dit proces er exact zal uitzien is afhankelijk van de extra toegevoegde data die gebruikt wordt om de dashboards te maken. Volgende velden dienen minimaal aanwezig te zijn:

- *Foto ID*: Van welke foto deze observatie afkomstig is.
- *Model ID*: Het model die deze uitkomst heeft gegenereerd.
- *Het detectie data object*: Een lijst die per foto de gedetecteerde zwerfvuilobjecten weergeeft. Elk zwerfvuilobject heeft volgende informatie:
 - *Zwerfvuilobject ID*: Een unieke ID die het stuk zwerfvuil identificeert.
 - *Zwerfvuilobject fractie*: De zwerfvuilfractie waartoe dit stuk zwerfvuil behoort.
 - *Zwerfvuilobject locatie pixel level*: De locatie waar het stuk zwerfvuil op de foto zichtbaar is. Dit zijn pixel coördinaten die overeenkomen met het centrum van de *bounding box*.
- *zwerfvuil locatie GPS level*: De geografische locatie van het stuk zwerfvuil.
- *Cel ID*: De ID die aangeeft in welke 10 bij 10 meter cel het stuk zwerfvuil zich bevindt.
- *Tijdstip foto*: Tijdstip dat aangeeft wanneer deze foto genomen was.
- *Tijdstip model*: Tijdstip dat aangeeft wanneer deze foto verwerkt is door het AI model.

2.6.1.6 Bouw rapport en dashboard

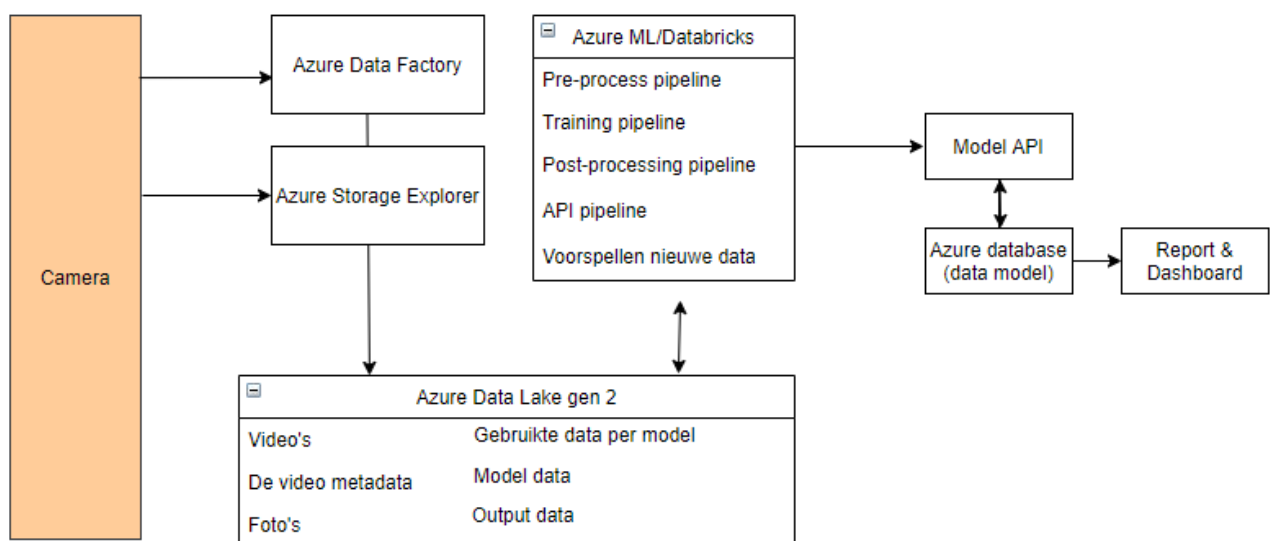
Het deel aangaande het bouwen van rapporten zit niet in scope van deze opdracht, maar is opgenomen omwille van architecturale volledigheid. Het maken of ophalen van de rapportage gebeurt in de flowchart op basis van trigger 7.

2.6.2 Architecturale mapping

Aan de hand van de functionele flowchart wordt een architectuur uitgetekend die men kan terugvinden in

Figuur 43. De voorgestelde architectuur geeft een overzicht van de gebruikte diensten. In dit onderdeel wordt het datamodel en rapportering buiten beschouwing gelaten.

Figuur 43: Architecturale mapping



Er zijn twee grote componenten die het grootste deel van de nodige functionaliteiten in vorige flowchart dekken. De eerste component is een Azure Data Lake Storage (ADLS), waar alle data zal worden opgeslagen. De tweede component is Azure Machine Learning (AML) of Azure Databricks waar de verschillende processen uitgevoerd worden. Deze services werden reeds gekaderd in paragraaf 2.5. Ook hier worden de aparte stappen uit dit vorige onderdeel besproken.

2.6.2.1 Verzameling en verwerking data

Voor het verzamelen van de data zullen de videobeelden worden opgenomen via een camera. Deze data zal opgeslagen worden in een Azure Data Lake gen 2. Er zijn twee mogelijkheden om de data op te laden:

1. De data kan manueel worden opgeladen via de Azure Storage Explorer.
2. De data kan via een Azure Data Factory worden ingeladen.

Eenmaal de data opgeslagen is, kan het door de *Pre-processing* pipeline. Deze pipeline zet automatisch video's om in *frames* en maskeert persoonlijke gegevens. Dit proces gebeurt via Azure ML of Azure Databricks. De output, zoals vermeld bij de functionele flowchart, wordt opgeslagen in de ADLS.

2.6.2.2 Annoteren data

De *frames* kunnen geannoteerd worden in AML (2.5.3.3). De annotatie kan ook uitbesteed worden aan een externe partij. De output wordt opgeslagen in de ADLS.

In de POC zal Ordina zelf annoteren, aangezien de ontwikkeling van een *API-endpoint*, nodig voor externe annotatie, buiten de scope van deze opdracht valt. Een *API* zorgt dat twee systemen met elkaar kunnen communiceren. In dit geval de OVAM en het annotatiebedrijf. Via een *API-endpoint* kunnen de afbeeldingen online uitgewisseld worden. Gezien het kostenvoordeel wordt in de uitrol gebruik gemaakt van externe annotatie (paragraaf 5.1.6.2).

2.6.2.3 Bouwen, trainen en testen model

Tijdens de POC worden de AI modellen ontwikkeld in AML. Op dit platform kunnen verscheidene berekeningsinstanties opgezet worden: CPU's en GPU's met elk een bepaald RAM-geheugen. Dit geeft flexibiliteit aan de POC om te onderzoeken wat de prestatie is van elk type hardware zonder zelf machines aan te kopen. De training van het model gebeurt manueel.

Na de uitrol kan het model verder verbeterd worden door her-training in AML. Dit kan zowel manueel als meer automatisch via een tweede pipeline. De trigger om deze trainingen te starten kan zowel event-gedreven zijn als periodiek. Bij een event-gedreven trigger is het bv. mogelijk om het trainen automatisch te laten starten nadat nieuwe gegevens zijn opgeladen. Indien nieuwe data slechts een paar keer per jaar of een paar keer per week worden opgeladen is dit niet heel nuttig en kan dit evengoed manueel gebeuren. Daarnaast is mogelijk om het trainen periodiek te triggeren op bepaalde tijdstippen. De planning kan ingesteld worden op meerdere malen per dag, week, maand of jaar. De voor- en nadelen van een manuele training of automatische training worden weergegeven in Tabel 12.

Tabel 12: Voor- en nadelen manueel en automatisch trainen

	Manueel	Automatisch: Event gedreven	Automatisch: Periodiek
Voordelen	• Trigger naar keuze • Enkel kosten bij de trigger	• Geen manuele interventie nodig • Automatisch	• Proces kan starten, ook wanneer de data nog niet volledig is ingeladen. • Tijdstip naar keuze
Nadelen	• Manuele actie moet plaatsvinden	• Proces wordt altijd getriggerd bij nieuwe data (ook wanneer dit niet wenselijk is)	• Proces draait altijd Wachten op de volgende cyclus vooraleer het proces getriggerd wordt.

Op basis van alle voor- en nadelen opteren we in de POC voor een semi-manuele, event-gedreven trigger. Door bestanden manueel op te laden in een "trigger" folder wordt het trainingsproces automatisch in gang gezet. Deze aanpak geniet van de voordelen van beide oplossingen. Deze opzet wordt aangeboden door Azure Data Factory.

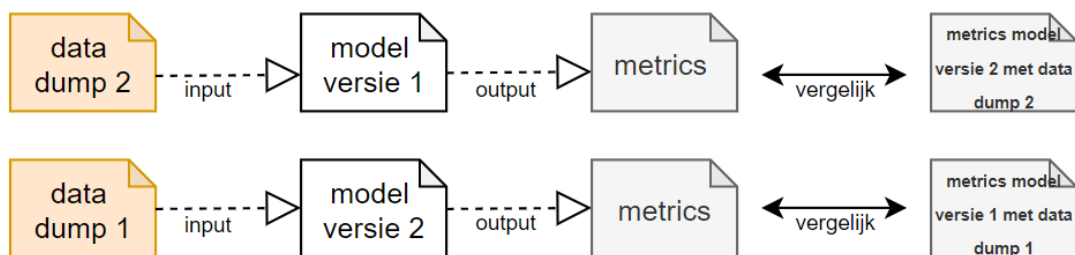
2.6.2.4 Voorspellen nieuwe data

Nadat een model getraind is, kan het uitgerold worden via onder andere Azure Container Services. Dit kan ook automatisch verlopen via een *deployment* pipeline. Voor elk model kan een *API* gemaakt worden om het specifieke model aan te kunnen spreken in de toekomst. Men kan zo nieuwe data door het model sturen om voorspellingen terug te krijgen. Dit kan zowel in AML, als in Azure Databricks. De uitkomst wordt opgeslagen in de ADLS.

2.6.2.5 Versiebeheer van het model

Doorheen het verloop van de meetcampagnes komt er meer en meer data ter beschikking. De OVAM kan beslissen om met deze data het model te her-trainen met als doel het model beter te maken. Om vergelijkbaarheid over de jaren heen te garanderen, moet er een manier zijn om de modellen objectief te vergelijken zodoende foute conclusies te voorkomen. Dit kan enerzijds door nieuwe data door het oude 'slechtere' model te sturen of anderzijds door oude data door het nieuwe 'betere' model te sturen.

Figuur 44: Methoden van objectieve vergelijkbaarheid tussen modellen



Zo kan de OVAM bijvoorbeeld uit nieuwe metingen afleiden dat het aantal blikjes is gestegen met 5% ten opzichte van een vorige meting. Het kan evenwel zijn dat het in de realiteit op een constant niveau is gebleven, maar dat het nieuwe model beter presteert in het herkennen van bv. blikjes dan voorheen. Door de oude metingen door het betere model te sturen, kunnen foute conclusies vermeden worden. Dit verklaart waarom het belangrijk is om de verschillende modellen met hun versies en metadata bij te houden. Het volstaat om het model via een *API* aan te spreken. Dit proces wordt vereenvoudigd door MLflow, een open source platform voor het beheren van een end-to-end machine learning levenscyclus. MLflow kan makkelijk geïntegreerd worden met AML.

3 JURIDISCHE HAALBAARHEID

De uitvoering van de beoogde opzet gaat gepaard met het capteren van beelden op de openbare weg en het opslaan van geografische locaties. Hierbij is captatie van persoonlijke gegevens onvermijdelijk, maar ongewild aangezien deze opdracht niet als doel heeft om persoonsgegevens te verwerken.

Enige verwerking van persoonsgegevens die wel plaatsvindt, betreft een loutere nevenwerking van de eigenlijke opdracht. Het betreft in het bijzonder de opname via camerabeelden van de beeltenis van mogelijke betrokkenen-passanten. Om hieraan tegemoet te komen wordt de toepassing voorzien van een filter die de originele beelden bewerkt en de beeltenis (in het bijzonder het gelaat en bij extensie de rest van het lichaam) van de passanten-betrokkenen op de beelden niet-identificeerbaar maakt door middel van korreling of *blurring*. Deze verwerking vindt haar rechtsgrond in artikelen 6.1.e en 6.3 Algemene Verordening Gegevensbescherming (AVG), vermits de verwerking noodzakelijk is ter uitvoering van het Samenwerkingsprotocol tussen het departement Omgeving, afdeling Handhaving en de OVAM dd. 30 januari 2019 en artikelen 10.3.2 tot en met 10.3.4 van het Decreet van 5 april 1995 houdende algemene bepalingen inzake milieubeleid.

Verder is de geo-lokalisatie van de OVAM-werknemers niet in real-time achterhaalbaar, maar kan deze wel achterhaald worden tijdens de analyse van de beelden in combinatie met de *timestamps*. Dit is echter een gevolg van de verwerking en maakt wederom geen deel uit van het verwerkingsdoel. Deze verwerking vindt haar rechtsgrond in artikel 6.b AVG, deze verwerking is immers noodzakelijk ter uitvoering van de arbeidsovereenkomst gesloten met de OVAM-werknemers.

Uit onze analyse volgt dan ook dat er geen gegevensbeschermingseffectbeoordeling (GBEB) vereist is, daar geen van de gevallen voorzien in artikel 35.3 AVG hier toepassing vindt, noch kan er sprake zijn van een hoog risico voor de rechten en vrijheden van natuurlijke personen ten gevolge van de verwerking van de persoonsgegevens (overeenkomstig artikel 35.1 AVG). Hoewel gebruikgemaakt wordt van innovatieve technologie (artificiële intelligentie), worden er voldoende technologische en organisatorische maatregelen genomen ter beperking van eventuele risico's, zelfs indien een *data breach* plaats zou vinden. Er worden geen bijzondere categorieën van gegevens verwerkt; er worden geen persoonlijke data van betrokkenen-passanten verwerkt, want de camera's zijn naar beneden gericht en mogelijke betrokkenen-passanten worden via artificiële intelligentie softwarematig zwart gemaakt; nummerplaten van voertuigen die op beeld komen worden op dezelfde wijze via artificiële intelligentie zwart gemaakt; de beelden en daarbij horende GPS-coördinaten worden niet in real-time verwerkt, maar pas achteraf; de toegang tot het systeem houdende de beelden en GPS-coördinaten is beperkt tot enkel zij die toegang vereisen; het verwerkingsdoel betreft de identificatie en telling van zwerfvuil. Er worden echter wel persoonsgegevens verwerkt van de werknemers, in het bijzonder hun GPS-locatie op een bepaald tijdstip. Dit kan afgeleid worden uit de correlatie tussen welke werknemer op een gegeven moment aan boord

is van een specifiek voertuig en de locaties die dit voertuig aflegt. Dit maakt echter geen deel uit van het verwerkingsdoel van de verwerkingsverantwoordelijke, maar is er een gevolg van. De geo-lokalisatie van de OVAM-werknemers is bovendien niet mogelijk in real-time, maar slechts achteraf. Hoewel geen GBEB nodig is, werd deze toch ingevuld en toegevoegd in de bijlage.

4 POC

Het doel van de POC is om de theoretische fundamenten die gelegd zijn in de haalbaarheidsstudie te testen en valideren. De POC heeft volgende doelstellingen:

1. Bepaling en evaluatie van de positie van de camera en de beeldkwaliteit
2. De bouw en evaluatie van een object detectie model
3. Onderzoeken van de functionaliteiten van de ZED
4. Object tracering

4.1 POSITIE CAMERA EN BEELDKWALITEIT

Zoals eerder vermeld zal in productie gewerkt worden met video's. Deze video's worden opgesplitst in *frames* en door het object detectie model gestuurd. Daar de accuraatheid van het AI model volledig af zal hangen van de beeldkwaliteit, is het onderzoeken van deze kwaliteit van cruciaal belang. Zonder een goede beeldkwaliteit zullen zelfs de meest geavanceerde object detectie modellen niet werken. Om de beeldkwaliteit te evalueren wordt gebruik gemaakt van een simpel maar efficiënt medium: de mens. Indien de mens er niet in slaagt om een object op een foto te duiden, dan zal het object detectie model hier wellicht ook niet in slagen.

4.1.1 Beeldkwaliteit GoPro

In paragraaf 2.1.2 werden verschillende modi van captatie besproken. Er werd uiteindelijk beslist met een flexibele oplossing verder te werken, gebruik makend van de zogenoemde grond-modi: een auto, fiets en bodycam. Deze drie wijzen van datacaptatie resulteren in verschillende invalshoeken, hoogten en snelheden. In dit opzicht worden drie camera posities gedefinieerd en geëvalueerd met behulp van een GoPro:

1. Voertuig – Hoog: De camera wordt gepositioneerd boven op een auto. De camera staat verder van het zwerfvuil, maar heeft een bovenaanzicht dat de detectie kan bevorderen.
2. Voertuig – Laag: De camera wordt gepositioneerd op de motorkap van een auto. De camera staat dicht bij het zwerfvuil, maar kan enkel het zwerfvuil in zijaanzicht filmen.
3. Bodycam: De camera wordt gepositioneerd op borsthoogte via een borstharnas. De opstelling laat toe om, in tegenstelling tot camera's gemonteerd op auto's, dicht bij het object te komen.

Op elk van deze hoogtes worden video's gemaakt in 2K en 4K kwaliteit. Eerdere projecten konden succesvol zwerfvuil detecteren met 2K camera's (paragraaf 2.3.3.1). Echter stond hier de camera loodrecht naar de grond gericht waardoor de afstand van de camera tot het zwerfvuil beperkt werd. Deze opstelling beperkt echter het beeld van de camera. Een test opstelling met de camera loodrecht naar de grond gericht toonde aan dat zelf

zwerfuit objecten in naastgelegen goten moeilijk tot niet in beeld kwamen. Met hogere voertuigen zou dit beeld mogelijk verruimt kunnen worden. Het blijft echter onwaarschijnlijk dat met dergelijke opstelling zwerfuit op aangrenzende voetpaden gedetecteerd zou worden. Combinatie met andere modi, zoals bodycam, in een zelfde cel stelt zware eisen aan de GPS accuraatheid. Om deze redenen werd dus bewust gekozen voor een alternatieve camera opstelling, zoals in meer detail besproken in paragraaf 2.4.2 en gevisualiseerd in Figuur 25. In de verdere opstelling van de POC is de camera dus horizontaler georiënteerd. Deze positionering doet de afstand tussen camera en afval toenemen. Omwille van dit verschil is de inclusie van een 2K versus 4K analyse opportuun. In wat volgt worden voor de drie camera posities *frames* uit een 2K en 4K video vergeleken. De verschillen worden visueel aangetoond.

4.1.1.1 Voertuig – Hoog

Deze situatie (Figuur 45) is gelijkaardig aan de beoogde manier van opnemen: het plaatsen van een camera op een (vuilnis)wagen. Beelden resulterend uit deze camera positie staan weergegeven op Figuur 46. Het is voor het blote oog onmogelijk om de zwerfuitobjecten te herkennen. Het verschil in beeldkwaliteit tussen 2K en 4K is nauwelijks zichtbaar. Al modellen bekijken de objecten echter op pixel niveau, wat betekent dat het model mogelijks nauwkeuriger is dan het blote oog. Figuur 47 toont dat inzoomen niet leidt tot een eenvoudiger objecten classificatie. Ook hier is er nauwelijks verschil tussen 4K en 2K kwaliteit, wat onverwacht is gezien 4K afbeeldingen dubbel zoveel pixels bevatten als de 2K variant. Mogelijke oorzaken worden besproken in 4.1.3.

Figuur 45: Constructie gelijkend op een vuilniswagen



Figuur 46: Vergelijking voertuig – hoog 4K (links) vs. 2K (rechts) kwaliteit.



Figuur 47: Ingezoomde vergelijking voertuig – hoog 4K (links) vs. 2K (rechts) kwaliteit.



4.1.1.2 Voertuig – Laag

In deze opstelling wordt de camera op de motorkap gezet. Eenzelfde conclusie als vorige opstelling kan gemaakt worden. Op de originele foto is het onmogelijk het zwerfvuil te duiden. Als er wordt ingezoomd is de kwaliteit opnieuw onvoldoende.

4.1.1.3 Bodycam

Werken met een bodycam heeft als voordeel dat men dicht bij zwerfvuilobjecten kan komen. Dit laat toe duidelijk foto's in bovenaanzicht te nemen. In tegenstelling tot de vorige camera posities, zijn de zwerfvuilobjecten op zowel de originele als de ingezoomde *frames* beter waarneembaar.

Figuur 48: Vergelijking bodycam 4K (links) vs. 2K (rechts) kwaliteit.



Figuur 49: Ingezoomde vergelijking bodycam 4K (links) vs. 2K (rechts) kwaliteit.



4.1.2 Beeldkwaliteit ZED

Naast de GoPro, wordt er ook gebruikt gemaakt van de ZED. De ZED filmt, in tegenstelling tot de GoPro, aan een resolutie van 2K.

Figuur 50 toont de originele en ingezoomde frame uit een ZED-video. De framekwaliteit van de ZED is slechter dan de framekwaliteit van de GoPro. De contouren van de zwerfvuilobjecten zijn niet meer intact. Dit is problematisch voor object detectie modellen. De ZED heeft geen optie om foto's te maken.

Figuur 50: Originele frame (links) en ingezoomde frame (rechts) van een ZED-video



4.1.3 Foto's versus frames

Uit paragraaf 4.1.1 bleek dat *frames* uit video's gecapteerd door camera's op voertuigen een slechtere beeldkwaliteit hadden dan de *frames* uit een bodycam video. De meest logische verklaring is dat het voor camera's moeilijker is om op hogere snelheid scherpere video's te maken. De wagen in deze proefopstelling reed gemiddeld 20 km/uur. De snelheid van een wandelaar is gemiddeld 5,5 km/uur.

Naast snelheid, zorgt het werken met *frames* in plaats van foto's ook voor een verschil. Wanneer een camera een foto trekt stelt de camera zich in een fractie van een seconde scherp. Video's hebben deze mogelijkheid niet, wat leidt tot een verlies aan beeldkwaliteit. Om deze verklaring te testen wordt een 4K-frame uit een video van een bodycam vergeleken met een 4K-foto van hetzelfde object. Beide afbeeldingen staan in Figuur 51. Het is duidelijk dat de rechter foto scherper en duidelijker is dan de linker frame.

Figuur 51: Vergelijking 4K-frame (links) vs. 4K-foto (rechts)



Gezien het duidelijk verschil tussen *frames* en foto's is het nodig ook het verschil tussen 4K en 2K foto's uit te zetten. Figuur 52 maakt duidelijk dat in foto's het verschil in resolutie zichtbaarder is dan in video's. Toch is het

zwerfvuilobject voor een mens in beide gevallen duidelijk herkenbaar. In dit opzicht is het beter om met de lagere resolutie te werken aangezien dit een grote impact heeft op de opslagkost en de trainingstijd.

Figuur 52: Verschil 4K foto (links) vs. 2K foto (rechts)



4.1.4 Burst modus

Een mogelijk alternatief is om in plaats van video's, aan voldoende snelheid foto's na elkaar te nemen. Dit wordt in fotografie termen ook wel '*burst*' genoemd. Op deze manier beelden capteren laat de camera toe zich scherp te stellen. De GoPro heeft een *burst* modus waarin vijf foto's worden getrokken in één seconde. Figuur 53 en Figuur 54 tonen aan dat *burst* modus betere kwaliteit aanlevert dan het nemen van video's. Toch is de geleverde kwaliteit nog altijd lager dan die van een echte foto (Figuur 52 vs.

Figuur 54). Een mogelijke verklaring is dat de scherpstellingstijd te kort is. Deze theorie kan echter niet in praktijk getest worden daar de GoPro niet toelaat de initiële vijf fps te verlagen.

Figuur 53: vergelijking frame uit 2K-video (links) vs. foto uit 2K-*burst* (rechts)



Figuur 54: Ingezoomde vergelijking frame uit 2K-video (links) vs. foto uit 2K-*burst* (rechts)



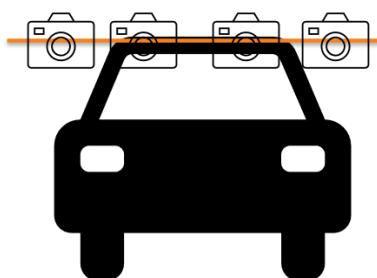
4.1.5 Conclusie

Bovenstaande maakte duidelijk dat de kwaliteit van *frames* uit video's van de GoPro en de ZED onvoldoende bleek. Het gebruik van betere camera's is bijgevolg aangewezen. Ordina geeft drie richtlijnen mee waar rekening mee gehouden moet worden bij de keuze van een toekomstige camera:

- Videocamera's zijn gemaakt om individuele beelden in elkaar over te laten gaan. Dit gebeurt door de camera veel foto's te laten trekken op een klein tijdsinterval (hoge fps). Hierdoor heeft de camera minder tijd om zich scherp te stellen, wat de framekwaliteit drastisch verlaagd (paragraaf 4.1.3). Een camera met *burst* modus daarentegen is specifiek gemaakt om snel foto's te maken aan hoge kwaliteit (paragraaf 4.1.4). Bijgevolg lijken deze camera's de beste optie voor een productiesysteem.
- Hoewel een te hoge fps de scherpstelling bemoeilijkt, zijn camera's met een lage fps niet aan te raden aangezien object tracering voldoende overlap nodig heeft tussen sequentiële *frames* (paragraaf 4.3.2).
- 2K resolutie blijkt, na testen, voldoende voor het systeem.

Een aanbevolen vervolgstap zou zijn om een aantal (*burst*) camera's uit te testen. Deze potentiële camera's kunnen naast elkaar gemonteerd worden zoals men kan zien in Figuur 55. Een mogelijke manier om te testen is de volgende: De camerabeelden worden door een object detectie model (blikjes of sigarettendoosjes model) gestuurd en de mAP-scores worden uitgezet en vergeleken. De camera met de beste resultaten, wordt de meest gepaste camera bevonden. De kost van deze vervolgstap wordt opgenomen in de financiële studie.

Figuur 55: Vervolgtraject camera's



4.2 AI – OBJECT DETECTIE

4.2.1 Het capteren van data

De eerste stap in het ontwikkelen van een AI model is het verzamelen van trainingsdata. In de POC kan dit op twee manieren:

1. Het gebruik van bestaande geannoteerde data.
2. Het zelf inwinnen van data.

Uiteraard geniet de eerste manier van inwinning onze voorkeur. Echter, open-source datasets zijn schaars, zeker gezien de specifieke context van zwerfvuil. Een niet-exhaustieve lijst staat in Tabel 6 (paragraaf 2.3.3.2).

Er is besloten geen gebruik te maken van de gepubliceerde datasets om onderstaande redenen. Bijgevolg is het noodzakelijk om zelf data in te winnen.

1. De Taco dataset bevat voornamelijk foto's waarin het zwerfvuilobject centraal staat op de foto. In een zwerfvuilmonitorsysteem is dit niet het geval.
2. De dataset van Let's do it foundation is reeds gebruikt voor de training van het Let's do it foundation model, welke het startpunt is voor de training van het Mask R-CNN model. Extra training op deze dataset is niet meer nodig.
3. De River plastic detection dataset bevat afbeeldingen van zwerfvuil in rivieren. Dit valt buiten de scope van de opdracht.
4. De Isaychris dataset is veelbelovend en zou mits bijkomende annotatie inspanningen omgevormd kunnen worden naar een goede dataset die representatief is voor dit project. Er is echter beslist om deze niet te gebruiken en ons toe te spitsen op bepaalde fracties, waarover later meer.

Voor elke fractie respectievelijk voldoende trainingsdata vergaren, annoteren en een goed model bekomen is niet het doel in deze POC. Het is louter te bedoeling om aan te tonen dat het mogelijk is om een aantal fracties goed te detecteren om zo de haalbaarheid in te schatten.

Er worden twee fracties gekozen uit de zwerfvuilfractielijst: blikjes en sigarettendoosjes. Sigarettendoosjes zijn gemakkelijk te detecteren, ze zijn namelijk gestandaardiseerd. Blikjes zijn in dit opzicht moeilijker te detecteren doordat ze in verschillende maten, vormen en kleuren voorkomen. Voor beide fracties wordt een apart model voorzien en de inspanning nodig om de modellen tot een hoog niveau te krijgen, wordt bijgehouden. Deze manier van werken laat toe een extrapolatie te maken naar de andere fracties omtrent de vereiste inspanning.

Datacaptatie gebeurt in deze context aan de hand van foto's zodoende duplicaten en bijgevolg *overfitting* (paragraaf 2.3.4.2) te vermijden. Op 25/03/2021 is Ordina data gaan capteren. Er werd zwerfvuil meegenomen en verspreid op verschillende ondergronden zoals te zien in

Figuur 56. De focus ligt op sigarettendoosjes en blikjes.

Figuur 56: De verschillende ondergronden waarop zwerfvuil werd gecapteerd



Ondanks het feit dat een 2K resolutie voldoende is (paragraaf 4.1), werden de foto's genomen in een 4K resolutie en later geconverteerd naar een 2K formaat. Het is immers niet mogelijk om een 2K resolutie naar een 4K resolutie te converteren. Bovendien werden de foto's genomen op een hoogte van 1,5 meter. Ordina verzamelde 1.568 foto's in acht uur tijd.

4.2.2 Het annoteren van data

De foto's genomen in het datacaptatie proces moeten geannoteerd worden alvorens bruikbaar als trainingsdata in een object detectie model. Zoals vermeld in paragraaf 2.3.1.2, gebeurt dit door *bounding boxes* te tekenen rond de zwerfvuilobjecten en deze te classificeren als een bepaald type fractie.

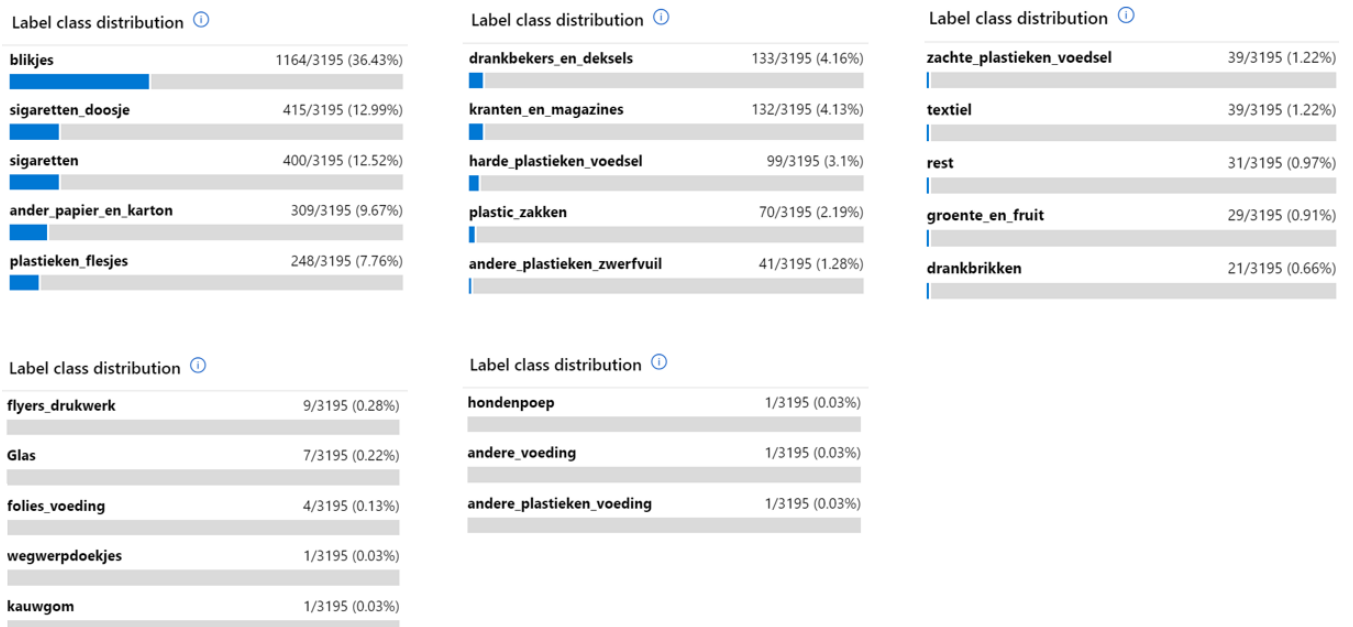
Er zijn twee opties i.v.m. het annoteren van data:

1. Zelf annoteren: De afbeeldingen worden manueel geannoteerd.
2. Outsourcing: Data annotatie kan aan een derde partij worden uitbesteed. Hiervoor moet een *API-endpoint* opgezet worden.

Aangezien de hoeveelheid trainingsdata in de POC gelimiteerd is, wordt ervoor gekozen om zelf te annoteren daar de opzet van een *API-endpoint* meer tijd in beslag zou nemen dan het annotatieproces zelf uit te voeren. De tool die gebruikt wordt voor dit annotatieproces is de Azure ML labeling service (paragraaf 2.5.3.3).

Maar 51,69% of 854 van de gemaakte foto's werd geannoteerd doordat veel data duplicaat aanwezig was, wat zou leiden tot *overfitting*. Er werden immers meerdere foto's genomen van dezelfde situatie. In retrospectief, valt het meermaals capteren van dezelfde situatie te vermijden gezien de vereiste manuele filtering achteraf. Doordat foto's soms meerdere fracties omvatte, werden in totaal 3.195 *bounding boxes* opgeleverd. Figuur 57, toont de fractiedistributie hiervan.

Figuur 57: Distributie zwerfvuil *bounding boxes*



4.2.3 Trainen van de object detectie modellen

In paragraaf 2.3.3.1 werd beslist om twee modellen te trainen: Het Let's do it foundation model en het YOLO v5 model. Alleen het Let's do it foundation model is voorgetraind op zwerfvuil. Het YOLO v5 model is voorgetraind op de COCO-benchmark taak, maar heeft nog geen kennis van zwerfvuil.

Gezien de lange trainingstijd (paragraaf 2.3.4.2.1), worden beide modellen 50 *epochs* getraind op blikjes data. Vervolgens wordt empirisch het beste model gekozen door ze te testen op ongeziene data (test dataset). Een aantal test foto's worden afgebeeld in

Figuur 58.

Figuur 58: Vergelijking Let's do it foundation model (links) vs. YOLO v5 model (rechts)





Beide modellen slagen erin om in een beperkt aantal *epochs* blikjes te detecteren. Beiden detecteren een aantal FP's (paragraaf 2.3.2). Het YOLO v5 model maakt hierin minder fouten en wordt daardoor als beste model bevonden.

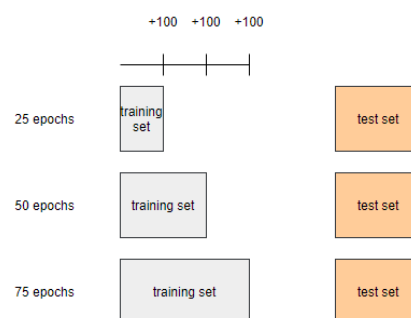
Dit is enigszins verassend. Het Let's do it foundation model is namelijk een double-shot model en zou in theorie beter moeten presteren. Ook had dit model al eerder zwerfvuil gezien in tegenstelling tot het YOLO v5 model. Bovendien is het trainings- en inferenceproces van het YOLO v5 model sneller op één TeslaK80 GPU:

- **Trainingsproces:** Het duurt gemiddeld 1,2 seconden om één afbeelding één *epoch* te trainen. De Mask R-CNN doet het met 1,83 seconden in dit opzicht slechter.
- **Inferenceproces:** Het YOLO v5 model kan 15 foto's per seconde verwerken. Het Mask R-CNN model kan er vier per seconde verwerken.

Verder gaan met het YOLO v5 model lijkt dus de beste optie.

Er is bewust gekozen om het trainingsproces van het blikjes en sigarettendoosjes model anders te laten verlopen dan conventioneel voorgeschreven. Het doel is om een inschatting te maken over de nodige inspanning om een goed werkend model te krijgen. Om dit te bereiken wordt de trainingsopzet, geïllustreerd in Figuur 59, gehanteerd. In de eerste 25 *epochs* wordt getraind op 100 foto's van het desbetreffende zwerfvuil. In de 25^{ste} *epoch* wordt het tussentijds model getest op de testset en de mAP resultaten opgeslagen. Vervolgens worden 100 extra datapunten toegevoegd aan de trainingsset. Wat het totaal aantal foto's in de trainingsset op 200 brengt. Opnieuw wordt er 25 *epochs* getraind en in de 50^{ste} *epoch* de mAP resultaten opgeslagen. Dit proces herhaalt zich tot alle trainingsdata gebruikt is. Op deze manier kan de impact van extra data op de mAP worden geanalyseerd. Het is belangrijk te vermelden dat niet elke foto zwerfvuil bevat. Op deze manier leert het model ook wanneer er geen zwerfvuil aanwezig is. In dit project wordt ongeveer 40% van de trainingsset van een model gevuld met foto's zonder een zwerfvuilobject.

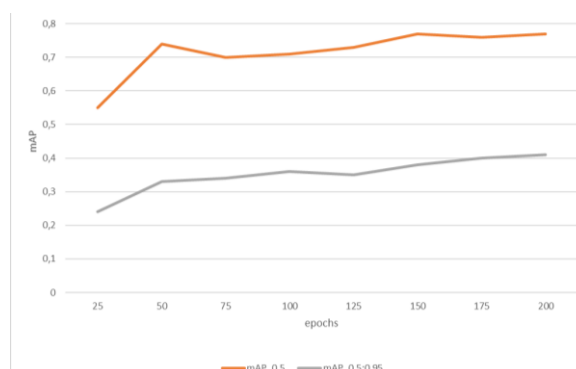
Figuur 59: Trainingsopzet



4.2.3.1 Resultaten blikjes model

Het blikjes model doorloopt de trainingsopzet acht keer. Dit betekent dat er getraind wordt op 800 foto's vanaf *epoch* 175. 33% van de foto's bevatten geen zwerfvuilobject. Bijgevolg zijn de 1164 *bounding boxes* van blikjes (paragraaf 4.2.2) verdeeld over 536 foto's. De resultaten van het trainingsproces staan in Figuur 60. Deze resultaten zijn behaald op een testset die 97 foto's bevat. Deze is manueel samengesteld en bevat afbeeldingen van situaties die nog niet in de trainingsset aanwezig waren. Een aantal foto's van de testset met de voorspellingen van het model staan in de bijlage (paragraaf 7.3).

Figuur 60: Resultaten blikjes model



De grafiek vertoont een afnemende stijging doorheen de tijd (logaritmisch). Dit is normaal gedrag aangezien het na verloop van tijd moeilijker wordt om het model te verbeteren. De mAP_0.5 (paragraaf 2.3.2) wordt vlakker rond *epoch* 150 (600 afbeeldingen). Dit wil zeggen dat het model met de nieuwe informatie die de extra 200 afbeeldingen verschaftte, geen extra zwerfvuil kon detecteren of FP te vermijden in de testset. In tegenstelling tot de mAP_0,5, blijft de mAP_0,5:0,95 wel stijgen. Dit wil zeggen dat het model beter wordt in de lokalisatie (Figuur 61): de voorspelde *bouding boxes* overlappen meer met de echte *boundig boxes* (paragraaf 2.3.2).

Figuur 61: De invloed van de mAP_0,5:0.95 metriek



In de context van dit project is het uitermate relevant om verder te trainen aangezien de mAP_0,5:0.95 metriek blijft stijgen. Object lokalisatie (paragraaf 2.4.2) en object tracering (paragraaf 2.4.3) zijn compleet afhankelijk van een goede locatie van de *bouding box*.

Deze resultaten tonen *niet* aan dat het blikjes model nooit verder zal geraken dan een mAP_0.5 van 77%. Het bouwen van een object detectie model verloopt in verschillende iteraties. Een goede methode is om te kijken naar de fouten die het model maakt (

Figuur 62) en gericht mogelijkheden tot verbetering te definiëren:

1. Probleem: In 50% van de gevallen detecteert het model een sigarettendoosje als een blikje.
Actie: Voeg extra foto's van sigarettendoosjes toe aan de trainingsset zonder een *bounding box*. Op deze manier wordt aan het model duidelijk gemaakt dat sigarettendoosjes geen blikjes zijn.
2. Probleem: Het model heeft moeite met het detecteren van blikjes die relatief dicht bij de camera staan.
Actie: Voeg extra foto's toe aan de trainingsset van blikjes die dicht bij de camera staan.
3. Probleem: Het model heeft moeite met het detecteren van blikjes die relatief ver van de camera verwijderd zijn.
Actie: Voeg extra foto's toe aan de trainingsset van blikjes die ver van de camera staan.

Figuur 62: Fouten blikjes model



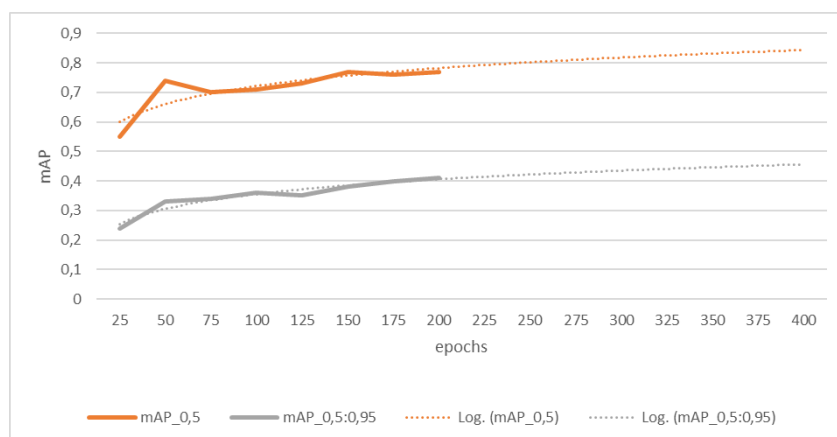
Dit demonstreert het belang van een goed gediversifieerde dataset die alle mogelijke situaties die kunnen voorkomen in een uitrol bevatten. Ook zal het van belang zijn om aan het model het verschil tussen objecten die sterk op elkaar lijken aan te tonen. Een robuuste manier om dit aan te pakken, is om eenzelfde model te ontwikkelen voor gelijkaardige objecten. Op deze manier wordt aan het model geleerd om de verschillen tussen de zwerfvuilobjecten te detecteren. Hierbij moet wel opgelet worden om niet te veel fracties te combineren in één model, om de specificiteit van het model niet te schaden. Immers hoe meer fracties opgenomen worden, hoe slechter de mAP wordt, zoals beschreven in paragraaf 2.3.3.

Verder rest nog de vraag hoeveel bijkomende inspanning nodig is om het blikjes model uitrol waardig te maken. Deze vraag hangt af van de eisen van de OVAM en zal proefondervindelijk moeten bepaald worden eenmaal een volledig systeem klaar is. Er zal dan moeten worden uitgereden naar test-10 bij 10 meter cellen met de opzet om de fout die de modellen maken in kaart te brengen en deze uit te zetten ten opzichte van de fouten gemaakt door manuele tellers. Dit vraagstuk wordt niet verder onderzocht in deze studie.

Om een inschatting te kunnen maken over het budget nodig om 26 fracties te detecteren met goed werkende object detectie modellen, wordt een $mAP_{0,5}$ van 85% gehanteerd. Ordina is ervan overtuigd dat een mAP van 85% voldoende is voor een efficiënte werking in een productieomgeving.

Figuur 63 toont hoeveel extra datapunten nodig zijn om een $mAP_{0,5}$ van 85% te behalen. Deze voorspelling is gebaseerd op de observatie dat de data een logaritmisch patroon vertoont. Rond *epoch* 400 zal een $mAP_{0,5}$ van 85% worden behaald, wat overeenstemt met 1500 afbeeldingen. Wellicht zal de mAP hoger uitvallen als rekening gehouden wordt met de hierboven vernoemde actiepunten bij het toevoegen van extra trainingsdata. Deze cijfers worden gebruikt om een extrapolatie te maken voor andere fracties.

Figuur 63: Projectie mAP blikjes model

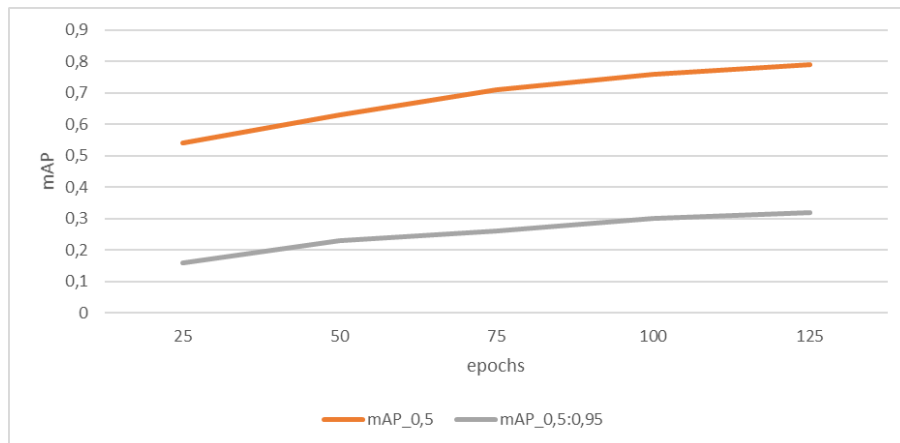


4.2.3.2 Resultaten sigarettendoosjes model

De resultaten van het sigarettendoosjes model staan op Figuur 64. Er werd getraind op 500 foto's waarvan 40% geen sigarettendoosje bevat (415 sigarettendoosjes verdeeld op 296 foto's). In 125 *epochs* behaalt het model een $mAP_{0,5}$ van 78%. Ook hier is duidelijk een logaritmisch verband aanwezig. In vergelijking met het blikjes model heeft het sigarettendoosjes model minder afbeeldingen nodig om gelijkaardig te presteren. Dit is naar verwachting gezien sigarettendoosjes relatief gestandaardiseerd zijn.

De $mAP_{0,5:0,95}$ is lager dan het blikjes model. Dit is niet verwonderlijk aangezien het blikjes model 75 extra *epochs* heeft getraind. Een aantal foto's van de testset met de voorspellingen van het model staan in bijlage (paragraaf 7.3).

Figuur 64: Resultaten sigarettendoosjes model



Opnieuw worden de fouten van het model geëvalueerd. Figuur 66 toont hoeveel extra datapunten nodig zijn om een mAP_0,5 van 85% te behalen. Deze voorspelling is opnieuw gebaseerd op de observatie dat de data een logaritmisch patroon vertoont. Rond *epoch* 175 zal een mAP_0,5 van 85% worden behaald, wat overeenstemt met 700 afbeeldingen. Deze cijfers worden gebruikt om een extrapolatie te maken. Op Figuur 65 staan drie foto's waar het model fouten maakt. De pijlen duiden de werkelijke sigarettendoosjes aan. Tijdens het vergaren van extra beeldmateriaal wordt best rekening gehouden met onderstaande actiepunten.

- Probleem:** Het model heeft moeite met het detecteren van sigarettendoosjes die relatief ver van de camera verwijderd zijn.

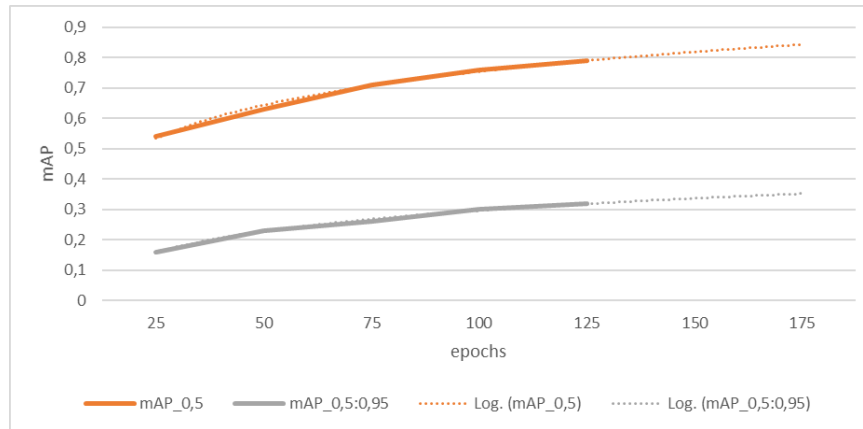
Actie: Voeg extra foto's toe aan de trainingsset van sigarettendoosjes die ver van de camera staan.
- Probleem:** Het model heeft moeite met het detecteren van sigarettendoosjes die half zichtbaar zijn.

Actie: Voeg extra foto's toe aan de trainingsset van sigarettendoosjes die half zichtbaar zijn.

Figuur 65: Fouten sigarettendoosjes model



Figuur 66: Projectie mAP sigarettendoosjes model



4.2.4 Extrapolatie trainingsresultaten

Nu geweten is wat de vereiste inspanning is om een blikjes en sigarettendoosjes model te bouwen met een mAP_0,5 van 85%, kan een extrapolatie gemaakt worden naar de overgebleven fracties. De verschillende fracties worden uitgezet op een Fibonacci-moeilijkheidsschaal: objecten worden geordend op basis van de verwachte moeilijkheid die het object detectie model zal ondervinden om het desbetreffende zwerfvuilobject juist te detecteren. De verwachte moeilijkheid werd bepaald door input van zowel domein experts als data scientists. De toegekende cijfers zijn Fibonacci getallen. Deze laten toe inspanning te schatten volgens de scrum methodologie⁴⁶. Twee zwerfvuilobjecten worden in onderling overleg met de OVAM buiten beschouwing gelaten: hondenpoep en andere plasticen verpakking niet voeding of drank (alles wat niet duidelijk identificeerbaar is als verpakking van voeding). Dit brengt het totale aantal te detecteren zwerfvuilobjecten op 24.

⁴⁶ <https://www.scrum.nl/blog/waarom-schatten-punten-en-niet-uren/>

Tabel 13: Extrapolatie moeilijkheid 24 zwerfvuilfracties

fractie	moeilijkheid
Ballonnen en stokjes van ballonnen	2
Sigaretendoosjes	2
Kranten en magazines	2
Peuken	3
Plastieken flessen tot drie liter	3
Drankbrikken	3
Drankbekers en deksels	5
Blikjes	5
Flyers, drukwerk, promotiemateriaal, autoruitkaartjes	5
Wegwerpdoekjes voor persoonlijke hygiëne	5
Plastic zakken	5
Afval van groente- en fruit en andere voeding	8
Kauwgom	8
Harde plastieken voedselverpakkingen voor eenmalig gebruik (bv. dozen)	8
Zachte plastieken voedselverpakkingen voor eenmalig gebruik (bv. wikkels, zakjes)	8
Andere serviceverpakkingen voor voeding (zowel papieren als folieverpakkingen)	8
Folies	8
Glas (flessen, bokaal e.a., splinters en gehele stukken)	8
Textiel en kledij (sokken, schoenen, handschoenen...)	8
Ander verpakking voeding	13
Ander plastieken zwerfvuil, niet verpakking (bv. Balpen, speelgoedjes...)	13
Ander papier en karton	13
Grote stukken zwerfvuil (minimumafmeting: groter dan drie literfles)	13
Andere (restcategorie wat nergens anders onder valt)	13

Er wordt een exponentiële groeicoëfficiënt gehanteerd om de extrapolatie te maken naar de andere moeilijkheid niveaus. Om deze factor te definiëren voor de hoeveelheid trainingsdata wordt volgende vergelijking opgelost:

Vergelijking 8: berekening groeicoëfficiënt hoeveelheid trainingsdata

$$700 * \text{groeicoëfficiënt} = \frac{1.500}{\text{groeicoëfficiënt}}$$

De uitwerking van vorige vergelijking resulteert in de factor 1,4638. Dezelfde werkwijze wordt gehanteerd om de groeifactor van de trainingstijd te bepalen (1,5118). Tabel 14 staat het geprojecteerd aantal trainingsdata en de vereiste trainingstijd voor elk moeilijkheidsniveau. Bijvoorbeeld het aantal trainingsdata voor moeilijkheidsniveau drie en 13 wordt respectievelijk berekend door: $700 \times 1,4638^{(1)} = 1.025$ en $700 \times 1,4638^{(4)} = 3.214$.

Tabel 14: Aantal trainingsdata en trainingstijd per moeilijkheidsniveau

moeilijkheidsniveau	Aantal trainingsdata	Trainingstijd (epochs)
2	700	175
3	1.025	265
5	1.500	400
8	2.196	605
13	3.214	914

Passen we deze cijfers toe op Tabel 13, dan is de vereiste hoeveelheid trainingsdata gelijk aan 46.312 ($700 \times 3 + 1025 \times 3 + 1500 \times 5 + \dots$) en wordt er in totaal 12.728 *epochs* getraind. De totale trainingstijd in uren kan berekend worden door Vergelijking 9:

Vergelijking 9: Berekening trainingstijd

$$\text{trainingstijd (uren)} = (\# \text{ uur om één 2K foto één epoch te trainen} \times \# \text{ data} \times \text{aantal epochs})$$

Zoals eerder aangegeven duurt het 1,20 seconden om een YOLO v5 model en 1,82 seconden voor een Mask R-CNN model op één TeslaK80 GPU te trainen voor één *epoch*. Als bovenstaande formule wordt toegepast, wordt er in totaal 9,833 uur of 409,69 dagen getraind met een YOLO v5 model op één TeslaK80 GPU. Met een Mask R-CNN wordt er 621,36 dagen getraind.

Volgende methodes kunnen deze trainingstijd reduceren:

- In de Cloud kan er voor elk model een aparte TeslaK80 GPU worden opgezet waardoor de trainingsprocessen van de modellen in parallel uitgevoerd kunnen worden.
- In de Cloud is er de optie om te trainen op vier TeslaK80 GPU's. Theoretisch gezien zou de trainingstijd bijgevolg met een factor vier moeten dalen.

In praktijk zal men waarschijnlijk niet 24 modellen tegelijk trainen. Toch wordt ter vereenvoudiging van de financiële studie uitgegaan dat elk model sequentieel wordt gebouwd. In realiteit zal dit echter deels parallel lopen. Zo kan een data scientist bijvoorbeeld beslissen om tijdens het trainingsproces van het sigaretten model over te schakelen naar het bouwen van het fruit/groente model.

Wel wordt uitgegaan van de training op vier TeslaK80' GPU's. Dit reduceert de trainingstijd met een factor 4 tot 102,42 dagen voor het YOLO v5 model en 155,34 dagen voor het Mask R-CNN model.

Tot slot gaan we dieper in op moeilijkheidsniveau 13. Onder dit niveau vallen de rest categorieën die eerder vaag gedefinieerd zijn. Zonder deze specifieke aflijning is het voor een AI model onmogelijk om dit te detecteren. Stel je traint een model op de fractie *'ander plasticen zwerfvuil, niet verpakking'* door te trainen op afbeeldingen van balpennen en speelgoed. Dan zal het model andere zwerfvuilobjecten die ook behoren tot deze fractie niet herkennen (vb. Tipp-ex). Het model zoekt louter naar de objecten die gezien zijn in de trainingsfase. Samenvattend kan gesteld worden dat de fracties vallend onder moeilijkheidsniveau 13 preciezer gedefinieerd dienen te worden. Indien dit gebeurt, kan het zijn dat de fractie daalt in moeilijkheidsniveau waardoor de geprojecteerde kosten lager zullen uitvallen en de haalbaarheid zal stijgen.

4.2.5 Inference op video's en burst

De modellen uit voorgaand onderdeel zijn getraind op fotomateriaal. Bij uitrol zal echter gewerkt worden met *frames* resulterend uit video's of *bursts*. Bijgevolg moet getest worden of de object detectie modellen nog altijd zwerfvuilobjecten kunnen detecteren gezien we eerder reeds zagen dat *frames* uit video's of foto's uit *bursts* van mindere kwaliteit zijn (paragraaf 4.1). Beide manieren werden getest en door het model gestuurd (inference). De resultaten hiervan zijn terug te vinden in Figuur 67 en slagen er in blikjes alsnog te detecteren.

Dit impliceert niet dat er geen andere uitdagingen meer zijn m.b.t. de camera. De blikjes staan nog altijd relatief dicht tegen de camera. De impact van het kwaliteitsverlies door video's of *burst* zal meer doorwegen bij ver gelegen objecten. Dit dient terug getest te worden nadat getraind wordt op meer data van ver gelegen objecten (actiepunt paragraaf 4.2.3.1).

Figuur 67: Inference op een frame uit een video (links) en een foto uit *burst* (rechts)



4.2.6 Conclusie

Twee object detectie modellen werden ontwikkeld: een sigarettendoosjes- en een blikjesmodel. Beiden behaalden goede mAP scores en hebben potentieel om beter te scoren mits goede data wordt toegevoegd. Het is dus mogelijk om met AI zwerfvuil te detecteren op foto's. Ook op de minder kwalitatieve *frames* en *burst* kon het model zwerfvuil aanduiden.

Met deze opgedane kennis kon een inschatting worden gemaakt over de hoeveelheid trainingsdata, nodig om 24 fracties succesvol te detecteren. Dit laat toe de modelontwikkelingskosten in de financiële studie te budgetteren (paragraaf 5).

4.3 ZWERFVUILTELLING

Zoals eerder aangehaald, dienen de *frames* die door de object detectie modellen werden gestuurd tevens gelinkt worden aan een geografische coördinaat en moeten dubbeltellingen vermeden worden (paragraaf 2.4). Object lokalisatie op basis van stereo-fotogrammetrie kan in combinatie met een magnetometer een antwoord bieden op het eerste probleem. Omwille van de beperkte GPS-accuraatheid zal voor het probleem van dubbeltelling echter een andere aanpak moeten voorzien worden, nl. object tracering.

In dit luik wordt eerst aangetoond dat object lokalisatie mogelijk is. Hiervoor wordt de ZED gebruikt. Vervolgens wordt het SimpleSort algoritme (paragraaf 2.4.3) geïmplementeerd om aan te tonen dat ook object tracering mogelijk is.

4.3.1 Object lokalisatie

Zoals besproken in paragraaf 2.4.2 kan de ZED ingeschakeld worden voor verschillende doeleinden:

- De ruimtelijke oriëntatie van de camera ten opzichte van de aarde via een IMU. Of na vereenvoudiging de oriëntatie van de kijkrichting van de camera ten opzichte van het noorden. Dit wordt geïmplementeerd in paragraaf 4.3.1.1.
- De relatieve locatiebepaling van gedetecteerde objecten ten opzichte van de camera, gebruikmakend van stereo-fotogrammetrie. Dit wordt in paragraaf 4.3.1.2 geïmplementeerd.

Met deze componenten kan een nauwkeurige locatie van zwerfvuilobjecten worden bepaald (paragraaf 4.3.1.3).

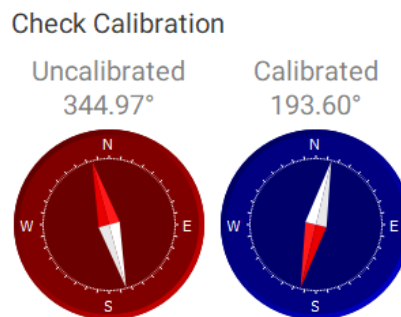
4.3.1.1 De ruimtelijke oriëntatie van de camera

De ruimtelijke oriëntatie van de camera wordt in navolging van paragraaf 2.4.2.2.2 bepaald, gebruikmakend van de magnetometer. De SDK van de ZED geeft de mogelijkheid om zowel gekalibreerde als niet-gekalibreerde data van de magnetometer op te vragen. Na enkele testen wordt snel duidelijk dat het essentieel is om gebruik te maken van de gekalibreerde data. De niet-gekalibreerde data varieert namelijk niet meer dan 30° rond het noorden ook als naar het zuiden gericht wordt.

Vooraleer gebruik gemaakt kan worden van de gekalibreerde data, moet de ZED gekalibreerd worden. Dit gebeurt door middel van de ZED Sensor View, een grafische interface om te connecteren met sensoren in de ZED. Na kalibratie zijn duidelijke verschillen waar te nemen tussen de gekalibreerde en niet-gekalibreerde data zoals zichtbaar in

Figuur 68.

Figuur 68: Kalibratie van de magnetometer



Opvragen van de gekalibreerde data van de magnetometer levert waarden voor de magnetische inductie, uitgedrukt in Tesla. Voor elk van de drie lokale assen van de camera, zoals voorgesteld op Figuur 36, wordt een waarde bekomen. Deze waarden geven aan hoe sterk het magnetisch veld is in de betreffende richtingen. Op deze manier kan drie dimensionaal bepaald worden waar het magnetische noorden zich bevindt.

Omdat, zoals in paragraaf 2.4.2.2.2 besproken, er uitgegaan wordt van een kijkrichting die evenwijdig loopt met het aardoppervlak, kan gebruik gemaakt worden van de waarde in X en Z richting om het noorden te bepalen via Vergelijking 10.

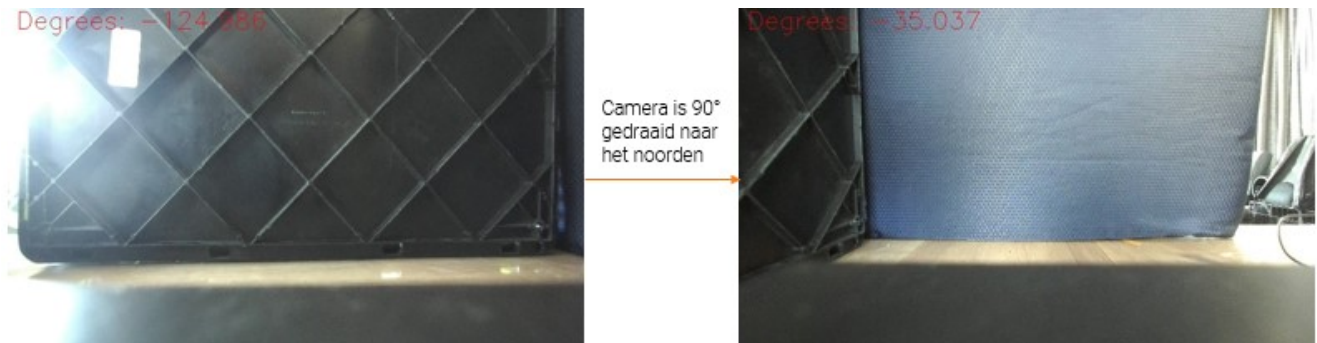
Vergelijking 10: Omzetting magnetische inductie naar een hoek in graden

$$\theta [^\circ] = \arctan(x/z) * 180 / \pi$$

Een voorbeeld kan teruggevonden worden in

Figuur 69. De camera werd in een eerste positie geplaatst, willekeurig ten opzichte van het noorden. De magnetometer gaf hierbij aan dat deze eerste positie onder een hoek van -125° staat ten opzichte van het noorden. Vervolgens werd de camera 90° gedraaid richting het noorden. De magnetometer geeft vervolgens aan dat de hoek met het noorden nog -35° bedraagt. Het verschil tussen beide hoeken is 90° . Deze proef toont de mogelijkheid om een magnetometer te gebruiken voor de camera oriëntatie.

Figuur 69: Camera oriëntatie ten opzichte van het noorden



Meermaals dezelfde oriëntatie opmeten zonder de camera te draaien leidt tot afwijkingen tussen 0° en 3°. Voor locatiebepaling dient dus rekening gehouden te worden met een fout van 3° aangaande de camera oriëntatie ten opzichte van het noorden.

4.3.1.2 De relatieve locatiebepaling van gedetecteerde objecten

Om de mogelijkheid tot relatieve locatiebepaling via stereo-fotogrammetrie te demonstreren wordt gebruik gemaakt van de ZED. De ZED zal een puntenwolk genereren die getoetst kan worden aan de realiteit. Ter validatie van deze puntenwolk wordt gebruik gemaakt van de afstand tot een object en niet van de coördinaten in de puntenwolk zelf. Het is immers veel eenvoudiger om een afstand na te meten, dan het nagaan van coördinaten in een drie dimensionale ruimte relatief ten opzichte van de ZED.

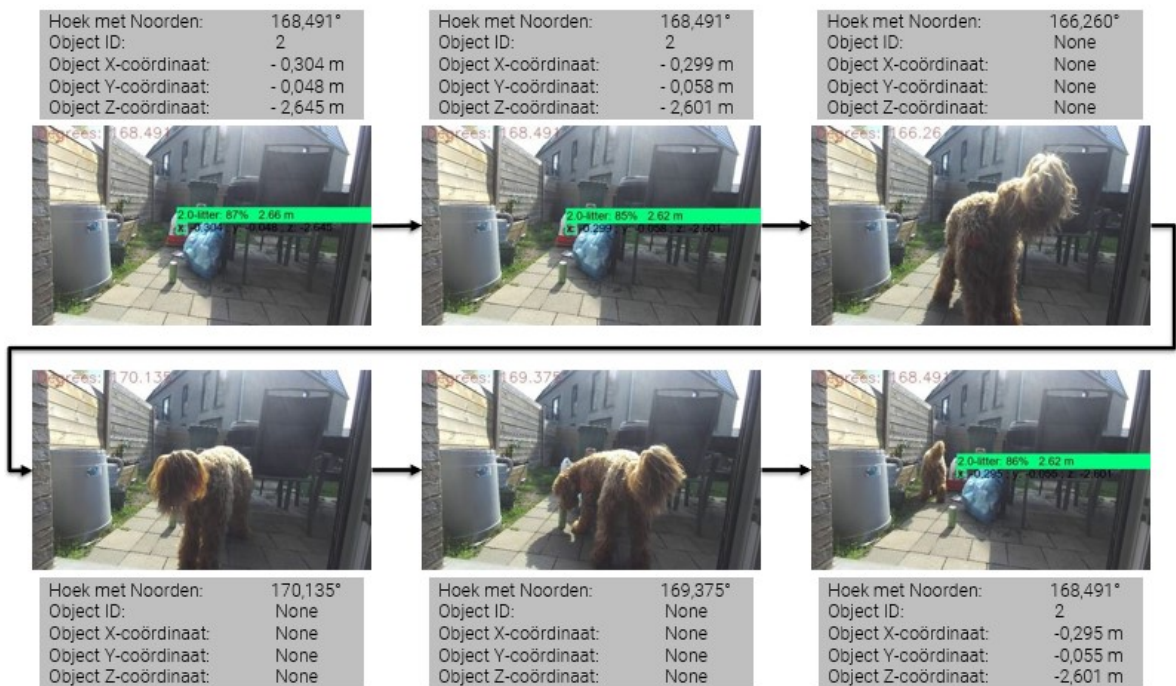
Als test werd de ZED zo horizontaal mogelijk opgesteld op een hoogte van 0.5 m op de volgende lambertcoördinaten:

- $X = 125\,166,40$
- $Y = 201\,895,48$

Deze Lambertcoördinaten werden fictief gewijzigd ten opzichte van de werkelijke coördinaten omwille van privacy. Dit doet echter geen afbreuk aan de methodologie. Vervolgens werd de ZED gericht naar een aantal geplaatste zwerfvuil objecten. Omdat deze proeven parallel verliepen aan de AI proeven is voor de eenvoud en consistentie gebruik gemaakt van het Let's do it foundation model, dat eerder werd besproken in paragraaf 2.3.3.1. Er dient in deze proef dus geen aandacht te worden besteed aan de kwaliteit van het model zelf. De resultaten van deze proefopstelling kunnen worden teruggevonden in

Figuur 70.

Figuur 70: Resultaat testopstelling relatieve locatiebepaling



In

Figuur 70 worden zes sequentiële *frames* weergegeven met een tussentijd van telkens twee seconden. Voor elk van de *frames* wordt eerst de hoek van de camera met het noorden opgemeten, conform paragraaf 4.3.1.1. Vervolgens wordt voor elk gedetecteerd object in de frame de coördinaten geëxtraheerd uit de puntenwolk. Op basis van deze coördinaten kan de afstand worden berekend gebruikmakend van Vergelijking 11. Merk op dat in deze vergelijking slechts één x-, y- en z-coördinaat staat. Dit omdat de camera gelijkgesteld wordt met de oorsprong: (0,0,0).

Vergelijking 11: Afstandsberekening in drie dimensies

$$d = \sqrt{x^2 + y^2 + z^2}$$

In de *frames* waar het object zichtbaar is, zijnde frame 1, 2 en 6, wordt telkens een afstand van ongeveer 2,62 m berekend. Nameten van deze afstand levert 2,37 m op. Het verschil tussen geobserveerde en gemeten waarde bedraagt dus 25 cm. Uit

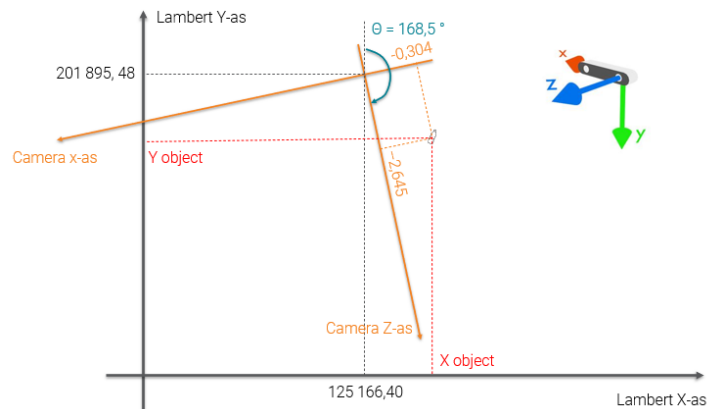
Figuur 70 blijkt tevens dat de coördinaten telkens kleine afwijkingen vertonen ten opzichte van voorgaande *frames*, terwijl zowel het object als de camera zich in stilstand bevonden. Hieruit kan geconcludeerd worden rekening gehouden moet worden met een zekere fout op de metingen.

4.3.1.3 De absolute locatiebepaling van objecten

De principes uit paragraaf 2.4.2.3 kunnen worden toegepast om de uiteindelijke geografische coördinaat van het object te bepalen. Voor de eenvoud wordt verder gewerkt op de resultaten van de proefopstelling in paragraaf 4.3.1.2. Om de redenering terug te duiden, worden de gegevens die verband houden met frame 1 uit

Figuur 70 nog eens grafisch voorgesteld in Figuur 71.

Figuur 71: Absolute locatiebepaling



In Figuur 71 wordt de gemeten GPS-coördinaat van de camera uitgezet ten opzichte van de assen van het Lambertcoördinatensysteem (zwarte stippenlijnen). Deze coördinaat is het nulpunt van het relatieve assenstelsel van de ZED. De juiste ligging van dit relatieve coördinatensysteem wordt bekomen uit de oriëntatie van de ZED ten opzichte van het Noorden (oranje assensysteem). Dit assensysteem vormt het lokale assenstelsel van de ZED. De coördinaten van het object kunnen afgezet worden ten opzichte van dit lokale assenstelsel. De coördinaten zelf kunnen worden gevonden uit de puntenwolk die gevormd is op basis van stereo-fotogrammetrie. De oranje stippenlijnen geven de locatie weer van het object ten opzichte van het relatieve assenstelsel van de ZED. Merk hierbij op dat hoewel de z-waarde negatief werd voorgesteld in Figuur 70, deze toch een positieve zin heeft. De ZED beschouwt een diepte nl. als een negatief getal. Het zou juist zijn om voor z een positief getal te hebben.

De coördinaten van het object ten opzichte van het Lambertcoördinatensysteem kunnen wiskundig berekend worden met de formules in Vergelijking 12.

Vergelijking 12: Rotatie- en translatie formules

$$Y_{object} = Z_{object,relatief} * \cos(\theta) - x_{object,relatief} * \sin(\theta) + Y_{oorsprong,relatief}$$

$$X_{object} = x_{object,relatief} * \cos(\theta) + Z_{object,relatief} * \sin(\theta) + Z_{oorsprong,relatief}$$

De formules in Vergelijking 12 invullen met de waarden uit frame 1 uit Figuur 70: Resultaat testopstelling relatieve locatiebepaling levert:

$$Y_{object} = 2.645 * \cos(168,5^\circ) - (-0.304) * \sin(168,5^\circ) + 201895.48$$

$$X_{object} = (-0.304) * \cos(168,5^\circ) + 2.645 * \sin(168,5^\circ) + 125166.40$$

$$Y_{object} = -2.53116 + 201895.48 = 201892.9$$

$$X_{object} = 0.825623 + 125166.40 = 125167.2$$

Op basis van een GPS-ontvanger, stereo-fotogrammetrische hardware en een magnetometer is het dus mogelijk om aan object lokalisatie uit te voeren. De accuraatheid hiervan is afhankelijk van de accuraatheid van de data

van elk van deze onderdelen. De afwijking per onderdeel werd reeds besproken in de respectievelijke paragrafen 0, 4.3.1.1 en 4.3.1.2. De exacte fout op het eindresultaat kan nagerekend worden door gebruik te maken van de foutentheorie. Omwille van de grote fout op de GPS-ontvanger is het echter duidelijk dat deze dominant zal zijn in de eindfout. De uiteindelijke fout zal telkens groter zijn als de fout van de GPS-ontvanger, die tussen twee en vijf meter ligt. De fouten gemaakt door de ZED en de magnetometer zijn verwaarloosbaar ten opzichte van deze fout.

4.3.2 Object tracering

Object tracering wordt gebruikt om dubbeltelling op sequentiële *frames* te voorkomen (paragraaf 2.4.3). Omdat deze proeven parallel verliepen aan de AI proeven is voor de eenvoud opnieuw en consistentie gebruik gemaakt van het Let's do it foundation model. De output van het Let's do it foundation model is de input van het SimpleSort algoritme en is bijgevolg compleet afhankelijk van een goede werking van het object detectie model.

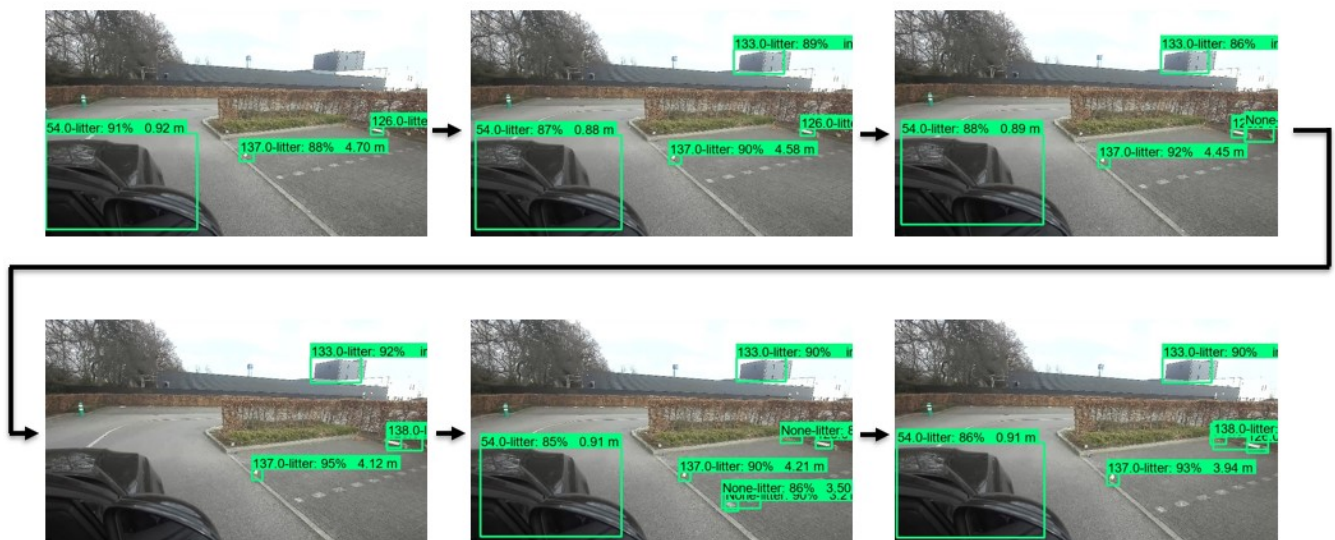
Op

Figuur 72 staat de output van het SimpleSort algoritme voor zes sequentiële *frames*. Elke *bounding box* toont respectievelijk drie cijfers: het tracersingsnummer, de kans dat het gedetecteerde object effectief zwervuil is en de afstand tot het gedetecteerd zwervuil. Alleen het eerste cijfer, het tracersingsnummer, interesseert ons in dit geval. Indien het tracersingsnummer van een zwervuilobject doorheen de *frames* ongewijzigd blijft dan gaat het over hetzelfde zwervuilobject, vb.: zwervuilobject nummer 137 behoudt zijn tracersingsnummer doorheen zes *frames*. Op deze manier wordt dubbeltelling op sequentiële *frames* voorkomen.

Paragraaf 2.4.3 gaf aan dat het SimpleSort algoritme niet om kan met objecten die tijdelijk uit beeld verdwijnen door bijvoorbeeld een obstakel die het zicht op het zwervuilobject verspert.

Figuur 70, toont aan dat dit wel degelijk mogelijk is. Het zwerfvuilobject behoudt hetzelfde traceringsnummer, zelfs nadat de hond passeerde. Hierbij moet vermeld worden dat deze eigenschap beperkt is. Na tien *frames* waar het zwerfvuilobject buiten beeld is, verliest het algoritme de tracering.

Figuur 72: Output SimpleSort algoritme



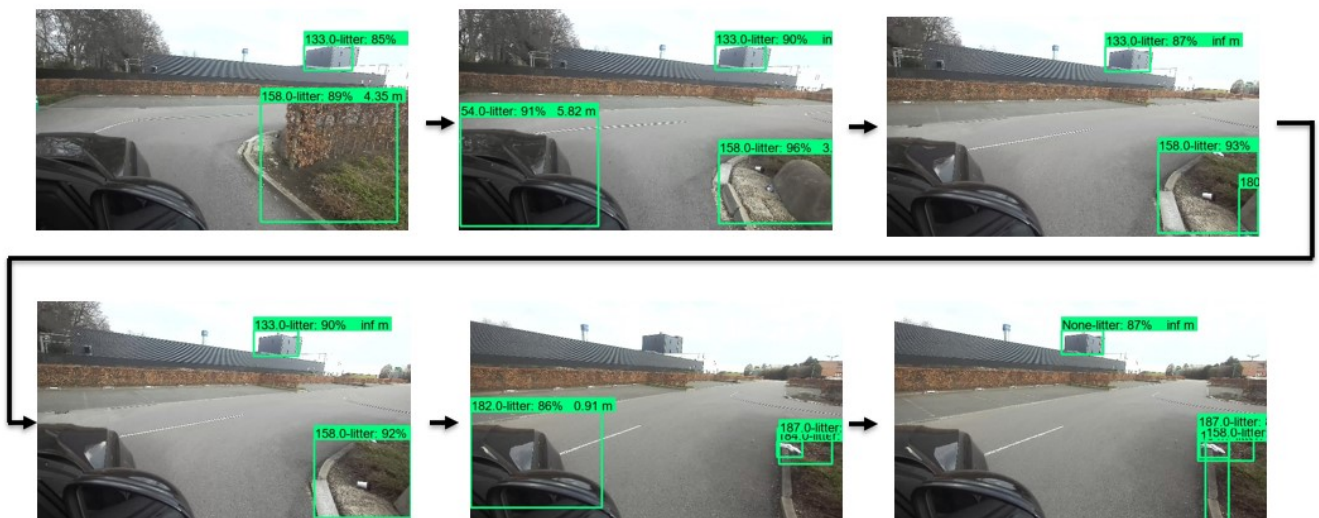
Het SimpleSort algoritme werkt goed, maar is niet robuust genoeg voor een mogelijke uitrol. Een voldoende werkend object tracering algoritme moet antwoord bieden op volgende problemen:

- **Omgaan met kleine objecten:** Het SimpleSort algoritme baseert zich op de overlap van *bounding boxes* (paragraaf 2.4.3), wat een mogelijk probleem kan vormen voor zwerfvuilobjecten die klein in omvang zijn en bijgevolg een kleinere kans hebben tot overlap.
- **Snelheid sequentiële frames:** De snelheid van de wagen en de *frames* per second van de camera hebben een significante invloed op de prestatie van object detectie algoritmen. Hoe lager de fps en hoe hoger

de snelheid van de wagen, hoe minder overlap er zal zijn tussen de voorspelde *bounding box* van het object tracing algoritme en het AI model. Dit tast de resultaten van het object tracing algoritme aan.

- **Onverwachte bewegingen:** In object tracing voorspelt de Kalmanfilter de locatie van de volgende *bounding box* aan door middel van de vorige locaties van de *bounding box* (paragraaf 2.4.3). Bijgevolg introduceren onverwachte bewegingen ruis in deze voorspellingen. Dit probleem wordt aangetoond in
- Figuur 73. De Kalmanfilter verwacht een rechte lijnige beweging. Bochten zorgen ervoor dat de voorspelde *bounding box* van de Kalmanfilter van het gebouw niet voldoende overlapt met de door het AI model voorspelde *bounding box*, waardoor het algoritme de tracing verliest. Dit probleem kan worden opgelost door een IMU-apparaat te koppelen. Met behulp van deze data kan de Kalmanfilter beter voorspellen wat de locatie zal zijn van de volgende *bounding box*.

Figuur 73: De geïntroduceerde ruis door onverwachte bewegingen



Samenvattend kan gesteld worden dat dubbeltelling op sequentiële *frames* voorkomen kan worden door object tracing algoritmen. In een uitrol zullen meer geavanceerde object tracing algoritmen zoals DeepSort in combinatie met IMU-data normaliter een robuuste oplossing bieden voor dit probleem (paragraaf 2.4.3).

5 FINANCIËLE STUDIE

Na een diepgaande technische studie en de uitwerking van de POC kan er een zo goed mogelijke inschatting gemaakt worden van de kostprijs van de beschreven oplossing. In deze studie wordt nagegaan of het voor de OVAM economisch verantwoord zou zijn een AI-systeem voor automatische zwerfvuildetectie te laten bouwen.

De financiële studie wordt opgedeeld in twee delen: productie set-up (alles wat nodig is om de oplossing te bouwen) en productie (alles in verband met het onderhouden en het gebruik van de oplossing). De berekeningen werden gemaakt in Excel en worden apart overgedragen aan de OVAM. In het rapport worden alle cijfers afgerond tot twee cijfers na de komma. Dit kan leiden tot beperkte discrepanties tussen het rapport en het overgedragen Excel bestand. Bijkomend werd gekozen om niet alle berekeningen toe te lichten. Hetzelfde geldt voor de maximale kostenberekeningen, hiervoor wordt verwezen naar het Excel bestand. Bijkomend worden alle kosten berekend exclusief BTW.

Alvorens dieper in te gaan op de verschillende kostcomponenten, moet er toegelicht worden waarop beslissingen over bepaalde componenten zich baseren. Vooreerst wordt er dieper ingegaan op de werkwijze van de **datacaptatie** in een 10 bij 10 meter cel. Deze manier bepaalt namelijk de hoeveelheid data en bij gevolg de kost die erbij komt kijken. Uit de POC komen drie manieren van beelden capteren naar boven: video's, *bursts* en ZED-video's. Gezien de POC geen uitsluitsel geeft over de ideale captatievorm, wordt uit eenvoudsoverwegingen uitgegaan van video's. De impact van deze onzekerheid op de financiële studie is verwaarloosbaar aangezien de opslagkost een minimale impact heeft op de totale kost en de grootte van de bestanden die elke manier oplevert, nauwelijks verschillen.

Ten tweede moeten deze gemaakte video's worden **omgezet naar frames**. Camera's nemen typisch 30 *frames* per seconde. Er dient dus een keuze gemaakt te worden hoeveel van de originele 30 *frames* er worden gebruikt. De fps te hoog zetten zou leiden tot te hoge kosten (langere berekeningstijd, hogere opslagkost, etc.). De fps te laag zetten zou echter een negatieve impact kunnen hebben op de object tracering (paragraaf 2.4.3). Bijgevolg wordt gekozen voor een veelgebruikte fps van tien. Dit geldt zowel in set-up productie (paragraaf 5.1) als in productie (paragraaf 5.2). Om de maximale kost te berekenen wordt uitgegaan van 30 fps in set-up productie. Dit omdat het systeem uitvoerig getest moet worden. In productie wordt de maximale fps gelijkgesteld aan de normale fps.

Ten derde worden de meeste kosten uitgedrukt in mandagen. Er wordt een kost van €800/dag (exclusief BTW) gehanteerd voor taken die domeinkennis vereisen zoals bijvoorbeeld model training. Dit is de kost voor een senior consultant binnen Ordina. Voor de taken zonder specifieke domeinkennis wordt het gemiddeld brutoloon van een werknemer in België genomen. Dit bedraagt €2.916 per maand⁴⁷. Hierop betaalt de werkgever 24,82% patronale bijdragen⁴⁸. In de veronderstelling dat er 20 werkdagen in een maand zitten, komt dit neer op €182 per dag.

⁴⁷ <https://www.sdworx.be/nl-be/pers/2021/2021-01-28-brutoloon-steeg-in-2020>

⁴⁸ <https://www.socialsecurity.be/employer/instructions/dmfa/nl/latest/instructions/socialsecuritycontributions/contributions.html>

Tot slot dient vermeld te worden dat er geen eenduidig, end-to-end systeem gedefinieerd werd. De OVAM dient zelf keuzes te maken op bepaalde vlakken waarbij ze de kosten ten opzichte van de baten in acht neemt. Bijgevolg zullen de kosten afhangen van deze keuzes. In de totaalkostberekening wordt uitgegaan van het systeem dat de voorkeur draagt volgens Ordina.

5.1 SET-UP PRODUCTIE

Zoals uitgebreid beschreven in de technische studie zijn er verschillende bouwstenen en processen nodig om de oplossing te bouwen. De kosten worden opgedeeld in vijf categorieën: infrastructuur, processen, annotatie, hardware en modelontwikkeling.

5.1.1 Infrastructuur

Voor de opbouw van het model zullen de meest geschikte infrastructuur componenten moeten worden opgezet (set-up). De opslag componenten zullen ook een opslagkost met zich meedragen. Een overzicht van de kosten kan in onderstaande tabel teruggevonden worden.

Tabel 15: Kostenberekening set-up productie infrastructuur (exclusief btw)

Kosten infrastructuur		
Onderdeel	Minimale totale kost	Maximale totale kost
Set-up	€20.800,00	€33.600,00
Opslag	€53,90	€145,19
Totaal	€20.853,90	€33.745,19

5.1.1.1 Set-up kost

De belangrijkste bouwstenen werden reeds in paragraaf 2.5 besproken. Drie componenten die nog niet werden besproken zijn:

- Abonnement en resource group in Azure.
- Azure DevOps: End-to-end DevOps toolchain voor het ontwikkelen en *deployen* van software, zorgt onder andere voor betere samenwerking en beschikbaarheid van de code.
- Azure Key Vault: Databescherming en compliance.

De kost wordt berekend aan de hand van het aantal mandagen nodig voor het opzetten van de infrastructuur componenten. De inschatting is gebaseerd op eigen ervaring van Ordina.

Tabel 16: Kostenberekening set-up - productie infrastructuur – detailberekening (exclusief BTW)

Kosten infrastructuur set-up					
Infrastructuur component	Gemiddeld Aantal mandagen	Maximum aantal mandagen	Kost per eenheid mandag	Gemiddelde totale kost	Maximale totale kost
Azure Cloud subscriptions & resource groups	2,00	2,00	€800,00	€1600,00	€1600,00
Azure Data Lake Storage (incl. retentie)	2,00	2,00	€800,00	€1600,00	€1600,00
Azure Machine Learning	2,00	2,00	€800,00	€1600,00	€1600,00
Azure Databricks	2,00	2,00	€800,00	€1600,00	€1600,00
Azure Key Vault	2,00	2,00	€800,00	€1600,00	€1600,00
Azure DevOps	5,00	10,00	€800,00	€4000,00	€8000,00
Azure SQL Database	2,00	2,00	€800,00	€1600,00	€1600,00
Authenticatie & beveiliging	2,00	5,00	€800,00	€1600,00	€4000,00
Networking	2,00	5,00	€800,00	€1600,00	€4000,00
Versionering data (database opzetten, rapportage)	2,00	5,00	€800,00	€1600,00	€4000,00
Monitoring dashboard met KPI's	3,00	5,00	€800,00	€2400,00	€4000,00
Totaal	26,00	42,00		€20.800,00	€33.600,00

5.1.1.2 Opslagkost

Zoals aangeven in 2.5.3 zullen de video's, foto's en annotaties worden opgeslagen in de ADLS. Wanneer er data wordt opgeslagen in de ADLS zullen er opslagkosten worden aangerekend. De kost is afhankelijk van de hoeveelheid data en wordt berekend per TB per maand.

We onderscheiden drie typen bestanden die worden opgeslagen in set-up productie. Het eerste type is video's. Deze worden gebruikt om het end-to-end systeem (AI model + zwerfvueltelling) te valideren. Het in kaart brengen van 20 (maximaal: 50) 10 bij 10 meter cellen zou voldoende moeten zijn om het systeem uitvoerig te testen. Als het 1,5 min duurt om één cel te filmen dan levert dit 30 min (maximaal: 75) videomateriaal op, wat overeenkomt met **0,02 TB** (maximaal: 0,05) aan videomateriaal.

De opslagkosten zullen grotendeels bestaan uit het 2^{de} type: afbeeldingen. Het totaal aantal afbeeldingen dat wordt opgeslagen komt voort van twee processen: het end-to-end validatie- en het trainingsproces.

1. **End-to-end validatieproces:** 30 min videomateriaal levert 18.000 foto's op (**0,03 TB**). Bijgevolg neemt de video (30 fps) minder geheugen in als de foto's geëxtraheerd aan tien fps. Dit kan verklaard worden door het feit dat video's maar enkele *frames* aan hun volledige kwaliteit coderen. De resterende *frames* worden gecodeerd als het verschil tussen de frame en de vorige frame. Deze grootte is logischerwijs minder dan de volledige foto. Bijkomend wordt er bij elke foto metadata gegenereerd.
2. **Trainingsproces:** Het trainingsproces vergt 46.312 afbeeldingen (**0,07 TB**). Deze berekening is gebaseerd op de ondervindingen in de POC en wordt in paragraaf 4.2.4 toegelicht.

Bijgevolg wordt er in totaal **0,10 TB** aan afbeeldingen opgeslagen.

Ten derde worden de annotaties opgeslagen die nodig zijn om het model te trainen: annotaties voor 46.312 trainingsfoto's nemen **0,00011 TB** geheugen in beslag. Tabel 17 geeft een overzicht van de kosten die per maand gemaakt worden in verband met de opslag. In Azure kan data langere tijd worden opgeslagen op drie lagen: de hot, archive en cool laag (paragraaf 2.5.3.1). De snelste, maar ook duurste variant van opslaan is opslaan op de hot laag. Snelheid is immers een belangrijke factor wanneer modellen ontwikkeld worden aangezien de data meermaals gebruikt wordt. Echter, wanneer de trainingsfase voorbij is, kan de data op een goedkopere, tragere opslag laag worden opgeslagen: de archive laag. Bijgevolg stelt Ordina voor om de data alleen op de hot laag te zetten als deze effectief gebruikt wordt in de trainingsfase. In paragraaf 4.2.4 werd de totale trainingstijd berekend. Deze bedroeg 409 dagen met een YOLO v5 model. Echter als er getraind wordt met vier TeslaK80 GPU's wordt de trainingstijd gereduceerd tot 102,5 dagen. Dit betekent dat het trainingsproces vier maanden duurt. Bijgevolg wordt de data (foto's) eerst vier maanden op de hot laag en vervolgens 14 maanden op de archive laag opgeslagen. De video's worden meteen omgezet en moeten één maand op de hot laag. Dit resulteert in een kost van €53,90 en maximaal €145,19 voor een periode van 18 maanden.

Tabel 17: Kostenberekening set-up - productie infrastructuur opslag (exclusief BTW)

Kosten infrastructuur opslag					
Soort opslag	Aantal TB per maand	Maximum aantal TB per maand	Kost per eenheid TB per maand	Totale kost per maand	Maximum totale kost per maand
Video's Hot laag	0,02	0,05	€85,00	€1,70	€4,25
Foto's en annotaties Hot laag	0,10	0,27	€85,00	€8,50	€22,95
Foto's en annotaties Archive laag	0,10	0,27	€13,00	€1,30	€3,51
Totaal 18 maanden				€53,90⁴⁹	€145,19

⁴⁹ 1,70 + (4 x 8,50) + (14 x 1,30)

5.1.2 Hardware

Om dit project succesvol uit te voeren zal er extra hardware aangekocht moeten worden. Meerdere versies van de verschillende hardwarecomponenten bestaan waardoor de prijs varieert naargelang de vereiste specificaties. De maximale kost heeft betrekking op de hardware versie met de meeste specificaties.

Tabel 18: Kostenberekening set-up – hardware (exclusief BTW)

Kosten hardware		
Hardware	Totale kost	Maximale totale kost
Ardusimple GPS ⁵⁰	€211,00	€662,00
IMU ⁵¹	€100,00	€450,00
Goede camera	€1.000,00	€5.000,00
Totaal	€1.311,00	€6.112,00

5.1.3 Processen

Bij processen dient betaald te worden voor het creëren van een proces en het uitvoeren. De kost voor het opzetten van de diverse processen wordt berekend in Paragraaf 5.1.3.1 aan de hand van het aantal mandagen. Wanneer de processen zijn opgezet, kunnen ze uitgevoerd worden. Zoals vermeldt in paragraaf 5.1.1.2, zal een 30 minuten (maximaal: 75) validatievideo van het end-to-end systeem worden verwerkt.

Tabel 19: Kostenberekening set-up – productieprocessen (exclusief BTW)

Kosten processen		
Onderdeel	Totale kost	Maximale totale kost
Creatie	€79.200,00	€155.200,00
Berekening	11,45	€94,62
Totaal	€79.211,45	€155.294,62

5.1.3.1 Creatie van een proces

Om een robuuste end-to-end oplossing te bekomen moet de data door drie processen:

1. **Opzet pre-processing proces:** Het *Pre-processing* proces bestaat uit twee sub-processen:
 - a. De bouw van een *blurring* model, juridisch gezien noodzakelijk (paragraaf 3).
 - b. Transformatie van video naar afbeeldingen werd al gebouwd in de POC, maar moet geoptimaliseerd en geïntegreerd worden in *pipelines*.
2. **Opzet inference proces:** Het inference proces dient om de gecapteerde data automatisch door het object detectie model te sturen.

⁵⁰ <https://www.ardusimple.com/store/>

⁵¹ <https://www.digikey.be/products/en/sensors-transducers/motion-sensors-imus-inertial-measurement-units/567>

3. **Opzet zwerfvuittelling proces:** Het bouwen van dit proces bestaat uit twee componenten: object lokalisatie en object tracering (paragraaf 2.4).
- Object lokalisatie gebeurt idealiter met een stereocamera. Gezien de slechte camerakwaliteit van de ZED (paragraaf 4.1.2), dienen andere diepte camera's onderzocht te worden (tenzij de ZED betere beeldkwaliteit aanlevert) of in het slechtste geval zal een eigen stereo-fotogrammetrisch systeem moeten gebouwd worden. Dit zal gelden als de maximum kost.
 - Gebruik van stereocamera: Dit is de goedkoopste oplossing aangezien de software al geïntegreerd is in de stereocamera. Aangezien nog niet geweten is welke stereocamera's (of andere methodologieën⁵²) voldoen, moet ook extra onderzoek hiernaar gebeuren.
 - Zelf bouwen stereo-fotogrammetrisch systeem: Dit is de duurste oplossing aangezien de oplossing zelf gebouwd moet worden via stereo-fotogrammetrische principes (paragraaf 2.4.2.1.2)
 - Voor object tracering werd in de POC gebruikt gemaakt van het SimpleSort algoritme (paragraaf 4.3.2). Echter in een uitrol is er nood aan een meer robuuste oplossing zoals het DeepSort algoritme. Bijkomend moet een IMU geïntegreerd worden (paragraaf 2.4.3).

De kostenberekening wordt weergegeven in Tabel 20.

Tabel 20: Kostenberekening set-up – productieprocessen creatie (exclusief BTW)

Kosten processen creatie					
Soort proces	Aantal mandagen	Maximum aantal mandagen	Kost per eenheid mandag	Totale kost	Maximum totale kost
<i>Pre-processing:</i> Blurring persoonsgegevens	5,00	8,00	€800,00	€4.000,00	€6.400,00
<i>Pre-processing:</i> Video transformatie naar foto's	4,00	8,00	€800,00	€3.200,00	€6.400,00
Training (inference)	5,00	8,00	€800,00	€4.000,00	€6.400,00
<i>Post-processing:</i> a) Object lokalisatie	50,00	100,00	€800,00	€40.000,00	€80.000,00
b) Object tracering					
- robuust algoritme	30,00	60,00	€800,00	€24.000,00	€48.000,00
- integratie IMU	5,00	10,00	€800,00	€4.000,00	€8.000,00
Totaal	99,00	194,00		€79.200,00	€155.200,00

⁵² <https://www.intelrealsense.com/lidar/>

5.1.3.2 Berekeningskost

Zoals vermeld in de opslagkost, wordt er uitgegaan van 30 (maximaal: 75) minuten filmtijd ter validatie van het end-to-end systeem. Deze video moet sequentieel door de processen *Pre-processing*, inference en zwerfvuiltelling.

1. *Pre-processing* bestaat uit twee sub-processen: het *blurren* van persoonsgegevens en het transformeren van video naar foto's. Aangezien er momenteel nog geen *blurring* model aanwezig is, veronderstellen we dat in totaal de *Pre-processings*fase dubbel zo lang zal duren. Bijgevolg schat Ordina dat de *Pre-processings*fase 4,14 min duurt per één min video. In totaal wordt het *Pre-processing*proces dus 124,80 min (maximaal: 312) uitgevoerd. Bemerk dat hierboven staat vermeld dat het sub-proces 'transformatie van video naar frames' nog moet geoptimaliseerd worden. De berekende 4,14 min per één min video zal aanzienlijk dalen.
2. Het YOLO v5 model en het Mask R-CNN model kunnen respectievelijk 15 en 4 frames per seconde behandelen (paragraaf 4.2.3) op één TeslaK80 GPU. Op vier TeslaK80 GPU's worden bijgevolg 60 en 16 frames per second behandeld. 30 min filmtijd (maximum: 75 min filmtijd) levert, met een fps van tien, 18.000 (maximaal: 45.000) frames op. Dit brengt de totale inferencetijd op respectievelijk 5 en 46,875 min. Bijkomend moeten deze cijfers vermenigvuldigd worden met het totaal aantal gemaakte modellen. De assumptie wordt gemaakt dat 24 modellen worden gebouwd. De totale inferencetijd is bijgevolg 120,00 en 1.125,00 minuten.
3. Het is empirisch vastgesteld door Ordina dat het zwerfvuiltelling proces vijf afbeeldingen kan processen op één seconde. Het zwerfvuiltelling proces duurt bijgevolg op 18.000 (maximum: 45.000) frames, 60 (maximaal: 150) min.

De *Pre-processing* en zwerfvuiltelling processen worden uitgevoerd op een CPU-cluster. Het inferenceproces wordt uitgevoerd op vier TeslaK80 GPU's. De totale kosten staan in Tabel 21.

Tabel 21: Kostenberekening set-up – productie processen berekeningskost (exclusief BTW)

Kosten processen berekening					
Soort proces	Uitvoeringstijd proces (min)	Maximum uitvoeringstijd proces (min)	Kost per minuut processing	Totale kost	Maximum totale kost
<i>Preprocessing</i>	124,80	312,00	€0,01	€1,25	€3,12
<i>Inference</i>	120,00	1.125,00	€0,08	€9,60	€90,00
Zwerfvuiltelling	60,00	150,00	€0,01	€0,60	€1,50
Totaal	304,80	1587,00		€11,45	€94,62

5.1.4 Onderzoek en opzet

Extra onderzoek is nodig voor verschillende componenten in een end-to-end systeem:

1. **Onderzoeken Galileo GPS:** Zoals besproken in paragrafen 2.4.2 en 4.3 heeft een robuust end-to-end systeem een goede GPS nodig. Gezien de tekortkomingen van de huidige systemen voor locatiebepaling, is de preciezere Galileo GPS nodig om aan object lokalisatie te doen (paragraaf 2.4.2). Deze dient getest te worden waarvoor een kost wordt voorzien.

2. Hardware koppeling:

- a. Het systeem voor locatiebepaling dient geïntegreerd te worden met bv. een Raspberry Pi of Arduino board en vervolgens gesynchroniseerd te worden met de camera (interne klokken). Een opzettijd van zes weken (hardware-expert) tot acht weken (niet-hardware-expert) wordt gedefinieerd (paragraaf 2.2.2.2).
 - b. Bijkomend dient een IMU geïntegreerd te worden met een object traceringsalgoritme. Hiervoor wordt tien mandagen gerekend.
3. **Testen camera's:** Aangezien noch de GoPro, noch de ZED goede beeldkwaliteit aanleveren (paragraaf 4.1), moeten extra dagen worden ingerekend om potentiële camera's te testen. In paragraaf 4.1.5 staat de methodologie voor een mogelijk vervolgtraject.
 4. **Data upload:** Data gecapteerd in het beeldopname proces moet worden opgeslagen in de Cloud. In paragraaf 2.6.2 staat vermeld dat er twee mogelijke opties zijn: een manuele upload en een automatische upload via Azure Data Factory. Alleen Azure Data Factory wordt meegenomen in de totale kost berekening aangezien dit de geadviseerde keuze is.
 5. **Privé/privaat domein:** Indien object lokalisatie gebruikt wordt, kan kadastrale informatie gebruikt worden om privé en privaat domein te onderscheiden (paragraaf 2.1.1).
 6. **Transport modi onderzoek:** extra onderzoek moet worden gedaan naar de transport modi:
 - De impact van zichtversperrende objecten.
 - Hoe een flexibel systeem (bodycam, auto & fiets) operationeel kan worden opgezet.
 - Etc.

Tabel 22: Kostenberekening set-up – productie onderzoek & opzet (exclusief BTW)

Kosten processen creatie					
Soort proces	Aantal mandagen	Maximum aantal mandagen	Kost per eenheid mandag	Totale kost	Maximum totale kost
Onderzoeken Galileo	15,00	30,00	€800,00	€12.000,00	€24.000,00
Hardware koppeling	52,00	66,00	€800,00	€41.600,00	€52.800,00
Testen camera's	10,00	20,00	€800,00	€8.000,00	€16.000,00
Batch upload (Azure Data Factory)	10,00	20,00	€800,00	€8.000,00	€16.000,00
Privé/privaat domein	10,00	20,00	€800,00	€8.000,00	€16.000,00
Onderzoek transport modi	30,00	50,00	€800,00	€24.000,00	€40.000,00
Totaal	127,00	206,00		€101.600,00	€164.800,00

5.1.5 Annotatie

Zoals vermeld in paragraaf 4.2.2 kan de OVAM de annotatie zelf uitvoeren of outsourcen naar een externe partij. Indien gekozen wordt om zelf te annoteren, moet er annotatiesoftware opgezet worden. De gebruikte annotatiesoftware is Azure ML labeling service (paragraaf 2.5.3.3). Dit opzetten vergt een halve dag werk. Indien

de tweede optie verkozen wordt, moet een *API-endpoint* worden gebouwd. Een *API* zorgt ervoor dat twee systemen met elkaar kunnen communiceren: de OVAM en het annotatiebedrijf. Via een *API-endpoint* kunnen de afbeeldingen online uitgewisseld worden. Het bouwen van een dergelijk systeem duurt vijf dagen.

Gezien het grote financiële voordeel dat outsourcing biedt (paragraaf 5.1.6.2), wordt alleen de opzetkost van een *API-endpoint* in rekening gebracht in het eindtotaal. Bemerkt dat in onderstaande tabel enkel de kosten voor de opzet van de annotatie-infrastructuur in rekening wordt gebracht. De kosten die gemaakt worden voor het uitvoeren van de annotatie worden berekend in paragraaf 5.1.6.2.

Tabel 23: Kostenberekening set-up – productie opzet annotatie-infrastructuur (exclusief BTW)

Kosten annotatie					
Annotatie manier	Minimum aantal mandagen	Maximum aantal mandagen	Kost per eenheid	Minimum totale kost	Maximum totale kost
Zelf annoteren	0,50	1,00	€800	€400,00	€800,00
Outsourcing (gekozen optie)	5,00	8,00	€800	€4.000,00	€6.400,00

5.1.6 Modelontwikkeling

Tot slot worden de kosten berekend voor het bouwen van de nodige modellen. Dit bestaat uit vijf stappen: beelden opnemen, beelden annoteren, model trainen, model testen en beschikbaar stellen of *deployment*.

Tabel 24: Kostenberekening set-up – productie model development (exclusief BTW)

Kosten model development		
Onderdeel	Minimale totale kost	Maximale totale kost
Beelden opnemen	€8.226,40	€10.082,80
Beelden annoteren	€1.806,17	€1.806,17
Model trainen	€109.807,56	€166.538,65
Model testen	€19.200,00	€38.400,00
<i>Deployment</i>	€2.400,00	€4.000,00
Totaal	€141.440,13	€220.827,62

De kosten uit deze tabel worden meer in detail bekeken in Tabel 25 t.e.m. 28.

5.1.6.1 Beelden opnemen

Er moeten beelden opgenomen worden voor twee specifieke processen:

1. **End-to-end validatie:** Dit werd reeds meermaals besproken. Deze video dient om het uiteindelijke systeem uitvoerig te testen.
2. **Trainingsproces:** Beelden worden gecapteerd en dienen als input voor het trainingsproces. Voor het trainingsproces wordt er geen gebruik gemaakt van video's, maar van manueel getrokken foto's. Dit wordt verduidelijkt met een voorbeeld. Stel er wordt zes uur aan videomateriaal opgenomen om het model te trainen. Met een fps van tien resulteert dit in 216.000 afbeeldingen. Deze dienen manueel gefilterd te worden om na te gaan of de foto zwerfvuil bevat en uniek is⁵³. Het spreekt voor zich dat dit proces lang zal duren, wat leidt tot de keuze om foto's te nemen. Voor deze opzet zijn twee mankrachten nodig: iemand die het voertuig bedient en iemand die foto's neemt.

Vooreerst wordt berekend hoelang het duurt om de end-to-end validatievideo op te nemen. In het deeltje opslagkost (paragraaf 5.1.1), werd berekend hoe groot dit videobestand zou zijn. Er werd geschat dat twintig 10 bij 10 meter cellen voldoende zijn om het systeem te valideren. Als we kunnen berekenen hoelang het duurt om een validatievideo te maken in één 10 bij 10 meter cel, dan volstaat het om dit getal te vermenigvuldigen met 20 cellen om de totale inspanning te meten voor het maken van een end-to-end validatievideo. Volgende componenten bepalen de tijd:

- Transporttijd: De tijd die het vergt om van cel A naar cel B te gaan.
- Rondstrooitijd: De tijd die het vergt om zwerfvuil uit te strooien en op te ruimen in een cel.
- Opnametijd: De tijd die het vergt om de cel te filmen.
- Opruimtijd: De tijd die het vergt om het zwerfvuil na de opname op te ruimen.

De opnametijd wordt op 1,5 min gezet, de transporttijd op tien min. Dit omdat een ideale validatievideo verschillende situaties (ondergronden, wegen, etc.) beschouwt. De rondstrooitijd en opruimtijd werd empirisch vastgesteld in de POC en bedraagt voor beiden vijf min. De berekening van de duur (in mandagen) voor de beeldopname van de validatievideo gebeurt als volgt:

Vergelijking 13: Berekening opnametijd validatievideo

$$(10 + 1,5 + 5 + 5) * 20 \text{ cellen} / 60 \text{ seconden} / 8 \text{ uur}$$

In totaal zijn dit 0,9 mandagen voor 20 cellen en 2,24 mandagen voor 50 cellen (maximum). Voor deze actie is slechts één persoon nodig die het voertuig bestuurt.

⁵³ De uniciteit van een foto is belangrijk om *overfitting* te voorkomen

Rest ons enkel nog de vereiste inspanning te berekenen voor het capteren van de beelden voor het trainingsproces. Hiervoor moet het product genomen worden van twee componenten: hoelang het duurt om één foto te nemen en de hoeveelheid beeldmateriaal vereist voor de ontwikkeling van de modellen:

1. In de POC werden 1.568 foto's verzameld op zes uur tijd (paragraaf 4.2.1). Dit betekent dat één foto op 0,0038 uur kan worden genomen. Er wordt met een buffer van 20% gewerkt om het maximum bedrag te bepalen: 0,0046 uur per foto.
2. Paragraaf 4.2.4 meldt dat 46.312 foto's vereist zijn.

Bijgevolg zijn 44,30 (maximaal: 53,16) mandagen vereist voor de opname van beeldmateriaal voor de ontwikkeling van het model en dit voor twee personen. De kost per mandag wordt gelijkgesteld aan €182 gezien geen specifieke domeinkennis vereist is voor dit proces.

Tabel 25: Kostenberekening set-up – productie model development beeldopnames (exclusief BTW)

Kosten model development beelden opnemen					
Proces	Aantal mandagen	Maximum aantal mandagen	Kost per eenheid	Minimum totale kost	Maximum totale kost
End-to-end validatie	0,90	2,24	€182,00	€163,80	€407,68
Training	44,30	53,16	€182,00	€8.062,60	€9.675,12
Totaal	45,20	55,40		€8.226,40	€10.082,80

5.1.6.2 Beelden annoteren

Zoals berekend in paragraaf 4.2.4 zijn er 46.312 beelden nodig om de zwerfvuildetectie modellen te trainen. Deze beelden moeten geannoteerd worden alvorens ze kunnen gebruikt worden als input voor het AI model (paragraaf 4.2.4).

Wanneer de OVAM beslist het annotatieproces zelf uit te voeren, zal er eerst een lijst met labels gegenereerd moeten worden (30 min) waarna elke foto zal geannoteerd worden wat gemiddeld één min duurt (maximaal: twee min). Deze cijfers zijn gebaseerd op ondervindingen uit de POC. In totaliteit zal dus 96,55 (maximaal: 193,03) mandagen geannoteerd moeten worden.

Een tweede mogelijkheid is om de annotatie uit te besteden. De kost om 1000 zwerfvuilobjecten te laten annoteren, bedraagt €52,26⁵⁴. Bemerk dat de externe partij aanrekent per zwerfvuilobject en niet per foto. Niet alle foto's in de trainingsset moeten geannoteerd worden aangezien 40% van de dataset foto's geen zwerfvuil bevat (paragraaf 4.2.3). Met onderstaande formule wordt berekend dat 36.123 zwerfvuilobjecten geannoteerd dienen te worden. Er wordt geschat dat er gemiddeld 1,30 zwerfvuilobjecten per foto aanwezig is. De 40% wordt niet in rekening gebracht bij het zelf annoteren aangezien er ook tijd wordt verloren bij het zoeken naar zwerfvuil op een foto dat er niet is.

⁵⁴ <https://Cloud.google.com/ai-platform/data-labeling/pricing>

Vergelijking 14: Berekening totaal aantal te annoteren zwerfvuilobjecten

$$\# \text{ zwerfvuilobjecten te annoteren} = \text{aantal foto's} \\ \times \text{percentage foto's met zwerfvuil} \\ \times \text{gemiddeld aantal zwerfvuilobjecten per zwerfvuifoto}$$

Een overzicht van de verschillende kosten worden weergegeven in Tabel 26. Alleen het extern annoteren wordt opgenomen in de totale kost gezien het duidelijke financiële voordeel.

Tabel 26: Kostenberekening set-up – productie model development beelden annotatie (exclusief BTW)

Kosten model development beelden annoteren						
Annotatie methode	Aantal	Maximum aantal	Eenheid	Kost per eenheid	Minimum totale kost	Maximum totale kost
Zelf annoteren	96,55	193,03	Mandagen	€182,00	€17.572,10	€35.131,46
Extern (gekozen optie)	36.123,36	36.123,36	Zwerfvuil-objecten	€0,05	€1.806,17	€1.806,17

5.1.6.3 Model trainen

Wanneer de beelden zijn opgenomen en geannoteerd, kan de training beginnen. De model trainingskost heeft twee componenten: de berekeningskost en de model *tunings* kost. De berekeningskost is de kost die wordt aangerekend door Azure ML voor het gebruik van vier TeslaK80 GPU's. Per uur wordt €4,67 aangerekend. Er wordt in totaal 2458,15 uur (maximaal: 3728,19 uur) getraind (paragraaf 4.2.4), wat een berekeningskost van €11.479,54 (maximaal: 17.410,63) oplevert. Er wordt tevens geschat dat een Data Scientist ongeveer 40% van de trainingstijd actief zal tussenkomen. Het maximumbedrag werd berekend uitgaande van de trainingstijd van het Mask R-CNN model.

Tabel 27: Kostenberekening set-up productie model development training (exclusief BTW)

Kosten model development model trainen						
Actie	Aantal	Maximum aantal	Eenheid	Kost per eenheid	Totale kost	Maximum totale kost
Berekeningskosten	2458,15	3728,19	Uren berekeningstijd	€4,67	€11.479,56	€17.410,65
<i>Modeltuning</i>	122,91	186,41	Mandagen	€800,00	€98.328,00	€149.128,00
Totaal					€109.807,56	€166.538,65

5.1.6.4 Model testing en deployment

Nadat het model getraind is, moet het getest en beschikbaar gesteld worden. Er wordt gerekend met een minimum van één mandag en een maximum van twee mandagen testen per fractie. De schatting in mandagen voor de *deployment* is gebaseerd op Ordina's ervaring.

Tabel 28: Kostenberekening set-up productie – model development testing en *deployment* (exclusief BTW)

Kosten model development model testing en <i>deployment</i>					
Actie	Minimum aantal mandagen	Maximum aantal mandagen	Kost per eenheid	Minimum totale kost	Maximum totale kost
Testing	24,00	48,00	€800,00	€19.200,00	€38.400,00
<i>Deployment</i>	3,00	5,00	€800,00	€2.400,00	€4.000,00
Totaal	27,00	53,00		€21.600,00	€42.400,00

5.2 PRODUCTIE

Wanneer de set-up voor productie is gefinaliseerd, kan de uitrol starten. Kosten voor het gebruik en onderhoud worden hier in kaart gebracht. Om een zo goed mogelijke vergelijking te kunnen maken tussen het huidige manuele systeem en het toekomstig AI-systeem, wordt er uitgegaan van 6.538, 10 bij 10 meter cellen die gemonitord moeten worden. Dit is het exact aantal cellen dat bij de vorige manuele meetcampagne in kaart werd gebracht en laat toe beide systemen te vergelijken.

Een belangrijke assumptie die voor de uitrol moet worden gemaakt is de opnametijd. Dit is de tijd die het vergt om een 10 bij 10 meter cel te filmen en hangt af van de modi van captatie. De voor de OVAM meest efficiënte manier van opnemen is een camera op een wagen plaatsen. Zoals eerder aangegeven schat Ordina dat het 1,5 minuut duurt om deze goed in kaart te brengen. Toch zal deze manier van opnemen niet voor alle 10 bij 10 cellen voldoen. Denk aan drukke straten met veel verkeer en zicht versperrende objecten. Deze straten kunnen worden opgevangen door een bodycam (paragraaf 2.1.2). Door de cellen wandelen met een bodycam zal beduidend trager zijn dan met een auto en wordt geschat op tien minuten opnametijd. Tien minuten zal gelden als de maximale opnametijd.

5.2.1 Infrastructuur

De infrastructuur betreft de opslagkost van het beeldmateriaal vergaard in productie. Met een gemiddelde lengte van 1,5 minuut per cel zal er 9807 minuten aan videomateriaal zijn (**6,86 TB**). Worden de cellen gecapteerd met een bodycam (10 minuten per cel) dan is er 65.380 minuten aan videomateriaal (**45,77 TB**). Ook nu moeten de video's worden omgezet naar *frames*. Dit gebeurt door de *Pre-processing* pipeline opgezet in set-up productie. Deze stap levert 5.884.200 *frames* op (**8,84 TB**). 65.380 minuten video levert 39.228.000 *frames* op (**58,85 TB**). Paragraaf 5.1.1.2 legt uit waarom de sequentie van *frames* in opslag groter is dan die van de video. Bijkomend worden 7.200 foto's genomen om het model verder te verbeteren. Dit levert **0.0108 TB** aan beeldmateriaal op. De annotaties nemen **0,00003 TB** opslag in. De SQL-database die wordt opgezet in paragraaf 5.1.1.1, zal een maandelijkse kost met zich bijdragen van €62 – €250 per maand met een opslag van 250 GB. Tenslotte vereist Azure DevOps minstens vijf licenties voor gebruik, waarbij elke licentie per jaar €75 bedraagt. De kostenberekening wordt weergegeven in Tabel 29.

Tabel 29: Kostenberekening productie – infrastructuur

Kosten infrastructuur						
Soort storage	Minimum aantal	Maximum aantal	Eenheid	Kost per eenheid	Minimum totale kost	Maximum totale kost
Video's op Hot laag (ADLS)	6,86 TB	45,77 TB	Per TB per maand	€85,00	€583,10	€3.890,45
Foto's en annotaties op Hot laag	8,84 TB	58,85 TB	Per TB per maand	€85,00	€751,40	€5.002,25
Foto's en annotaties op Archive laag	8,84TB	58,85 TB	Per TB per maand	€13,00	€114,92	€765,05
Azure SQL Database	250GB	250GB	Per maand	€62,00 - €250,00	€62,00	€250,00
Azure DevOps	5	5	Per licentie per jaar	€75,00	€375,00	€375,00
Totaal 18 maanden					€7.063,58⁵⁵	€39.860,15⁵⁶

Ook in productie zullen de rauwe video's slechts één maand worden bijgehouden op de hot laag omwille van juridische redenen. De afbeeldingen en annotaties worden wederom vier maanden gestockeerd op de hot laag, waarna ze 14 maanden lang beschikbaar zullen zijn op de Archive laag.

5.2.2 Processen

De videobeelden (9807 en maximaal 65.380 minuten) doorlopen dezelfde processen als in set-up productie: *Pre-processing*, *inference* en zwerfvuiltelling. Paragraaf 5.1.3 bespreekt deze processen en de gerelateerde kost in meer detail. Het vergt in totaal 69,19 en maximaal 960,72 dagen om alle processen te overlopen. Dit hoeft niet te betekenen dat de OVAM die termijnen op de resultaten van de zwerfvuilmonitoring moet wachten. Meerdere berekeningsinstanties kunnen opgezet worden in de Cloud waardoor de video's in parallel kunnen verwerkt worden. De berekeningskost wordt berekend in Tabel 30.

⁵⁵ 583,10 + 751,40 x 4 + 114,92 x 14 + 62,00 x 18 + 375,00 x 2

⁵⁶ 3.890,45 + 5.002,25 x 4 + 765,05 x 14 + 250,00 x 18 + 375,00 x 2

Tabel 30: Kostenberekening productie – processen

Kosten processen					
Soort proces	Uitvoeringstijd proces (minuten)	Maximum uitvoeringstijd proces (minuten)	Kost per minuut processing	Totale kost	Maximum totale kost
<i>Pre-processing</i>	40.797,12	271.980,80	€0,01	€407,97	€2.719,80
<i>Inference</i>	39.228,00	980.700,00	€0,08	€3.138,24	€78.456,00
<i>Post-processing</i>	19.614,00	130.760,00	€0,01	€196,14	€1.307,60
Totaal				€3.742,35	€82.483,40

5.2.3 Modelontwikkeling

Onder modelontwikkeling in productie worden de beeldopname, beeldannotatie, model training en model testing verstaan. Met de beeldopname wordt de effectieve monitoring bedoeld om de evolutie van zwerfvuil in kaart te brengen. Deze stap levert beeldmateriaal op, waarvan een deel zal gebruikt worden om het model verder te ontwikkelen.

Tabel 31: Kostenberekening productie – model development

Kosten model development		
Onderdeel	Totale kost	Maximale totale kost
Beelden opnemen	€29.762,46	€50.834,42
Beelden annoteren	€850,47	€850,47
Model trainen	€254,61	€386,12
Model testen	€19.200,00	€38.400,00
Totaal	€50.067,54	€90.471,01

5.2.3.1 Beelden opnemen

Het aantal mandagen nodig voor het in kaart brengen van de 6.538, 10 bij 10 cellen worden op dezelfde manier berekend als in set-up productie (paragraaf 5.1.6.1) met de uitzondering dat het zwerfvuil niet rondgestrooid en opgeruimd wordt. Er wordt rekening gehouden met de tijd nodig om een 10 bij 10 meter cel te filmen (opnametijd), de transporttijd tussen cellen (transporttijd) en het aantal cellen. Voor 6538 cellen is de beeldopnametijd 156,64 mandagen (Modus = auto) en maximaal 272,42 mandagen (Modus = bodycam). Voor deze actie is slechts één persoon nodig.

Er worden ook foto's gemaakt om het model verder te trainen en beter te maken. Het voordeel van manueel foto's nemen is de mogelijkheid om specifiek te focussen op zwakheden van het model. Hoeveel foto's dit zijn, hangt af van de accuraatheid van de uit set-up productie vergaarde modellen. Er wordt uitgegaan van 300 extra foto's per fractie, wat resulteert in 7.200 foto's. In de POC werden 1.568 foto's verzameld op zes uur tijd (paragraaf 4.2.1). Voor dit proces zijn twee personen vereist en dus is de opnametijd 6,89 mandagen.

Tabel 32: Kostenberekening productie – model development beelden opnemen

Kosten model development beelden opnemen					
Soorten beelden	Aantal mandagen	Maximum aantal mandagen	Kost per eenheid	Totale kost	Maximum totale kost
Extra model training	6,89	6,89	€182,00	€1.253,98	€1.253,98
Meetcampagne	156,64	272,42	€182,00	€28.508,48	€49.580,44
Totaal	163,53	279,31		€29.762,46	€50.834,42

5.2.3.2 Beelden annoteren

De zwerfvuilmonitoringcampagne levert 5.884.200 *frames* op (paragraaf 5.2.1). 0,10% van de *frames* laten annoteren resulteert in 5.884,20 *frames*. Een goede methode hierbij zou zijn om de *frames* te nemen waar de AI modellen het meeste zwerfvuilobjecten hebben gevonden. Een willekeurige selectie zou immers kunnen leiden tot een hoog percentage aan *frames* zonder zwerfvuil. Er wordt ook zelf uitgereden om foto's te nemen om zwakheden van het model weg te werken. Dit levert 7.200 foto's op. In totaal worden 13.084 foto's verzameld. Opnieuw wordt een factor van 1,30 genomen die het aantal zwerfvuilobjecten per foto aanduidt. Alleen externe annotatie wordt in de totale kost opgenomen omwille van het financiële voordeel ten opzichte van zelf te annoteren.

Tabel 33: Kostenberekening productie – model development beelden annoteren

Kosten model development beelden annoteren						
Annotatie methode	Aantal	Maximum aantal	Eenheid	Kost per eenheid	Totale kost	Maximum totale kost
Extern (gekozen optie)	17.009,46	17.009,46	Zwerfvuil-objecten in foto's	€0,05	€850,47	€850,47
Zelf annoteren	27,32	54,58	Mandagen	€182,00	€4.972,24	€9.933,56

5.2.3.3 Model training en testing

Nadat de beelden zijn geannoteerd, kan het model verder getraind en getest worden. Aangezien de modellen al gebouwd zijn, hoeft de data scientist niet tussen te komen. Er zal op 13.084 (= 5.884 *frames* + 7.200 foto's) beelden worden getraind. Wederom wordt er één tot twee mandagen gerekend voor het testen per fractie.

Tabel 34: Kostenberekening productie – model development training en testing

Kosten model development model trainen en testing						
Actie	Aantal	Maximum aantal	Eenheid	Kost per eenheid	Totale kost	Maximum totale kost
Hardware berekenningskosten	54,52	82,68	Uren berekeningstijd	€4,67	€254,61	€386,12
Model testing	24,00	48,00	Mandagen	€800,00	€19.200	€38.400
Totaal					€19.454,61	€38.786,12

5.2.4 Technische ondersteuning

In geval van technische problemen in de werking van het systeem zoals het inladen van de data, dient er ondersteuning beschikbaar te zijn om dit op te lossen. Er wordt verwacht dat het aantal interventies eerder beperkt zal zijn en de fouten de werking van het systeem **niet zullen ondermijnen**. De kost van de technische ondersteuning hangt af van de eisen van de OVAM waardoor een inschatting van een passende kost moeilijk realiseerbaar is. Bijgevolg wordt deze buiten beschouwing gelaten. Beslissingen die door de OVAM gemaakt zullen moeten worden omtrent de geleverde ondersteuning betreffen de snelheid van de service, de snelheid van oplossing, etc.

5.3 OVERZICHT KOSTEN

In Tabel 35 wordt een overzicht gegeven van de minimale totale kosten en de maximale totale kosten geïdentificeerd in de financiële studie. Er wordt een onderscheid gemaakt tussen de éénmalige en recurrente kosten. De éénmalige kosten zijn de investeringskosten die moeten gemaakt worden om de oplossing te bouwen. Deze bedragen tussen de €348.416 en €587.180. De recurrente kosten zijn de kosten die bij elke meetcampagne gemaakt zullen worden en bedragen tussen de €60.873 en €212.813.

Tabel 35: Totale kosten – minimaal (exclusief BTW)

Totale kosten								
	Infra-structuur	Hard-ware	Processen	Onderzoek en opzet	Beelden capteren	Beelden annoteren	Model	Totaal
Eenmalige kosten (opzet)	€24.854 ⁵⁷	€1.311	€79.211	€101.600	€8.226	€1.806	€131.408	€348.416
Recurrente kosten	€7.064		€3.742		€28.508 ⁵⁸	€850	€20.709 ⁵⁹	€60.873

Tabel 36: Totale kosten – maximaal (exclusief BTW)

Totale kosten								
	Infra-structuur	Hard-ware	Processen	Onderzoek en opzet	Beelden capteren	Beelden annoteren	Model	Totaal
Eenmalige kosten (opzet)	€40.145 ⁵⁷	€6.112	€155.295	€164.800	€10.083	€1.806	€208.939	€587.180
Recurrente kosten	€39.860		€82.483		€49.580 ⁵⁸	€850	€40.040 ⁵⁹	€212.813

Deze kosten worden vergeleken met de huidige kosten (exclusief BTW) die per meetcampagne worden gemaakt door de OVAM:

- Eenmalige kosten = **€39.774,96**
 - a. Overleg: €18.880,49
 - b. Uitwerken methodologie: €20.894,47
- Recurrente kosten = **€397.099,22**
 - c. Opleiding personeel: €4.158,98
 - d. Zwerfvuil verzamelen: €287.452,93
 - e. Zwerfvuil tellen: €105.487,31

Tabel 37 geeft de geaccumuleerde kost weer over de meetcampagnes. Indien het minimum (maximum) scenario zich afspeelt, is de investering over twee (vier) meetcampagnes terugbetaald. Het zijn voornamelijk de ontwikkelingskosten die doorwegen in de geprojecteerde kosten. De recurrente kosten worden met een factor acht (twee) gereduceerd indien overgestapt wordt op het beoogd systeem.

⁵⁷ Totale infrastructuurkost + kost bouw API-endpoint

⁵⁸ Kost opname beeldmateriaal, exclusief extra model training

⁵⁹ Kost extra model training + hardware berekeningskosten + model testing

Tabel 37: Projectie van de geaccumuleerde kosten van meetcampagnes

	Meet-campagne 1	Meet-campagne 2	Meet-campagne 3	Meet-campagne 4
Toekomstige methode - MIN	€409.289	€470.162	€531.035	€591.908
Toekomstige methode - MAX	€799.993	€1.012.806	€1.225.619	1.438.432

Het is belangrijk te vermelden dat de geprojecteerde kosten gebaseerd zijn op inschattingen. Ordina heeft getracht alle componenten die vereist zijn voor de ontwikkeling en het onderhoud van het toekomstig systeem te identificeren en zo goed mogelijk in te schatten. Uiteraard moeten hiervoor assumpties worden gemaakt. Wijzigingen in deze assumpties kunnen een grote impact hebben op de uiteindelijke geprojecteerde kost, vb. voor de kost gerelateerd aan de opnametijd in productie (paragraaf 5.2.3.1) werd aangenomen dat één werkracht zou moeten worden om een 10 bij 10 meter cel in kaart te brengen. Toch is het mogelijk dat twee werkrachten in dit proces betrokken zullen worden.

5.4 RISICO'S

Ordina definieert drie risico's die een invloed kunnen hebben op het te ontwerpen systeem en bijgevolg ook de geprojecteerde kost. Deze worden in kaart gebracht door een bepaalde kans en impact aan het risico toe te wijzen.

Tabel 38: Risicotabel

3	Middel	Hoog	Kritiek
2	Laag	Middel	Hoog
1	Laag	Laag	Middel
	1	2	3

Risico	Beschrijving	Impact	Kans	Level	Kost
Faling methodologie	De geïdentificeerde methodologie moet herwerkt worden	3	1	Middel	€46.000
Compatibiliteit	De integratie van pipelines van de verschillende processen stuit op problemen waardoor code herwerkt moet worden.	3	1	Middel	€8.640 – €31.010
Faling werking systeem	Tijdens het uitvoeren van een meetcampagne treden technische problemen op	3	2	Middel	/

Wanneer het eerste risico 'faling methodologie' zich afspeelt, moet er in principe opnieuw een haalbaarheidsstudie worden uitgevoerd om een nieuw conceptueel plan op te stellen en vervolgens een POC ontwikkeld worden om de werking van het nieuwe idee aan te tonen. Uiteraard zal deze studie niet de omvang hebben van het door Ordina uitgevoerd project, maar zal zich focussen op de gefaalde componenten. Er wordt de assumptie gemaakt dat 40% van de kost van dit project zal nodig zijn om de componenten te herwerken.

Indien comptabiliteitsproblemen opduiken moet een deel van de code van de drie processen (*Pre-processing*, *inference* en zwerfvuiltelling) herschreven worden. Gezien de basiscode er staat, zullen kleine aanpassingen nodig zijn zodoende comptabiliteit te verzekeren. Er wordt gerekend met 20% van de creatie kost van de processen.

Het optreden van het derde risico 'faling werking systeem' ondermijnt de werking van het systeem waardoor technische ondersteuning dringend vereist is. Mogelijke oorzaken die de werking van het systeem hinderen zijn: integratie en/of installatie problemen van de verschillende hardwarecomponenten tijdens het monitoren van gebieden of het identificeren van onlogische resultaten waardoor een intensieve doorlichting van het systeem dient te gebeuren. Aangezien de mogelijke problemen uiteenlopend zijn, is het onmogelijk om dit risico in te schatten. Toch dient de OVAM op de hoogte te zijn van dit mogelijk risico.

6 BESLUIT

Na de technische en juridische haalbaarheidsstudie (paragraaf 0 en 3), werd in de POC aangetoond dat de verschillende componenten nodig voor een end-to-end zwerfvuildetectiesysteem werken. Mits **extra inspanningen** kunnen deze onderdelen gecombineerd worden tot een productiewaardige oplossing. Twee mogelijke end-to-end systemen worden gedefinieerd. Beide end-to-end systemen omvatten geen noemenswaardige juridische hindernissen. Enerzijds door de doelstelling van het AI systeem, nl. zwerfvuilmonitoring, waardoor dit systeem dus niet tot doel heeft persoonsgegevens van eventuele betrokkene-passanten te verwerken. Anderzijds door de keuze van beeldcaptatie, nl. geen drones, waardoor wetgeving in het kader van drones niet van toepassing zijn.

Het eerste end-to-end systeem is een relatief goedkoop maar weinig robuust systeem. Dit systeem stapt af van objectlokalisatie omwille van de slechte ZED-camera kwaliteit (paragraaf 4.1.2) en slechte GPS-accuraatheid (paragraaf 2.2). Bijgevolg wordt dubbeltelling op basis van sequentiële *frames* (paragraaf 2.4.1) verholpen door object tracering (paragrafen 2.4.3 en 4.3.2). Dubbeltelling op basis van meervoudige passage wordt vermeden door de routes op efficiënte wijze te bepalen. De mapping van het gedetecteerde zwerfvuil met de 10 bij 10 meter cel gebeurt door de camerolocatie, aangegeven door de ArduSimple GPS, te linken aan de dichtstbijzijnde cel.

Het staat buiten kijf dat deze methode een aantal problemen heeft. Ten eerste valt het te betwijfelen dat het mogelijk is om in één passage al het zwerfvuil te filmen door onder meer zicht versperrende objecten zoals voorbijrijdende auto's en fietsers die niet kunnen vermeden worden. Ook kan met deze aanpak niet vanuit meerdere perspectieven gefilmd worden. Er is maar één kans om het zwerfvuilobject goed in beeld te krijgen. Ten tweede zal de mapping van het zwerfvuilobject met de juiste 10 bij 10 meter cel inaccuraat zijn. Bovenop de fout van de GPS, wordt geacht dat het zwerfvuil zich op dezelfde locatie bevindt dan de camera zelf. Beide fouten opgeteld leiden tot een lage betrouwbaarheid voor de geografische locatie van een object.

Het tweede duurdere maar meer robuuste end-to-end systeem is een systeem dat objectlokalisatie gebruikt om de absolute plaats van zwerfvuilobjecten te bepalen. Door deze absolute plaatsbepaling kan dubbeltelling opgevangen worden. Dit systeem laat toe meermaals door de 10 bij 10 m cel te rijden om het zwerfvuil zo goed mogelijk in kaart te brengen. Ook is het mogelijk om een goede link te maken met de desbetreffende 10 bij 10 m cel. In paragraaf 4.3 werd aangetoond dat het mogelijk is om dit systeem te ontwikkelen, mits de oplossing van twee problemen:

1. Stereo-fotogrammetrie

Ten eerste moet een eigen stereo-fotogrammetrisch systeem worden gebouwd, tenzij een ZED wordt ontwikkeld met betere beeldkwaliteit (paragraaf 4.1.2) of andere, niet onderzochte stereocamera's, een beter alternatief vormen. De vereisten en problemen van zo een systeem staan respectievelijk in paragrafen 2.4.2.1.2 en 7.2.1. Een inschatting van deze kost staat gebudgetteerd in de maximale creatie kost van de processen in de financiële studie (paragraaf 5).

2. GPS-accuraatheid

Ten tweede is object lokalisatie om dubbeltelling te voorkomen alleen mogelijk mits een meer granulaire geolocatiebepaling beschikbaar is. Huidige technologieën voor locatiebepaling, zoals GPS, zijn niet accuraat genoeg waardoor dubbeltelling nog steeds mogelijk is, zelfs met een goede stereo-fotogrammetrische opstelling. Het is dus aan te raden om deze optie verder te onderzoeken eenmaal een precieze locatiebepaling tot de mogelijkheden behoort. Dit zou aangeleverd kunnen worden door het toekomstig Galileo project (paragraaf 2.2.2.3).

Samengevat accepteert de eerste oplossing de gebreken van de hardware, wat een gebrekkige werking impliceert en is de tweede oplossing een meer robuust, toekomstgericht systeem.

Naast de keuze tussen beide systemen zijn er nog drie bijkomende uitdagingen die een impact zullen hebben op de haalbaarheid van het beoogd systeem:

1. Framekwaliteit

De framekwaliteit is een pijnpunt door het gebrek aan scherpstellingmogelijkheden in video's (paragraaf 4.1). Zolang de afstand van de camera tot het zwerfvuilobject beperkt blijft, zal dit kwaliteitsverlies niet doorwegen (paragraaf 4.2.5). Echter wanneer de afstand stijgt, zal dit een negatieve impact hebben op de detectie van zwerfvuil. Dit is een probleem voor een opstelling van een camera op een wagen, wat de gewenste transport modus is. Het gebruik van betere camera's is bijgevolg aangewezen. In paragraaf 4.1.5 staan drie specificaties waarmee rekening gehouden dient te worden bij de keuze van een camera alsook een mogelijk vervolgtraject.

2. Transport modus

Een meer operationeel probleem is de efficiëntie van de transport modus (paragraaf 2.1.2). De modus die onze voorkeur draagt, is de constructie van een camera op een wagen. Deze modus valt immers het meest te automatiseren, wat leidt tot een groot kostvoordeel (paragraaf 5.2.3.1). Toch verwacht Ordina dat met deze modus het meeste zwerfvuil zal gemist worden door objecten die het zicht versperren. Zelfs al kan in een cel meerdere malen rondgereden worden (tweede mogelijkheid end-to-end systeem), dan nog zullen bepaalde plaatsen onbereikbaar zijn met een wagen. Een mogelijkheid voor de OVAM is om het percentage gemist zwerfvuil te meten per type omgeving en de resultaten van het systeem hiernaar te corrigeren met een extrapolatiefactor.

Een andere oplossing werd besproken in paragraaf 2.1.2: de bouw van een flexibel systeem met fietsen en bodycams. Echter is dit alleen mogelijk mits gekozen wordt voor het tweede end-to-end systeem. Het systeem moet immers dubbeltelling door meerdere passage kunnen voorkomen omdat er geen garantie is dat eenmaal rondgereden wordt met de auto, alleen ongeziene zwerfvuilobjecten worden gecapteerd met de fiets of bodycam. Zoals hierboven toegelicht heeft zo een systeem een granulaire geolocalisatie, IMU en minstens twee camera's, in het geval dat er geen goede stereocamera's zijn, nodig die ten opzichte van elkaar niet wijzigen in ruimtelijke zin. Het is duidelijk dat dergelijke constructie op een bodycam of fiets moeilijk realiseerbaar is. Indien deze oplossing gekozen wordt, dient een specifiek harnas gebouwd te worden die zich toespitst op comfort.

3. De 'rest' fracties

In de lijst van 24 fracties gedefinieerd door de OVAM bevinden zich fracties die door een AI-systeem onmogelijk te detecteren zijn. Meer specifiek zijn de fracties die een moeilijkheidsniveau van 13 hebben gekregen in paragraaf 4.2.4, te onduidelijk en breed van karakter. Onder dit moeilijkheidsniveau bevinden zich de 'rest' fracties, vb. ander papier en karton. Al is de inclusie van deze fracties evident, in een AI-systeem is dit zonder specifieke definitie onmogelijk te herkennen en dient bijgevolg herzien te worden door de OVAM.

Voor de uitdaging omtrent de modi van captatie zorgt voor de grootste uitdaging. De door Ordina toegereikte oplossingen moeten intern door de OVAM verder worden doorgelicht. Indien hier geen passend antwoord op gevonden wordt, heeft een verdere ontwikkeling van het beoogd systeem weinig zin.

Naast het operationele en technische aspect, is er uiteraard nog het financiële. De ontwikkelingskosten bedragen tussen de €348.416 - €587.180 en de recurrente kosten liggen tussen €60.873 - €212.813. Het grote verschil in minimum en maximum kost is toe te schrijven aan de keuzes die de OVAM nog dient te maken. Enerzijds op vlak van end-to-end systeem, anderzijds de resterende uitdagingen zoals de rest fracties. Ter illustratie kan de wijze van data captatie genomen worden. Wordt er gekozen om alle cellen te monitoren met behulp van een bodycam dan zullen de recurrente kosten naar het maximumbedrag hellen, wordt er gekozen om zelf te annoteren dan zullen de kosten aanzienlijk verhogen, etc. Een apart detail overzicht werd aan de OVAM geleverd om hen in staat te stellen de verschillende scenario's financieel door te lichten.

De financiële studie werd zo opgesteld dat een vergelijking tussen het huidige en toekomstige systeem kon gemaakt worden. Indien het minimum scenario zich afspeelt, is de investering na twee meetcampagnes terugbetaald. Het maximum scenario betaalt zich terug na vier meetcampagnes. Er is dus wel degelijk een business case om dergelijk systeem te bouwen.

Concluderend kan gesteld worden dat het beoogde systeem financieel interessant is, maar door operationele en hardware-technische problemen op dit moment nog niet volledig haalbaar lijkt. Ordina raadt dan ook aan de operationele kwestie omtrent de modi van captatie intern uit te klaren en te wachten met een verdere ontwikkeling van het systeem tot een preciezer locatiebepaling mechanisme voor handen is of extra onderzoek in dit veld zijn uitgevoerd.

7 BIJLAGEN

7.1 SPECIFIEKE ONDERZOEKSVRAGEN

7.1.1 Technisch

- Is het met de huidige technologie op de markt mogelijk om zwerfvuilitems te classificeren? Met welke accuraatheid en precisie?
- Kiezen we best voor een SaaS-oplossing? Of beter voor iets anders?
 - Gebruiken we best bestaande infrastructuur/services?
- Welke technische opties zijn er, bijvoorbeeld m.b.t. opslag van het beeldmateriaal? Hoe wordt het beeldmateriaal best verzameld?
- Hoeveel data verzamelen we op die manier?
- Hoeveel kost het om de data op te slaan, te consulteren, enz.?
- Waar moeten de data worden opgeslagen? Kan dit op de OVAM-servers?
- Hoe lang kunnen de data worden opgeslagen, bovenop de minimumbewaringstermijn
- Hoe verzekeren we de vergelijkbaarheid van de verschillende metingen over de jaren heen?
- Hoe kunnen we ervoor zorgen dat we de 10x10 m cellen voldoende respecteren?
- Bestaan er systemen die – met voldoende accuraatheid – kunnen herkennen dat verschillende losse stukken afkomstig zijn van één object?
- Bij het gebruik van een neurale netwerk:
 - hoe en wanneer verloopt de training? Wordt het systeem stelselmatig verbeterd, na elke meetcampagne (elk jaar, elk seizoen)?
 - Is het mogelijk om dit via *unsupervised learning* (voorkeur) te doen?
 - Kunnen we een pretrained model voor beeldherkenning gebruiken?
 - Als het netwerk naarmate de tijd vordert accurater wordt: Wat is de impact hiervan op de vergelijkbaarheid tussen de verschillende jaarresultaten?
 - Hoe vermijden we *overfitting*?
- Kunnen de drones en/of het systeem van het Agentschap voor Natuur en Bos worden ingezet voor (een deel van) de metingen?

7.1.2 Financieel

- Is er een business case?
- Voor elk van de uitgewerkte mogelijkheden of opties:
 - Hoeveel kost de ontwikkeling en de uitrol van een volwaardig systeem, vertrekkend vanuit de proof-of-concept?
 - Wat is de total cost of ownership?

- ▶ Hoeveel kost het onderhoud van dit systeem in productie per jaar (inclusief uitvoering van de metingen)?
- ▶ Hoeveel kost het verzamelen van het beeldmateriaal per cel van 10x10 m?

7.1.3 Juridisch

- Met welke juridische aspecten moeten we rekening houden?
- Persoonlijke levenssfeer?

7.2 TECHNISCHE UITDIEPINGEN

7.2.1 Praktische Stereo-fotogrammetrische problemen

7.2.1.1 Overlap

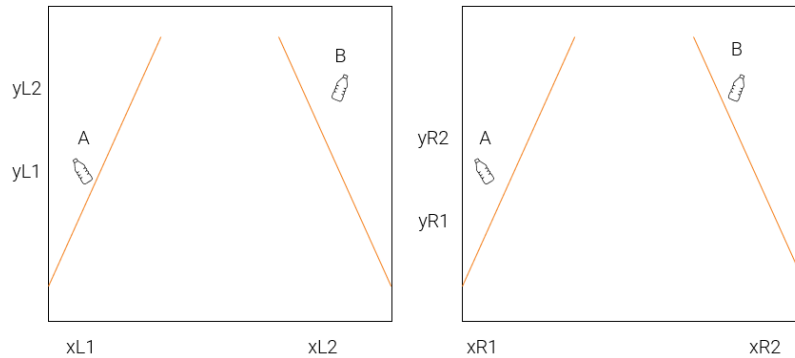
Om stereo-fotogrammetrie toe te passen moet een pixel op minimaal twee afbeeldingen te zien zijn. Daarnaast mogen de twee afbeeldingen niet uit exact dezelfde positie genomen zijn. Dit betekent dus dat er een overlap verzekerd moet worden.

Om dit toe te passen op dit project moet er dus nagedacht worden over hoe we bewust en gecontroleerd overlap kunnen introduceren. Er zijn twee mogelijkheden om dit te bekomen:

- Werken met twee camera's (linker en rechter oog) om een frame paar te bekomen.
- Werken met één camera en sequentiële video *frames* gebruiken als frame paren.

De eerste optie is relatief eenvoudig te implementeren. Indien twee camera's worden opgesteld die hetzelfde beeld capteren bekomen we een analoge opstelling als deze van menselijke ogen. Door beide camera's goed te positioneren kunnen twee afbeeldingen bekomen worden met een grote overlap. Deze twee afbeeldingen samen noemen we verder het frame paar. Voordeel van deze opstelling is dat er geen complexiteit ontstaat om overlap te introduceren. De afbeeldingen zullen ook telkens hetzelfde beeld in kaart brengen waardoor de bewegingssnelheid geen invloed heeft op de overlap. Nadeel is dat er extra hardware noodzakelijk is alsook dat het datavolume verdubbelt, en de kostprijs voor opslag dus ook verdubbelt. Een frame paar zou er kunnen uitzien zoals in Figuur 74. Het is duidelijk dat de objecten A en B licht verschuiven in de rechter afbeelding ten opzichte van de linker afbeelding. Deze verschuiving is het gevolg van de andere positie van de linker camera ten opzichte van de rechter camera. De beelden zelf zijn echter zeer gelijkaardig en capteren ongeveer dezelfde locatie. Op deze manier kan dus eenvoudig een grote, stabiele overlap gecreëerd worden.

Figuur 74: Twee camera's: frame paren

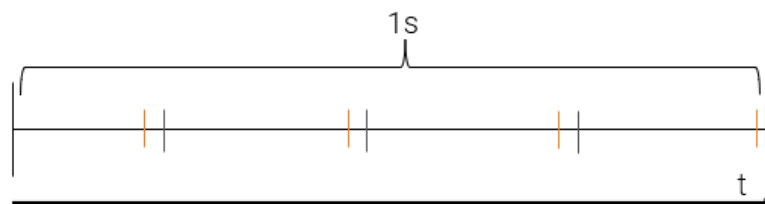


De tweede optie, werken met één camera verhoogt de conceptuele complexiteit aanzienlijk. Eerst en vooral is het minder evident om hier frame paren te introduceren. Bij optie 1 werden frame paren bekomen door twee foto's vanuit verschillende posities tegelijk te nemen. Dit vereist uiteraard twee camera's, wat in deze optie niet voor handen is. Daarom is het moeilijker om twee afbeeldingen met voldoende overlap, het frame paar, te bekomen. Het is immers nodig om twee *frames* te hebben die dicht bij elkaar aansluiten om een maximale overlap te hebben, maar waarbij de camera positie voldoende verschilt van de vorige.

Daar er maar één camera beschikbaar is in deze methode moeten dus snel opeenvolgende *frames* genomen worden en gekoppeld worden tot een frame paar. Dit impliceert direct dat stilstand een probleem vormt voor deze methode, maar ook te snelle bewegingen kunnen hier voor problemen zorgen. Stilstaande beelden zouden immers leiden tot identieke beelden waardoor stereo-fotogrammetrie onmogelijk wordt. Bij te hoge verplaatsingssnelheid zouden de beelden niet voldoende overlap hebben voor de berekeningen. Bij keuze voor deze methode moet dus een minimum en maximumsnelheid worden bepaald waarbinnen *frames* bruikbaar zijn.

Bovendien zal proefondervindelijk een bepaling moeten gevonden worden om frame paren samen te stellen. Indien we werken met vier fps als normale werking, betekent dit eigenlijk dat we acht *frames* zullen gebruiken (de oorspronkelijke vier en een naburige frame voor maximale overlap). Dit concept is geïllustreerd op Figuur 75. Hierbij stelt de frame van de oranje lijn de normale werking voor, terwijl de blauwe streep het bijhorende koppel voorstelt. De afstand tussen het koppel is afhankelijk van de snelheid waarmee gereden wordt. Een goede waarde hiervoor zou proefondervindelijk bepaald moeten worden.

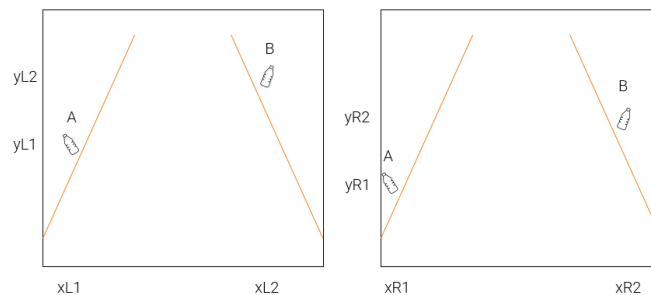
Figuur 75: Tijdsaspect frame paren bij één camera



Een dergelijk frame paar zou er uit kunnen zien zoals geïllustreerd op Figuur 76. Het is duidelijk voor dit frame paar dat de objecten op de linker frame naar voor zullen schuiven op de rechter frame bij een voorwaartse

beweging van de camera. Hoeveel de objecten naar voor schuiven zal afhangen van de tijd tussen de *frames* van het paar en de snelheid waarmee bewogen wordt.

Figuur 76: Eén camera: frame paren

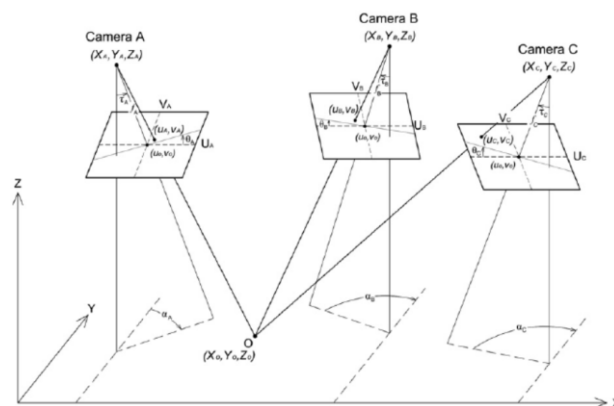


7.2.1.2 Ruimtelijke oriëntatie van de camera's

In paragraaf 2.4.2.1.2 wordt uitgegaan van perfect georiënteerde lenzen. In de praktijk zal dit wellicht niet het geval zijn. Beide lenzen kunnen lichte rotaties vertonen in drie richtingen, zoals weergegeven op Figuur 77. Dit is geldig voor zowel de situatie met één camera als de situatie met twee camera's.

In de fotogrammetrie wordt dit probleem praktisch verholpen door het gebruik van een aantal "tiepoints" of punten die men duidelijk op twee foto's herkent en die mooi verspreid zijn over de gehele foto. Op basis van die tiepoints kunnen de zes relatieve rotatieparameters bepaald worden, en kan men de relatieve ruimtelijke oriëntatie van de lenzen ten opzichte van elkaar bepalen. Het bepalen van die tiepoints is een manuele taak die voor elke nieuwe foto moet gebeuren. Dit is praktisch onmogelijk om te realiseren voor de beoogde opstelling van dit project.

Figuur 77: Ruimtelijke oriëntatie van de camera's⁶⁰



⁶⁰ https://www.researchgate.net/figure/Pinhole-camera-model-for-stereo-imaging-illustrating-the-collinearity-relationship_fig1_257211329

Bij de ontwikkeling van een eigen algoritme voor stereo-fotogrammetrie is het denkbaar om, voor deze specifieke toepassing, deze rotatie parameters te verwaarlozen. In paragraaf 2.1.2 is immers beslist om te werken met grond-modi. Afwijkingen als gevolg van camera rotaties zullen beperkt blijven voor een beperkt camera bereik. Dit probleem komt immers vooral voor bij een groot camerabereik, zoals bijvoorbeeld bij drones of vliegtuigen het geval is. Het verwaarlozen van deze rotaties kan wel aanleiding geven tot een afwijking op de bepaling van de relatieve locatie van het object ten opzichte van de camera.

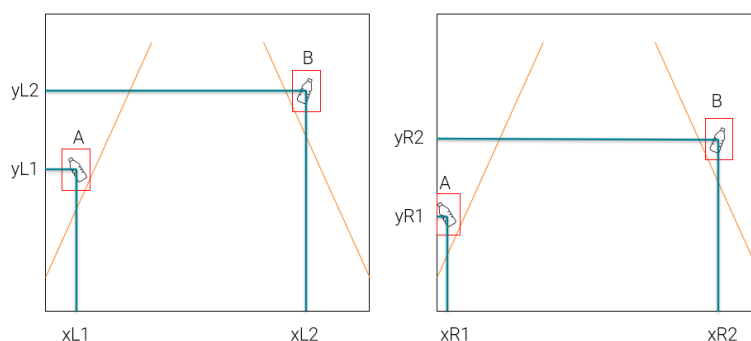
7.2.1.3 Bepaling van de projecties van het te zoeken punt in het frame paar

Om een voorwaartse insnijding te kunnen doen moet de projectie van een werkelijk punt op twee afbeeldingen gekend zijn, het frame paar. Wat een frame paar is en hoe dit bekomen wordt kan worden nagelezen in paragraaf 7.2.1.1. De projecties van een reëel punt A werden op Figuur 30 aangeduid als de punten a_l (op de linkse afbeelding) en a_r (op de rechtse afbeelding). De voorwaartse insnijding gaat er dus van uit dat we weten waar op de foto paren die projecties zich bevinden. In de topografie worden deze punten softwarematig bepaald. Dit gebeurt door een aantal referentiepunten op foto's aan te duiden. Op deze manier kan de software de foto's oriënteren en punten selecteren die in beide afbeeldingen voorkomen.

In dit project kan echter niet op deze manier gewerkt worden. Dat zou immers een manuele actie vereisen voor elk foto paar. Gelukkig kan wel gebruik gemaakt worden van het resultaat van de AI modellen. Deze modellen detecteren immers net die objecten op die foto waarvan de relatieve positie dient bepaald te worden. Dit betekent dat het probleem herleid wordt tot het koppelen van gedetecteerde objecten op frame paren. In Figuur 78 wordt een frame paar in meer detail bekeken. Voor de eenvoud en het begrip wordt hier uitgegaan van het scenario waarbij het AI model beide objecten detecteert in beide *frames*.

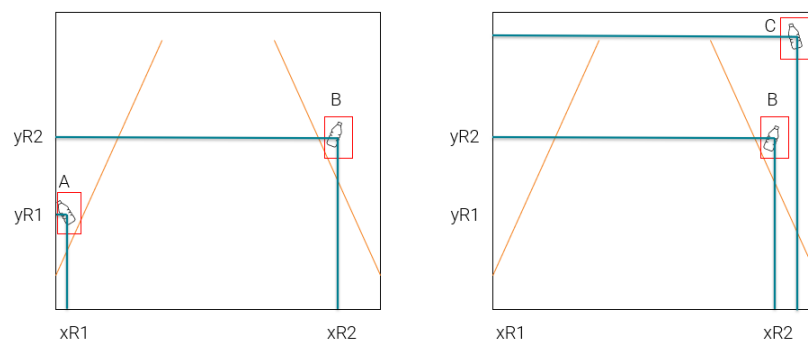
Omdat er meerdere objecten gedetecteerd kunnen worden is het van belang om een matching algoritme te maken opdat de gedetecteerde objecten op een correcte manier gekoppeld worden. Indien het frame paar nauwelijks verschillen vertoont, kan hiervoor een eenvoudig algoritme worden gebruikt. Door de tekens van de relatieve posities van de verschillende objecten ten opzichte van elkaar te beschouwen in beide *frames* van het paar kan men een volgorde van objecten vastleggen. Het is immers ondenkbaar dat een object A op een andere frame opeens een hogere y waarde zou bekomen dan object B. De tekens van de posities blijven dus ten alle tijden behouden.

Figuur 78: Gemeenschappelijke punten bepalen in frame paar



Voor een systeem met twee camera's kan een dergelijk algoritme dus eenvoudig geschreven worden. Voor een systeem met één camera zijn er echter nog bijkomende problemen. Het is bijvoorbeeld denkbaar dat een object A van een tweede frame verdwijnt terwijl een object C bovenaan de tweede frame verschijnt. Deze situatie is voorgesteld in Figuur 79. Bij gebruik van hetzelfde algoritme zal dit immers detecteren in een matching (A,B) & (B,C). De relatieve posities zijn immers hetzelfde.

Figuur 79: Probleem van object matching bij één camera



Het is heel moeilijk om dit probleem volledig te elimineren. Wel zouden een aantal eenvoudige controles ingebouwd kunnen worden die het probleem drastisch zouden reduceren:

- Objecten in een frame paar kunnen niet wijzigen van classificatie.
- Objecten in een frame paar kunnen niet achterwaarts bewegen.

De eerste controle is vanzelfsprekend. Moest een object C een andere classificatie hebben dan object B kan eenvoudig gesteld worden dat beide *frames* minstens één ander object in beeld hebben en dus eigenlijk geen geschikt koppel zijn.

De tweede controle brengt een nieuwe beperking mee naar de datacaptatie. Dit betekent immers dat het voertuig niet achteruit mag rijden. Als de camera achteruit beweegt, kan een object immers wel achterwaarts bewegen op opeenvolgende *frames*.

Deze twee controles zullen, zoals reeds aangegeven, niet alle fouten uitsluiten. Het is nog steeds denkbaar dat bijvoorbeeld door te snelle verplaatsingen dat het frame paar andere objecten ziet die wel aan beide controles voldoen. Deze problematiek is rechtstreeks verbonden met de problematiek van dubbeltellingen. De enige optie om dit te vermijden is het reguleren van de snelheid waarmee het voertuig zich mag verplaatsen.

7.3 RESULTATEN MODEL

7.3.1 Blikjesmodel









7.3.2 Sigarettendoosjes model













7.4 GEGEVENSBESCHERMINGSEFFECTBEOORDELING (GBEB)

In dit onderdeel van de bijlage wordt de GBEB toegevoegd. Ordina heeft de tabellen waar mogelijk en toegelaten ingevuld. De resterende, niet ingevulde vakken, dienen door de DPO van de OVAM aangevuld te worden indien de OVAM overgaat tot het effectief bouwen van het AI systeem.

7.4.1 Project overzicht

Samenvatting van het project	De OVAM is een Vlaamse overheidsdienst die ervoor zorgt dat we in Vlaanderen op een doordachte en milieubewuste manier omgaan met afval, materialen en bodem. Wij geven richting aan het beleid rond afval, materialen en bodem en beïnvloeden zo de uitvoering van de wetgeving. Het project in kwestie betreft de detectie en telling van zwerfvuil, gebruikmakend van artificiële intelligentie.
Verwerkingsdoeleinden	Na een uitvoerige voorstudie, wil OVAM nu de haalbaarheid van AI-beeldherkenning als robuust monitoringsinstrument voor zwerfvuil onderzoeken. Door middel van de studie kan OVAM de kosten en baten afwegen en een gefundeerde beslissing maken voor een eventueel vervolgtraject.
Rechtsgrond voor de verwerking	Deze verwerking vindt haar rechtsgrond in artikelen 6.1.e en 6.3 Algemene Verordening Gegevensbescherming (hierna: AVG), vermits de verwerking noodzakelijk is ter uitvoering van het Samenwerkingsprotocol tussen het departement Omgeving, afdeling Handhaving en de OVAM dd. 30 januari 2019 en artikelen 10.3.2 tot en met 10.3.4 van het Decreet van 5 april 1995 houdende algemene bepalingen inzake milieubeleid. Verder vindt er ook een mogelijke bijkomstige verwerking plaats van de geo-locatie van de OVAM-werknemers, die haar rechtsgrond vindt in artikel 6.1.b AVG en de arbeidsovereenkomst gesloten met deze werknemers. De geo-lokalisatie van de OVAM-werknemers is in elk geval niet mogelijk in real-time.

7.4.2 Identificeer de noodzaak van een DPIA

7.4.2.1 Doelstellingen en voordelen van het project

Wat wil het project bereiken?	Het project heeft tot doel zwerfvuilmonitoring te optimaliseren aan de hand van artificiële intelligentie.
Wat zijn de verwachte voordelen voor de [onderneming/bedrijf], individuen en/of andere partijen?	Het beter kunnen mooi/proper houden van stedelijke gebieden door snellere en efficiëntere zwerfvuilophaal.

7.4.2.2 Voorafgaande screening

Vraag	Antwoord/commentaar
Heeft het project betrekking op een of meer van de volgende zaken? -systematische en uitgebreide evaluatie van persoonlijke aspecten met betrekking tot natuurlijke personen op basis van geautomatiseerde verwerking (met inbegrip van profilering) waarop besluiten worden gebaseerd die rechtsgevolgen hebben voor de betrokkene of die hem of haar in aanzienlijke mate beïnvloeden -verwerking op grote schaal van persoonsgegevens van speciale categorieën of gegevens met betrekking tot strafrechtelijke veroordelingen en strafbare feiten -systematische bewaking van een voor het publiek toegankelijk gebied op grote schaal	Nee.
Gaat het bij de verwerking om het gebruik van nieuwe technologieën (inclusief kunstmatige intelligentie) of om de nieuwe toepassing van bestaande technologieën?	Ja.
Zal beslissingen nemen over de toegang van een individu tot een product, dienst, kans of voordeel: -zijn in enige mate gebaseerd op geautomatiseerde besluitvorming (met inbegrip van profilering) of -de verwerking van gegevens van speciale categorieën te betrekken?	Nee.
Gaat het project gepaard met het op grote schaal profileren van individuen?	Nee.
Verwerkt u biometrische gegevens?	Nee, hoewel de beeltenis van mogelijke betrokkenen-passanten kan opgenomen worden op de camerabeelden, worden deze beelden niet verder verwerkt ter bekoming van biometrische gegevens. Integendeel, er wordt softwarematige <i>blurring</i> toegepast op het gezicht en bij extensie het lichaam van mogelijke betrokkenen-passanten op de camerabeelden, ter minimalisering van enige mogelijke verwerking van persoonsgegevens.
Verwerkt u genetische gegevens (anders dan die welke door een huisarts of gezondheidsprofessional worden verwerkt voor het rechtstreeks verstrekken van gezondheidszorg aan het individu)?	Nee.

Vraag	Antwoord/commentaar
Gaat u zich bezighouden met het matchen van gegevens, d.w.z. het combineren en vergelijken van alle matchende persoonlijke gegevens die zijn verkregen uit meerdere bronnen?	Nee.
Verwerkt u persoonsgegevens die niet rechtstreeks van de betrokkene zijn verkregen in omstandigheden waarin het onmogelijk of onevenredig is om te voldoen aan de informatievereisten van artikel 14 van GDPR?	Nee.
Gaat de verwerking gepaard met het volgen van de geo-lokalisatie of het gedrag van een individu, met inbegrip van maar niet beperkt tot de online omgeving?	Nee, de geo-lokalisatie van de OVAM-werknemers is niet in <i>real-time</i> achterhaalbaar, maar kan wel achterhaald worden tijdens de analyse van de beelden in combinatie met de <i>timestamps</i> . Dit is echter een gevolg van de verwerking en maakt geen deel uit van het verwerkingsdoel.
Zal u zich richten op kinderen of andere kwetsbare personen: -voor marketingdoeleinden, profilering of andere geautomatiseerde besluitvorming, of -met de bedoeling om rechtstreeks online diensten aan te bieden aan kinderen	Nee.
Indien de voorgestelde verwerking van dien aard is dat een inbreuk op de persoonsgegevens de fysieke gezondheid of veiligheid van personen in gevaar kan brengen?	Nee.
Kan de verwerking anders leiden tot een groot risico voor de rechten en vrijheden van natuurlijke personen?	Neen, want er zijn adequate technologische en organisatorische maatregelen genomen ter risicovermindering. De camera's zijn naar beneden, naar de grond, gericht. Het softwarematig <i>blurren</i> van de gezichten en bij extensie het lichaam van mogelijke betrokkenen-passanten. De beelden/GPS-coördinaten worden niet in <i>real-time</i> opgevolgd, maar pas achteraf verwerkt. Het beperken van de toegang tot het systeem houdende de beelden, GPS-coördinaten en andere data. Het verwerkingsdoel betreft de optimalisering van zwerfvuilophaal, alsook het proper houden van de straten.

7.4.2.3 Conclusie

Zijn er naast de screeningsvragen nog andere relevante factoren waarmee rekening moet worden gehouden? Zo ja, geef dan nadere informatie	Nee.
Is het project, rekening houdend met alle factoren in dit hoofdstuk 3, waarschijnlijk een hoog risico voor de rechten en vrijheden van natuurlijke personen?	Zo ja, dan is een DPIA vereist. Als de organisatie een functionaris voor gegevensbescherming (DPO) heeft, moet u bij het uitvoeren van een DPIA het advies van de DPO in acht nemen. Nee. Er is geen hoog risico aanwezig, zelfs indien een data breach plaats zou vinden. Er worden geen bijzondere categorieën van gegevens verwerkt; er worden geen persoonlijke data van betrokkenen-passanten verwerkt want de camera's zijn naar beneden gericht en mogelijke betrokkenen-passanten worden softwarematig geblurred; de beelden en daarbij horende GPS-coördinaten worden niet in real-time verwerkt, maar pas achteraf; de toegang tot het systeem houdende de beelden en GPS-coördinaten is beperkt; het verwerkingsdoel betreft de optimalisering van zwerfvuilophaal, alsook het proper houden van de straten. Er worden echter wel persoonsgegevens verwerkt van de werknemers, in het bijzonder hun GPS-locatie op een bepaald tijdstip. Dit is echter een gevolg van het doel van de verwerking en gebeurt ook niet in real-time.
Is een DPIA vereist?	Neen, er zijn voldoende technische en organisatorische maatregelen genomen om mogelijke risico's te vermijden of minstens te verminderen.

7.4.3 Beschrijf de verwerking

Inzicht in de informatiestromen die bij een project betrokken zijn, is essentieel voor een goede beoordeling van de risico's voor de privacy. In dit deel wordt uiteengezet:

- hoeveel gegevens u gaat verzamelen en gebruiken
- hoe informatie wordt of zal worden verzameld, opgeslagen, gebruikt en verwijderd
- welke informatie wordt of zal worden gebruikt, waarom en wie er toegang toe heeft of zal krijgen (ook in welke geografische gebieden)

[voeg een beschrijving in van de manier waarop persoonlijke gegevens worden verzameld, gebruikt, bewaard en verwijderd. Het kan ook nuttig zijn om te verwijzen naar een stroomdiagram of een andere manier om de gegevensstromen te verklaren. U moet ook aangeven hoeveel personen waarschijnlijk door het project zullen worden beïnvloed.]

Het project betreft de automatisering van zwerfvuilmonitoring. De verwerking van persoonsgegevens maakt geen deel uit van het hoofddoel. Er vindt echter wel bijkomstige verwerking van persoonsgegevens plaats doordat er camera's (in het bijzonder: bodycams en vehicle cams) gebruikt worden op niet-besloten plaatsen. Deze verwerking wordt zoveel als mogelijk beperkt door het gezicht en het lichaam van mogelijke betrokkenen-passanten te *blurren*. De camera wordt ook altijd naar beneden gericht, gezien deze gebruikt worden om zwerfvuil te identificeren en te onderscheiden. Werknemers zijn voorzien van een GPS-module, maar er vindt geen live tracking plaats. De beelden en GPS-coördinaten worden pas nadien verwerkt, bij aankomst van de voertuigen/werknemers in het magazijn.

7.4.4 Overwegen om interne of externe belanghebbenden te raadplegen

7.4.4.1 Interne belanghebbenden

Wie gaat u raadplegen?	Hoe gaat u raadplegen?	Wanneer gaat u raadplegen?
[Project management team]	[Geef aan hoe u zult raadplegen]	[Geef aan in welk stadium van het project u zult raadplegen]
[DPO]		
[Engineers, developers and designers]		
[Information technology (IT)]		
[Procurement]		
[Potential suppliers and data processors]		
[Communications]		
[[client OR customer]-facing roles]		
[Corporate governance/compliance]		
[Researchers, analysts, and statisticians]		
[[Voeg andere toe]]		

7.4.4.2 Externe stakeholders

Wie gaat u raadplegen?	Hoe gaat u raadplegen?	Wanneer gaat u raadplegen?
[[beschrijving van de betrokken/voorbeeldgroepen [klant OF cliënt] groepen]]	[Geef aan hoe u zult raadplegen]	[Geef aan in welk stadium van het project u zult raadplegen]
[[beschrijving van de getroffen/voorbeeldgroepen invoegen]]		
[andere externe belanghebbenden toevoegen]		

7.4.5 Beoordeling van noodzaak en proportionaliteit

Vraag	Antwoord/commentaar
Bereikt de verwerking daadwerkelijk uw doel?	Ja, de verwerking is noodzakelijk om het project, met als doel de optimalisering van zwerfvalafvalverwerking, tot een goed einde te brengen.
Is er een andere manier om hetzelfde resultaat te bereiken?	Neen, er is geen andere, betere manier om dit doel te bereiken. Er worden echter voldoende technische en organisatorische maatregelen (TOM's) genomen om de veiligheid van de data te verzekeren.
Staat de verwerking in verhouding tot het doel dat u nastreeft?	Ja, de verwerking van persoonsgegevens is beperkt tot het uiterst noodzakelijke, vereist om het project tot een goed einde te brengen.

7.4.6 Identificeren en beoordelen van risico's

In dit deel wordt het huidige of potentiële karakter van de maatregel aangegeven:

- Privacy-risico's voor individuen
- wettelijke nalevingsrisico's, voornamelijk in verband met de algemene verordening inzake gegevensbescherming (GDPR)

Zodra de relevante risico's zijn geïdentificeerd, worden de mogelijke oplossingen geïdentificeerd en beoordeeld in punt 7. Een samenvatting van de relevante privacy-risico's en -oplossingen moet vervolgens worden geregistreerd in het register van privacy-risico's van het [bedrijf of de onderneming]. Het register van privacy-risico's moet worden bijgewerkt naarmate het project vordert.

7.4.6.1 Privacy Risico voor personen

Overweeg of de volgende risico's zich voordoen en zo ja, geef commentaar:

- De waarschijnlijkheid dat het risico zich voordoet (laag / gemiddeld / hoog)
- De impact op de persoon (personen) als het risico zich voordoet (laag / gemiddeld / hoog)

De verwerking zou kunnen bijdragen tot:	Waarschijnlijkheid	Impact	Nadere toelichtingen
onvermogen om rechten uit te oefenen (met inbegrip van maar niet beperkt tot privacyrechten)	L	H	Het doel van het proces zwerfvalmonitoring zou weinig tot geen impact mogen hebben op de uitoefening van de rechten van de betrokkenen.
verlies van controle over het gebruik van persoonsgegevens	L	H	<i>Blurring</i> , access management en beperkte opslagduur/retentieperiode hebben als doel dit risico te beperken.
verlies van vertrouwen	L	L	Zelfs als er een <i>data breach</i> plaatsvindt, softwarematige <i>blurring</i> zorgt voor de onherkenbaarmaking van de betrokkenen-passanten.

7.4.6.2 Wettelijke naleving Risico's

Overweeg of de volgende risico's zich voordoen en zo ja, geef commentaar:

- De waarschijnlijkheid dat het risico zich voordoet (laag / gemiddeld / hoog)
- De impact op de persoon (personen) als het risico zich voordoet (laag / gemiddeld / hoog)

Risico	Waarschijnlijkheid	Impact	Nadere toelichtingen
Niet-naleving van GDPR [(Vul de vragenlijst in de bijlage in en vat uw bevindingen samen.)]	L	L	Informatie kan gevonden worden op de website (transparency notice). In de mate dat er persoonsgegevens verzameld worden (hetzij van betrokkenen-passanten, hetzij van werknemers bij OVAM), gebeurt dit steeds rechtmatig, behoorlijk en transparant. De verwerking houdt zich aan de principes van doelbinding en minimale gegevensverwerking, doordat de gezichten en andere lichaamsdelen van mogelijke betrokkenen-passanten softwarematig <i>geblurred</i> worden en doordat er geen live-tracking plaatsvindt van de OVAM-werknemers die ter plaatse aanwezig zijn bij de zwerfvuilmonitoring. Betreffende de bewaring van de persoonsgegevens, hetzij de beelden, hetzij de GPS-coördinaten, vindt er zowel toegangscontrole als opslagbeperking plaats. Er worden in ieder geval passende technische en organisatorische maatregelen genomen ter bewaring van de integriteit en vertrouwelijkheid van de persoonsgegevens, maar ook ter naleving van de principes betreffende de gegevensbescherming door ontwerp en door standaardinstellingen.
Niet-naleving van sectorspecifieke wetgeving of normen.	NVT	NVT	
Mogelijke reputatieschade of verlies van vertrouwen bij het publiek	L	L	Zelfs als er een <i>data breach</i> plaatsvindt, softwarematige <i>blurring</i> zorgt voor de onherkenbaarmaking van de betrokkenen-passanten.

7.4.7 Identificeren en evalueren van maatregelen om het risico te verminderen

Zorg ervoor dat elk van de in punt 6 genoemde risico's in deze tabel wordt behandeld.

Potentiële oplossing	Welke risico's zou(den) deze actie aanpakken?	Resultaat	Kosten-baten
Het <i>blurren</i> van gezichten en andere lichaamsdelen van mogelijke betrokkenen-passanten.	Het risico van de verwerking van bijzondere categorieën van persoonsgegevens of potentiële gevoelige persoonsgegevens van mogelijke betrokkenen-passanten.	Het risico wordt verminderd door het nemen van passende TOM's.	Aanbevolen implementatie: Volledige <i>blurring</i> van betrokkenen-passanten is mogelijks technologisch en financieel niet haalbaar. Gezichts <i>blurring</i> van betrokkenen-passanten valt echter wel binnen de mogelijkheden.
Vermindering van de omvang van de verwerking	Risico van niet-naleving van het principe van minimale gegevensverwerking.	Het risico wordt verminderd door het nemen van passende TOM's.	Aanbevolen implementatie: Volledige <i>blurring</i> van betrokkenen-passanten is mogelijks technologisch en financieel niet haalbaar. Gezichts <i>blurring</i> van betrokkenen-passanten valt echter wel binnen de mogelijkheden.
Gebruik een andere technologie	/	/	/
Invoering van bewaartermijnen om informatie slechts zo lang te bewaren als nodig is	Risico van niet-naleving van het principe van opslagbeperking.	Het risico wordt verminderd door het instellen van bewaartermijnen waarna de verzamelde beelden en GPS-coördinaten worden verwijderd.	Aanbevolen implementatie: het bewaren van de beelden en GPS-coördinaten beperken tot de minimum vereiste opslagtermijn ter bereik van het verwerkingsdoel.
Veilige vernietiging van informatie die niet langer hoeft te worden bewaard	Risico van niet-geautoriseerde toegang.	Het risico wordt verminderd eens de niet langer nodige verzamelde gegevens zijn vernietigd.	Aanbevolen implementatie: het vernietigen van de beelden en GPS-coördinaten uiterlijk na (één maand?) opslag.
Tenuitvoerlegging van passende technologische veiligheidsmaatregelen	Risico van niet-geautoriseerde toegang, verlies van vertrouwelijkheid	Het risico wordt verminderd door het instellen van toegangscontrole, het versleutelen van de gegevens (encryptie), de pseudonimisatie/anonimisatie	Aanbevolen implementatie: Toegang tot de database houdende de beelden en GPS-coördinaten enkel mogelijk voor werknemers op basis van het 'need to know'-principe. Versleuteling van de

Potentiële oplossing	Welke risico's zou(den) deze actie aanpakken?	Resultaat	Kosten-baten
		van persoonsgegevens en andere TOM's.	gegevens. Anonimisatie/pseudonimisatie waar mogelijk.
Het personeel goed opleiden en bewust maken van de mogelijke risico's voor de privacy	Risico van niet-naleving van het principe van minimale gegevensverwerking.	Het risico zal gereduceerd worden eens de werknemers bewust worden gemaakt hoe de TOM's op een correcte wijze te implementeren.	Aanbevolen implementatie: wij raden aan om de werknemers in detail uit te leggen hoe de TOM's te volgen en hen ook te benadrukken deze te volgen.
Ervoor zorgen dat informatie waar mogelijk veilig geanonimiseerd wordt gepseudonimiseerd	Risico van verlies van vertrouwelijkheid	Het risico wordt verminderd door het anonimiseren/pseudonimiseren van alle persoonsgegevens, waar mogelijk.	Aanbevolen implementatie: Softwarematige <i>blurring</i> van het gezicht/alle lichaamsdelen van betrokkenen-passanten.
Geef begeleiding aan het personeel over hoe dat moet: - het nieuwe systeem te gebruiken, en -gegevens op de juiste wijze te delen	/	/	/
Het verzekeren van het nieuwe systeem stelt individuen in staat om: - gemakkelijker toegang te krijgen tot hun informatie, en -uitoefening van hun rechten	/	/	/
Het verzekeren van individuen: -zijn zich volledig bewust van de manier waarop hun informatie wordt gebruikt, en -kan contact met ons opnemen voor hulp wanneer dat nodig is	/	/	/

Potentiële oplossing	Welke risico's zou(den) deze actie aanpakken?	Resultaat	Kosten-baten
-kan zich in voorkomend geval terugtrekken			
Het selecteren van gegevensverwerkers die een grotere mate van veiligheid bieden	/	/	/
Het waarborgen van overeenkomsten met gegevensverwerkers om de informatie die namens ons wordt verwerkt te beschermen	/	/	/
Het garanderen van een overeenkomst voor het delen van gegevens maakt het duidelijk: -wat informatie zal worden gedeeld -hoe het zal worden gedeeld, en -met wie	/	/	/
[Voeg een andere oplossing toe die u hebt geïdentificeerd]	/	/	/
[Voeg een andere oplossing toe die u hebt geïdentificeerd]	/	/	/
[Voeg een andere oplossing toe die u hebt geïdentificeerd]	/	/	/
[Voeg een andere oplossing toe die u hebt geïdentificeerd]	/	/	/

7.4.8 Ondertekenen en vastleggen van DPIA-resultaten

7.4.8.1 [Samenvatting DPO advies]

[Als uw organisatie een DPO heeft, voeg dan een samenvatting van hun advies in, met inbegrip van de vraag of de verwerking kan worden voortgezet].

[Als het advies van de DPO is afgewezen, moet u uw redenen hiervoor uiteenzetten].

7.4.8.2 Samenvatting van de risico's voor de privacy en goedgekeurde oplossingen

Risico	Goedgekeurde oplossing	Goedgekeurd door
[Voeg een beschrijving van een privacyrisico in]	[[Beschrijf de oplossing die is goedgekeurd om dit risico te elimineren of te verminderen.] OF Residuele risico's aanvaard en goedgekeurd]	[Geef aan wie deze oplossing heeft goedgekeurd]
[Voeg een beschrijving van een privacyrisico in]	[[Beschrijf de oplossing die is goedgekeurd om dit risico te elimineren of te verminderen.] OF Residuele risico's aanvaard en goedgekeurd]	[Geef aan wie deze oplossing heeft goedgekeurd]
[Voeg een beschrijving van een privacyrisico in]	[[Beschrijf de oplossing die is goedgekeurd om dit risico te elimineren of te verminderen.] OF Residuele risico's aanvaard en goedgekeurd]	[Geef aan wie deze oplossing heeft goedgekeurd]

7.4.8.3 Belanghebbende consultatie

[Vat het resultaat van uw raadpleging van belanghebbenden samen. Als uw beslissing afwijkt van de standpunten of aanbevelingen van de belanghebbenden, licht dan uw redenen toe]

7.4.8.4 ICO consultatie

Geeft deze beoordeling aan dat de verwerking van het project een groot risico met zich meebrengt als er geen mitigatiemaatregelen worden genomen?	Ja
Zo ja, kunnen deze risico's dan met redelijke middelen worden beperkt in termen van de beschikbare technologieën en de kosten van de tenuitvoerlegging?	Ja [Indien dit niet het geval is, moet voorafgaand aan de verwerking overleg worden gepleegd met het Information Commissioner's Office (ICO).]
Als het nodig is om met de ICO te overleggen, is dit dan gedaan?	[Ja/Neen OF N.v.t] Zo ja, geef dan nadere informatie Nvt

7.4.8.5 Goedkeuring en aftekenen

[Ik OF wij] bevestigen dat [ik OF wij] deze DPIA hebben bekeken en [ben OF zijn] daar tevreden mee:

- 1.1.1 De DPO van [[bedrijf OF onderneming] is geraadpleegd en heeft over deze DPIA geadviseerd; en]

- 1.1.2 Het is [niet] noodzakelijk om met de ICO te overleggen.
- 1.1.3 Waar nodig zijn de belanghebbende partijen geraadpleegd en is er rekening gehouden met hun standpunten waar dat relevant is.
- 1.1.4 et voorgestelde project is in overeenstemming met de gegevensbeschermingsbeginselen van artikel 5 van GDPR.
- 1.1.5 Met betrekking tot de verwerking van persoonsgegevens is ten minste één van de wettelijke gronden in artikel 6 van GDPR van toepassing.
- 1.1.6 Met betrekking tot de verwerking van persoonsgegevens van bijzondere categorieën is ten minste één van de uitzonderingen in artikel 9 van GDPR ook van toepassing.
- 1.1.7 Alle relevante privacyrisico's en -oplossingen zijn opgenomen in het privacyrisicoregister van het bedrijf of de onderneming en er zijn regelingen getroffen om het register gedurende de looptijd van dit project regelmatig te controleren.
- 1.1.8 De oplossingen die in deze beoordeling worden genoemd, vormen een gerichte en evenredige reactie op de vastgestelde risico's voor de persoonlijke levenssfeer.

Naam/Namen	<i>[naam van de persoon of personen die de DPIA aftekenen en goedkeuren].</i>
Job Titel	<i>[Voeg de functiebenaming van de persoon of personen in die de DPIA aftekenen en goedkeuren].</i>
Datum	<i>[Datum invoegen]</i>

7.4.9 Integratie van de DPIA-resultaten in het algemene projectplan

7.4.9.1 High-level acties

Te ondernemen actie	Datum voltooiing van of frequentie	Verantwoordelijk voor actie
Integreer de DPIA-resultaten terug in het projectplan en werk de relevante projectmanagementpapieren/spreadsheets bij.	<i>[Datum invoegen]</i>	<i>[Naam invoegen]</i>
Voer de goedgekeurde oplossingen voor privacyrisico's uit.	<i>[Datum invoegen]</i>	<i>[Naam invoegen]</i>
Herzie het DPIA en het projectplan met regelmatige tussenpozen.	<i>[Voeg frequentie in, bijv. maandelijks]</i>	<i>[Naam invoegen]</i>
Update het DPIA en het projectplan indien nodig en/of met regelmatige tussenpozen.	<i>[Voeg frequentie in, bijv. maandelijks]</i>	<i>[Naam invoegen]</i>
Overleg voeren.	<i>[Datum invoegen]</i>	<i>[Naam invoegen]</i>

7.4.9.2 Specifieke acties

Te ondernemen actie	Datum van voltooiing	Verantwoordelijk voor actie
[Voeg specifieke actiepunten toe, met name uit uw 'Samenvatting van de risico's en oplossingen op het gebied van privacy'..]	[Datum invoegen]	[Naam invoegen]
[Voeg specifieke actiepunten toe, met name uit uw 'Samenvatting van de risico's en oplossingen op het gebied van privacy'..]	[Datum invoegen]	[Naam invoegen]
[Voeg specifieke actiepunten toe, met name uit uw 'Samenvatting van de risico's en oplossingen op het gebied van privacy'..]	[Datum invoegen]	[Naam invoegen]
[Voeg specifieke actiepunten toe, met name uit uw 'Samenvatting van de risico's en oplossingen op het gebied van privacy'..]	[Datum invoegen]	[Naam invoegen]
[Voeg specifieke actiepunten toe, met name uit uw 'Samenvatting van de risico's en oplossingen op het gebied van privacy'..]	[Datum invoegen]	[Naam invoegen]

BIBLIOGRAFIE

- Chung, A., Kim, D. Y., Kwok, E., Ryan, M., Tan, E., & Gamadia, R. (2018, October). Cloud Computed Machine Learning Based Real-Time Litter Detection using Micro-UAV Surveillance. In 2018 IEEE MIT Undergraduate Research Technology Conference (URTC) (pp. 1-4). IEEE.
- Hamza, V. S. (2020). Testing Multi-Frequency Low-Cost GNSS Receivers for Geodetic Monitoring Purposes.
- Kulkarni, H. N., & Raman, N. K. S. (2019). Waste Object Detection and Classification.
- Proença, P. F. (2020). TACO: Trash Annotations in Context for Litter Detection. arXiv.
- Rad, M. S., von Kaenel, A., Droux, A., Tieche, F., Ouerhani, N., Ekenel, H. K., & Thiran, J. P. (2017, July). A computer vision system to localize and classify wastes on the streets. In International Conference on computer vision systems (pp. 195-204). Springer, Cham.
- Seredkin, A. V., Tokarev, M. P., Plohih, I. A., Gobyrov, O. A., & Markovich, D. M. (2019, November). Development of a method of detection and classification of waste objects on a conveyor for a robotic sorting system. In Journal of Physics: Conference Series (Vol. 1359, No. 1, p. 012127). IOP Publishing.
- Soviany, P., & Ionescu, R. T. (2018, September). Optimizing the trade-off between single-stage and two-stage deep object detectors using image difficulty prediction. In 2018 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNAS) (pp. 209-214). IEEE.