



Eindrapport – LiDAR BIM vergelijking

Gemeente Den Haag

2022-01-18

Versie: 1.0

Confidential © 2022 CGI Inc.

Inhoudsopgave

[1 Introductie 3](#_Toc93399374)

[1.1 Achtergrond 3](#_Toc93399375)

[1.2 Onderzoeksvraag 3](#_Toc93399376)

[1.3 Overzicht oplossing 3](#_Toc93399377)

[1.4 Gebruik applicatie 4](#_Toc93399378)

[2 Methoden 6](#_Toc93399379)

[2.1 Input data 6](#_Toc93399380)

[2.2 Verwerkingsmodule 7](#_Toc93399381)

[2.3 Uitvoer 9](#_Toc93399382)

[3 Resultaten 10](#_Toc93399383)

[3.1 Verwerkingsmodule 10](#_Toc93399384)

[3.2 Use cases 13](#_Toc93399385)

[3.3 Resultaten API 16](#_Toc93399386)

[3.4 Vergelijking van puntenwolken verwerking met BIM 16](#_Toc93399387)

[4 Conclusie 17](#_Toc93399388)

[5 Lessons Learned 18](#_Toc93399389)

[5.1 Puntenwolken 18](#_Toc93399390)

[5.2 BIM 18](#_Toc93399391)

[5.3 Verwerking 18](#_Toc93399392)

[6 Aanbevelingen 21](#_Toc93399393)

[6.1 Methode 21](#_Toc93399394)

[6.2 Implementatie 21](#_Toc93399395)

[6.3 Aansluiting op de eindgebruikers 21](#_Toc93399396)

[Appendix A Vergelijking ruimtes puntenwolkenverwerking met BIM 22](#_Toc93399397)

# Introductie

## Achtergrond

De gemeente Den Haag is één van de drie Nederlandse gemeenten die deelneemt aan het subsidieprogramma “Proefopstelling Totaal Driedimensionaal” (T3D). Samen met de Vereniging van Nederlandse Gemeenten (VNG) richt dit programma zich op de effectieve verwerking en het gebruik van verschillende 3D-gegevens van de stedelijke omgeving uit de Nederlandse vastgoedwereld.

Gemeente Den Haag focust zich op het eerste deel van de “data pijplijn”, namelijk de extractie en voorverwerking van de bestaande 3D gegevens. Om deze gegevens in te winnen en te verwerken heeft de gemeente Den Haag CGI gevraagd een Proof of Concept (PoC) uit te voeren. In het kader van deze PoC streeft de gemeente ernaar om drie soorten gegevens op te nemen: BIM-modellen, puntenwolken en (potentieel) digitale beelden. Deze gegevens moeten vervolgens worden voorbewerkt in de regiekamer waarna de data kan worden doorgegeven aan het registratie component en uiteindelijk gebruikt kan worden.

CGI richt zich in deze PoC op het vergelijken van puntenwolken met BIM-modellen.

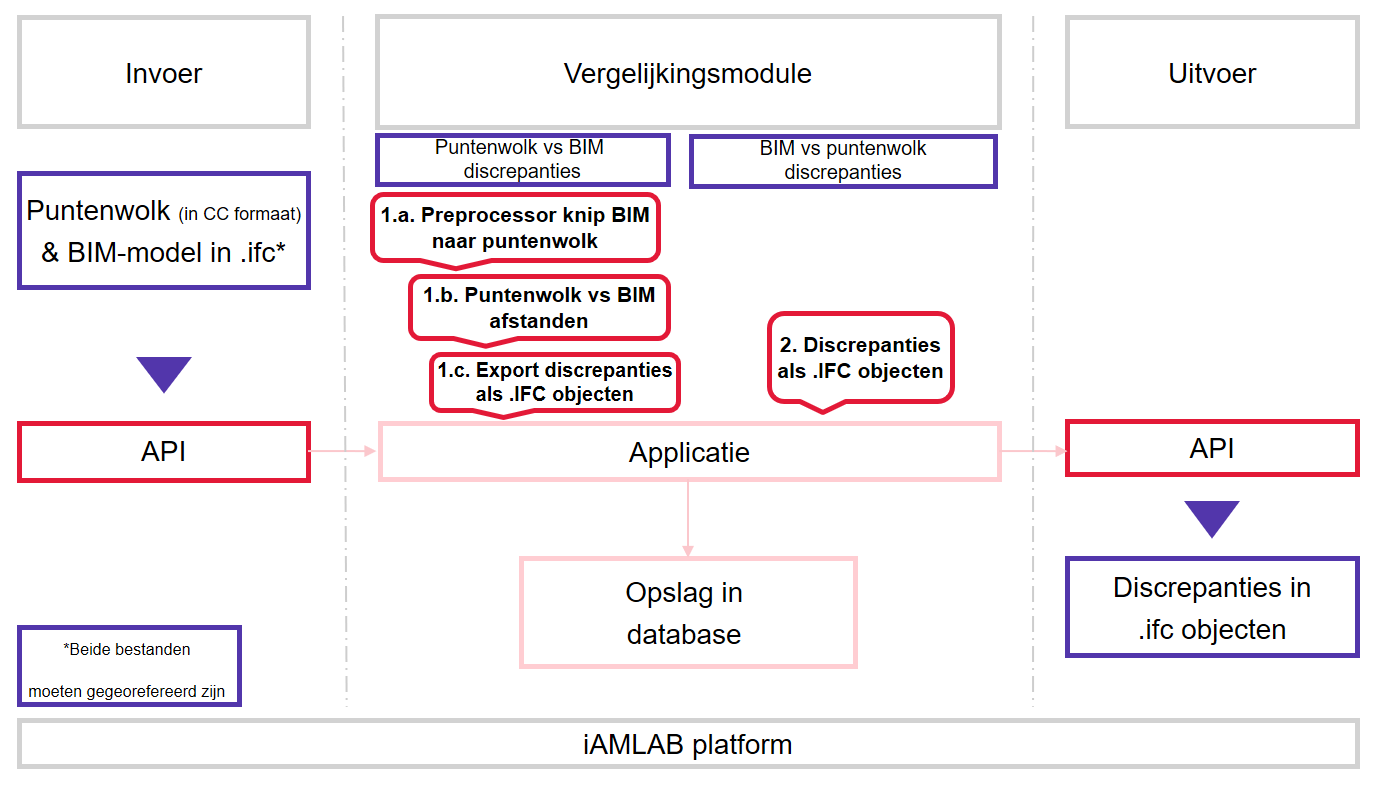
## Onderzoeksvraag

De scope van deze PoC is het maken van een vergelijkingsmodule voor puntenwolken en BIM-modellen die afkomstig zijn uit het vergunningsverleningsproces. Het doel van deze module is het effectief kunnen verwerken van de bestanden en het inzichtelijk maken van de verschillen voor de vergunningsverlener en de objectenregistrator. Dit leidt tot de volgende onderzoeksvraag:

*“Is het mogelijk om automatisch verschillen te detecteren tussen een ingewonnen puntenwolk en een BIM-model ten behoeve van het taxatieproces van de WOZ en/of het beoordelen van een bouwvergunning?”*

## Overzicht oplossing

In de aanbieding heeft CGI een overzicht van de beoogde oplossing getoond. In Figuur 1 is de oplossing gevisualiseerd zoals deze daadwerkelijk is toegepast tijdens de PoC door het iAMLAB team. Hierin zijn de verschillende stappen van de applicatie zichtbaar. In hoofdstuk 2 worden de verschillende stappen in meer detail besproken.



Figuur 1: Een overzicht van de oplossing.

De reden dat de vergelijkingsmodule in twee delen is opgeknipt is omdat we discrepanties willen berekenen van wat zowel in puntenwolk is maar niet in de BIM en andersom. In de onderstaande tabel is beschreven hoe we daarmee tot antwoorden kunnen komen op de verschillende use cases. De resultaten van de use cases zijn te vinden in hoofdstuk 3.2.

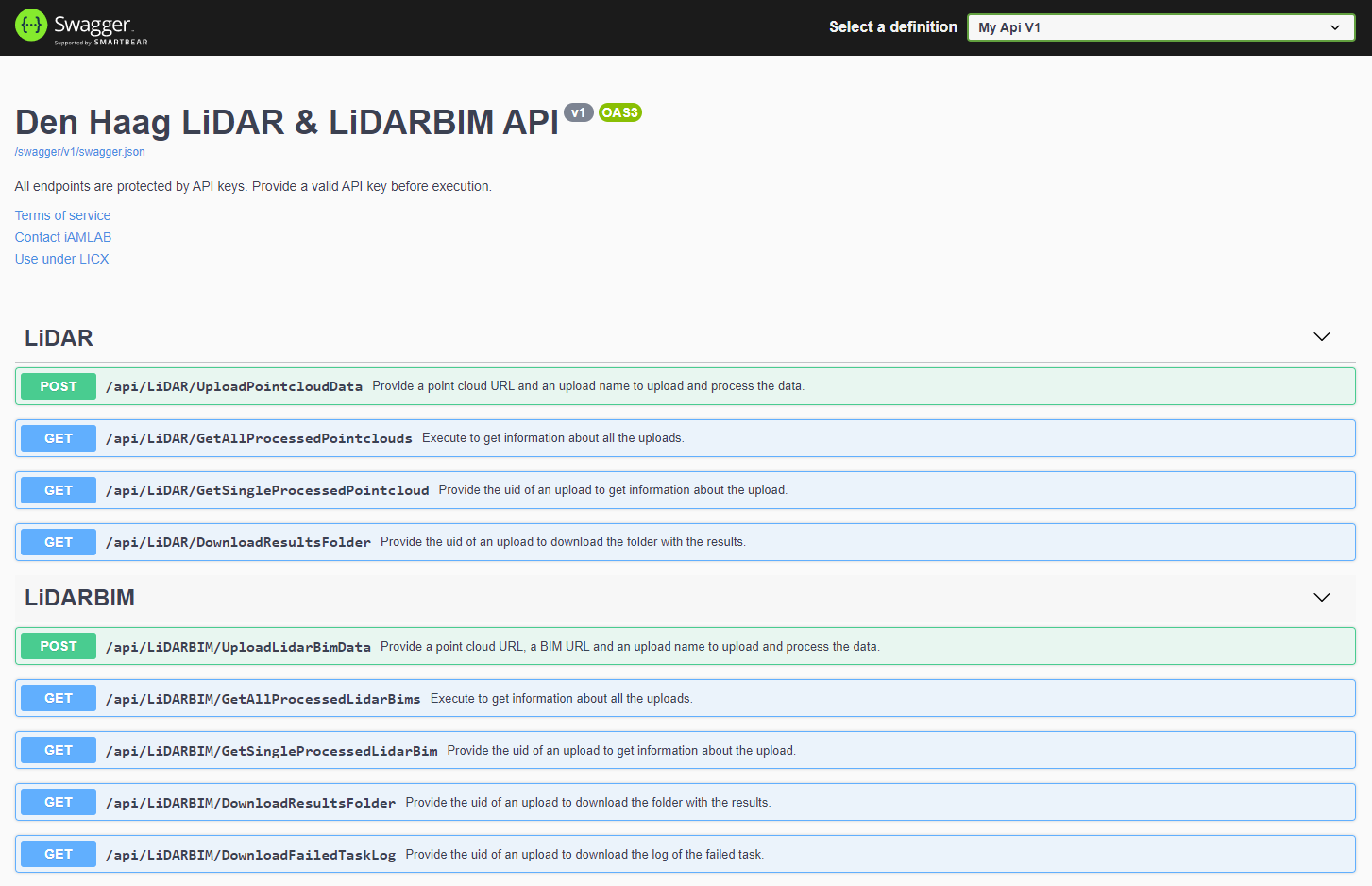
|  |  |  |
| --- | --- | --- |
|  | **Puntenwolk -> BIM** | **BIM -> Puntenwolk** |
| Wat detecteer je? | Aanbouw. Toevoegingen die niet in het BIM-model staan. | Verwijderingen. Objecten die niet in de puntenwolk staan |
| Wat mis je? | Verwijderingen. Objecten die niet in de puntenwolk staan. | Aanbouw. Toevoegingen die niet in het BIM-model staan. |

## Gebruik applicatie

De ontwikkelde code is in een .zip bestand aan de gemeente Den Haag geleverd. Hier bevinden zich ook de Readme.md bestanden waarin wordt uitgelegd hoe de code moet worden gebruikt. Op deze manier kan de gemeente zelf de pipeline opzetten en aanpassingen doorvoeren aan de code.

CGI heeft daarnaast verschillende APIs ontwikkeld waarmee op een laagdrempelige en gebruiksvriendelijke manier de pipeline automatisch wordt gestart en de resultaten kunnen worden opgevraagd. In de Post-request kunnen de URL’s van een puntenwolk bestand en een BIM-bestand ingevoerd worden. De puntenwolk kan in elk formaat waarmee CloudCompare kan werken. Voor de volledige lijst van mogelijke invoermogelijkheden kan de CloudCompareWiki*[[1]](#footnote-2)* bekeken worden. Het BIM-bestand moet een IFC-bestand zijn.

Met behulp van de Get-requests kunnen de logs en de resultaten opgehaald worden. In de resultaten map staan zowel de tussenresultaten als de eindresultaten. In hoofdstuk 3.3 worden de output bestanden in detail beschreven. Alle endpoints worden om veiligheidsredenen beschermd door API-keys, die aan Den Haag zijn verstrekt. De APIs worden in figuur 2 weergegeven in de vorm van Swagger UI, waar ze kunnen worden getest. Bovendien worden de APIs van het andere project (LiDAR) ook geleverd samen met de APIs van dit project (LiDAR-BIM). De URL van deze API-collectie is met Den Haag gedeeld*[[2]](#footnote-3).*



Figuur 2: De APIs in Swagger UI.

### API gebruik in detail

De APIs die in contact staan met de door CGI ontwikkelde pipeline bieden een interface tot de in- en uitvoer van deze informatie. Door correcte gegevens in te voeren (puntenwolk en IFC-bestanden) kan de analyse worden uitgevoerd.

In detail zijn er 5 endpoints voor dit project, 1 Post-request en 4 Get-requests. In de Post-request (/api/LiDARBIM/UploadLidarBimData) kunnen de URL's van de puntenwolk en het BIM-bestand worden opgegeven, samen met een naam voor de upload. Wanneer deze wordt uitgevoerd, worden de bestanden geüpload en verwerkt. Het duurt ongeveer 30 minuten voordat alle stappen zijn voltooid, afhankelijk van de grootte van de inputbestanden. De eerste Get-request (/api/LiDARBIM/GetAllProcessedLidarBims) kan worden gebruikt om de informatie over alle uploads op te halen (name, uid, status, upload date), en in de tweede Get-request (/api/LiDARBIM/GetSingleProcessedLidarBim) kan de uid van de upload worden opgegeven om de informatie over alleen deze specifieke upload te ontvangen. In de derde Get-request (/api/LiDARBIM/DownloadResultsFolder) kan de uid van een upload worden opgegeven om het .zip-bestand van de resultaten te downloaden. In het geval dat de upload is mislukt, kan de laatste Get-request (/api/LiDARBIM/DownloadFailedTaskLog) worden gebruikt om de uid van de upload op te geven en het .txt-bestand van de log van de mislukte taak te downloaden.

# Methoden

## Input data

### Puntenwolken

Voor de ontwikkeling van de puntenwolken en BIM-vergelijkingsmodule is gebruik gemaakt van de puntenwolken van Woonstad. Specifiek zijn hier drie appartementencomplexen uitgekozen die zijn gebruikt om de algoritmes op te valideren. Dit zijn de appartementencomplexen op de adressen Allard Piersonstraat 3B, Aelbrechtskade 6B en Aelbrechtsplein 2C.

### BIM

Voor de vergelijking met de drie bovengenoemde puntenwolken is gebruik gemaakt van de BIM (als IFC-bestand) genaamd Huur Alp Aelbrecht die het gehele gebouw omvat waarin appartementen zitten. Dit IFC-object bevat de objecten waarvan ook puntenwolken van Woonstad beschikbaar zijn. Het bestand is aangeleverd door de Gemeente Den Haag en is zichtbaar in Figuur 3 rechts.

### Georeferentie en uitlijning

Een eis van de input data is dat de beide datasets gegeorefereerd zijn en daarnaast uitgelijnd zijn aan elkaar. Dit betekent dat zowel puntenwolk als BIM hetzelfde coördinatensysteem moeten hebben. Aangezien dit voor de puntenwolken van Woonstad niet het geval was, is hier de handmatige stap beschreven die in dit project hiervoor zijn gedaan. De drie appartementencomplexen van paragraaf 2.1.1 zijn op deze manier bewerkt.



Figuur 3: Het gebouw in Google Maps (links) en de BIM in de viewer BIMvision (rechts).

Met behulp van Google Maps, de BIM en de puntenwolken is gekeken waar de adressen van de puntenwolken zich exact bevinden in de BIM. Daarna zijn deze bestanden in CloudCompare ingeladen en de volgende stappen uitgevoerd:

1. Selecteer zowel het BIM-bestand (als OBJ-bestand, anders kan CloudCompare het niet openen) als de puntenwolk.
2. Klik op *Tools > Registration > Align (point pairs picking)*.
3. Selecteer de puntenwolk en klik OK.
4. Op het venster dat verschijnt: laat de *show to ‘align’ entities* box aangevinkt en de *show to ‘reference’ entities* box leeg. Kies nu ten minste 3 punten van de puntenwolk.   
   Tip: Kies punten die in zowel BIM als puntenwolk zichtbaar en herkenbaar zijn. Hoe meer punten gebruikt worden en ook hoe beter verdeeld in de 3D space, hoe nauwkeuriger de resultaten. Fout gemarkeerde punten kunnen tussentijds altijd weer worden verwijderd.
5. Vink nu de *show to ‘align’ entities* box uit en vink de *show to ‘reference’ entities* box aan. Kies nu de respectievelijke punten op de BIM. Het aantal en de volgorde van de punten moet hetzelfde zijn als de vorige stap!
6. Na het selecteren van de puntencombinaties, klik op *align* en dan op ü. Mochten de resultaten nog niet goed genoeg zijn, kan het stappenplan nog een keer worden herhaald.
7. Exporteer de uitgelijnde puntenwolk nu als een nieuw puntenwolk bestand.

## Verwerkingsmodule

De verwerkingsmodule voert de stappen zoals omschreven in Figuur 1 automatisch achter elkaar uit. De verschillende stappen worden in detail in deze paragraaf beschreven. De vergelijking is opgesplitst in twee verschillende componenten. De vergelijking kan namelijk worden gedaan vanuit twee oogpunten: BIM als waarheid of Puntenwolk als waarheid. Beide “richtingen” worden hieronder verder toegelicht.

### Puntenwolk versus BIM discrepanties

#### Omzetten IFC naar OBJ

Om de puntenwolk met de BIM te kunnen vergelijken wordt het formaat van de BIM omgezet van IFC naar OBJ. De algoritmes die we gebruiken van CloudCompare kunnen wel op OBJ-formaat maar niet op IFC worden toegepast. De tool die de conversie uitvoert is IfcConvert, een tool van de open source IFC toolkit IfcOpenShell[[3]](#footnote-4). Deze tool is gekozen omdat het de originele coördinaten van de BIM bewaart tijdens de conversie en omdat het naast het OBJ-bestand ook een MTL-bestand maakt. Dit MTL-bestand bevat de “texture” eigenschappen van het OBJ mesh bestand. Als de gebruiker het OBJ-bestand samen met de textures in CloudCompare wil laden om het te bekijken moeten in de *Properties* tab de *Visible* en *Materials/texture* box aangevinkt en de *Normals* box uitgevinkt zijn.

#### Splitsen BIM per puntenwolk

Om de vergelijking tussen puntenwolk en BIM goed te laten verlopen, moet het BIM-bestand opgeknipt worden naar de bounding box van de puntenwolk. Hierdoor worden in het vervolg alleen de afstanden uitgerekend over hetzelfde gebied van interesse. De puntenwolk wordt voor de rest van de verwerking ook gesubsampled naar 0.01 meter zodat deze dataset niet onnodig hoge dataresolutie bevat waar we in de verdere verwerking niet veel aan hebben. Ook wordt de extensie omgezet naar een .las bestand in het geval dat deze dat nog niet was. Dan wordt de bounding box van de puntenwolk automatisch uitgelezen door middel van laspy[[4]](#footnote-5). Ten slotte wordt deze “extent” toegepast op het OBJ-bestand van de BIM en als een sub selectie hiervan opgeslagen met de *Crop[[5]](#footnote-6)* tool van CloudCompare.

#### Afstanden puntenwolk en BIM

Om de discrepanties van de puntenwolk t.o.v. de BIM te vinden, wordt eerst de afstand tussen de puntenwolk en de BIM uitgerekend. Dit gebeurt met de *Cloud-to-Mesh distance[[6]](#footnote-7)* tool van CloudCompare. Deze tool rekent voor elk punt in de puntenwolk de afstand uit tot de dichtstbijzijnde triangle van de mesh. De afstanden worden opgeslagen in een nieuw scalar veld van de puntenwolk, genaamd *C2M signed distances*. Dit scalar field wordt gebruikt om visueel de afstanden weer te geven. De kleurschaal kan namelijk worden gebruikt om punten met kleine en grote afstanden te visualiseren. Ook wordt dit scalar field gebruikt om een filtering toe te passen op de relevante verschillen. Het resultaat wordt geëxporteerd voor afstanden tussen de 0.2 en 10 meter. Dit zorgt ervoor dat hele kleine afstanden (onnauwkeurigheden) en punten ver buiten de mesh niet worden meegenomen.

#### Vlakkendetectie van Cloud-to-Mesh afstanden

Om van de discrepanties in de puntenwolk naar IFC-objecten te gaan worden eerst vlakken geëxtraheerd. Dit gebeurt door middel van Plane Segmentation[[7]](#footnote-8), die gebruik maakt van RANSAC en vervolgens DBScan Clustering[[8]](#footnote-9). Van deze vlakken wordt ook bepaald of ze horizontaal of verticaal georiënteerd zijn en dus een vloer/plafond object of een muur object zijn. Dit resulteert in geclusterde puntenwolkjes van de vlakken. Door middel van de convex hull[[9]](#footnote-10) worden deze tot een mesh gemaakt. De mesh wordt nog gesimplificeerd om te voorkomen dat er een complexe geometrie met veel vertices uitkomt die niet gemakkelijk naar een IFC-bestand kan worden geconverteerd. Al deze resulterende meshes worden vervolgens in één gezamenlijk mesh samengevoegd en als OBJ-bestand uitgeschreven.

#### Export als .ifc-bestand

Als laatste stap worden door middel van FreeCAD[[10]](#footnote-11) de OBJ-vlakken omgezet naar een IFC-bestand. FreeCAD is een open-source “parametric 3D modeler” die vanuit Python kan worden uitgevoerd in plaats van de GUI te gebruiken. Met FreeCAD via Python wordt de mesh (OBJ) geconverteerd naar geometrie en vervolgens geëxporteerd als IFC-bestand.

Het resultaat is uiteindelijk de vlakken uit de puntenwolk waarvan de groep punten gezamenlijk een grote discrepantie hebben tot het BIM-model. Dit kunnen dus vlakken zijn als (openstaande) deuren of tafeloppervlakten. Deze vlakken zijn geëxporteerd als lege IFC-objecten.

### BIM versus puntenwolk discrepanties

Voor de vergelijking tussen BIM en puntenwolk zijn minder stappen nodig dan andersom. Dit komt omdat we hier starten vanuit de objecten die al in de BIM staan. Het idee is om voor alle punten in de puntenwolk te controleren of deze in een object vallen van het IFC model. Het checken hiervan wordt geometrisch gedaan en is triviaal. De uitdaging van dit component is efficiëntie. Ondanks dat het een triviale check is voor een algoritme om te controleren of een punt in een object valt, bevat de puntenwolk veel punten. De module vermindert de zoekruimte in het IFC-object op een aantal manieren.

Ook voor dit component bieden we de puntenwolk gesubsampled aan. Dit vermindert het aantal punten dat wordt gecheckt, net zoals in paragraaf 2.1.2.2. De puntenwolk wordt in deze stap gesubsampled op 0.1 meter.

Vervolgens checken we grofweg of de bounding box van de BIM en de puntenwolk wel overlappen. Het gehele BIM-model wordt gebruikt (dus niet gesplitst zoals in 2.1.2.2). Voor ieder deelobject van de BIM wordt deze bounding box controle uitgevoerd. De delen van de BIM die compleet buiten de puntenwolk vallen worden verwijderd uit het BIM-bestand. De geometrieën van het BIM-component worden intact gelaten. Er wordt bijvoorbeeld geen deel van een muur verwijderd.

Daarna maken we van elke object in de BIM een mesh, en daarvan checken we of deze wel overlapt met de mesh van de puntenwolk. Het kan voorkomen dat de bounding box van het BIM-deelobject wel overlapt met de puntenwolk, maar de mesh-vorm niet. Als er geen overlap is, wordt dit object uit het BIM-bestand gehaald.

Tenslotte wordt gecheckt van elk object van de BIM als mesh, of daarin minimaal één punt van de puntenwolk aanwezig is. Als dit het geval is, wordt het object uit het BIM-bestand gehaald. Als dit dus niet het geval is, blijft het object over. Hierdoor blijven er objecten over die níet in de puntenwolk opgenomen zijn.

## Uitvoer

De uitvoer van de applicatie bestaat uit de IFC-objecten die een discrepantie hebben vertoond. Deze zijn dus in twee manieren berekend, namelijk vanuit de puntenwolk of vanuit het BIM-model. De discrepanties worden geretourneerd als IFC-objecten zodat deze de informatie van de binnenkant van gebouwen op een correct detailniveau weergeeft. De IFC-objecten vanuit het BIM-model zijn de originele objecten zoals in de BIM beschreven maar dan geselecteerd op waar discrepanties met de puntenwolk zijn gevonden. De IFC-objecten vanuit de puntenwolk analyse zijn geconstrueerde IFC-objecten die vanuit vlakken naar meshes zijn opgebouwd. Deze bevatten dus ook geen BIM-informatie. Ook de ruwe discrepantie puntenwolk is beschikbaar als output om eventueel de vlakken te kunnen interpreteren waar deze vandaan komen.

# Resultaten

In dit hoofdstuk worden de resultaten van de verschillende stappen getoond zoals omschreven in de methode.

## Verwerkingsmodule

In deze paragraaf worden de verschillende resultaten van de componenten van de verwerkingsmodule weergeven.

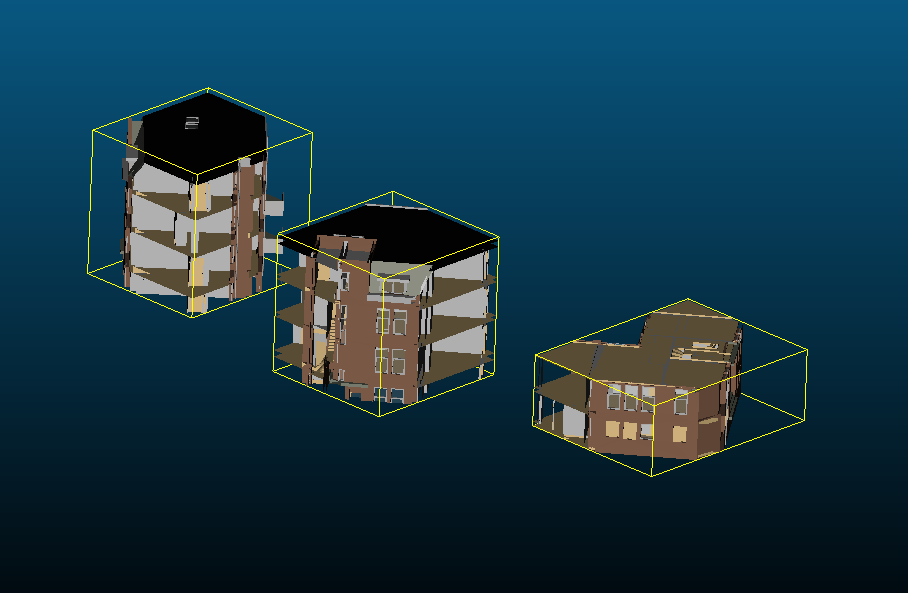
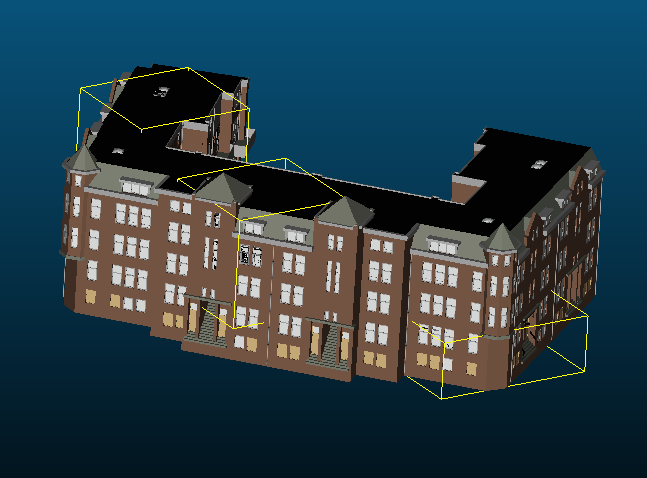
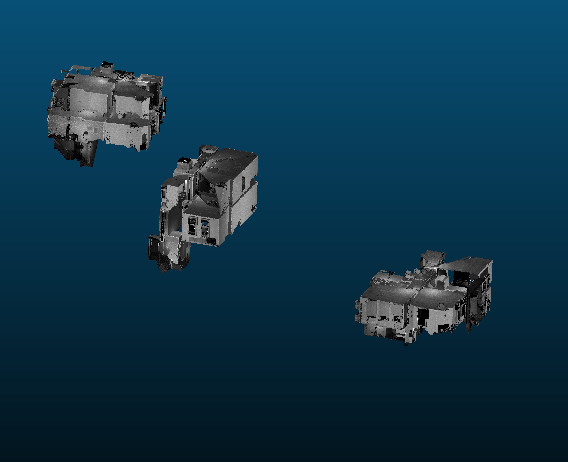
### Puntenwolk vs BIM discrepanties

#### Omzetten IFC naar OBJ

Het resultaat van de omzetting is het BIM-model als een OBJ-bestand dat CloudCompare kan lezen. Als voorbeeld is dit zichtbaar als het middelste plaatje van Figuur 4.

#### Splitsen BIM per puntenwolk

Na de uitlijning van de puntenwolk hebben deze een referentie ten opzichte van elkaar die ook klopt met de locatie in de BIM. Dit is zichtbaar op het linker en middelste plaatje van Figuur 4. Tenslotte is aan de rechterkant te zien dat het BIM-model als OBJ-bestand in drie verschillende subselecties is geknipt, behorend bij de bounding box van de puntenwolk.

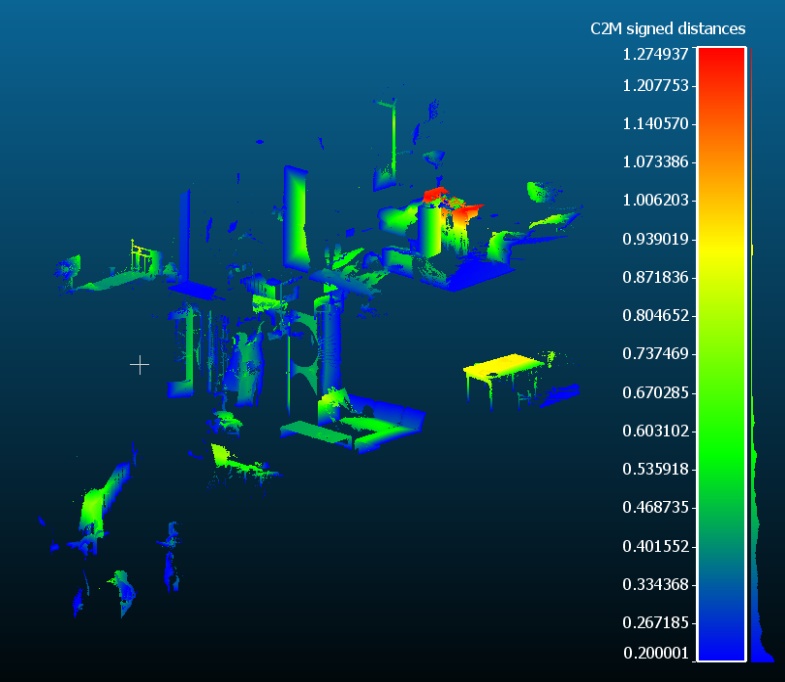
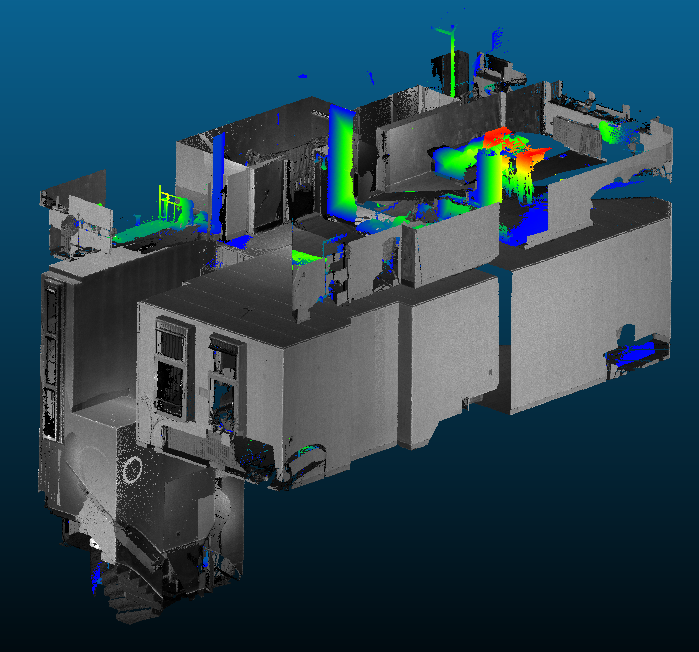


Figuur 4: Resultaten van splitsen BIM per puntenwolk.

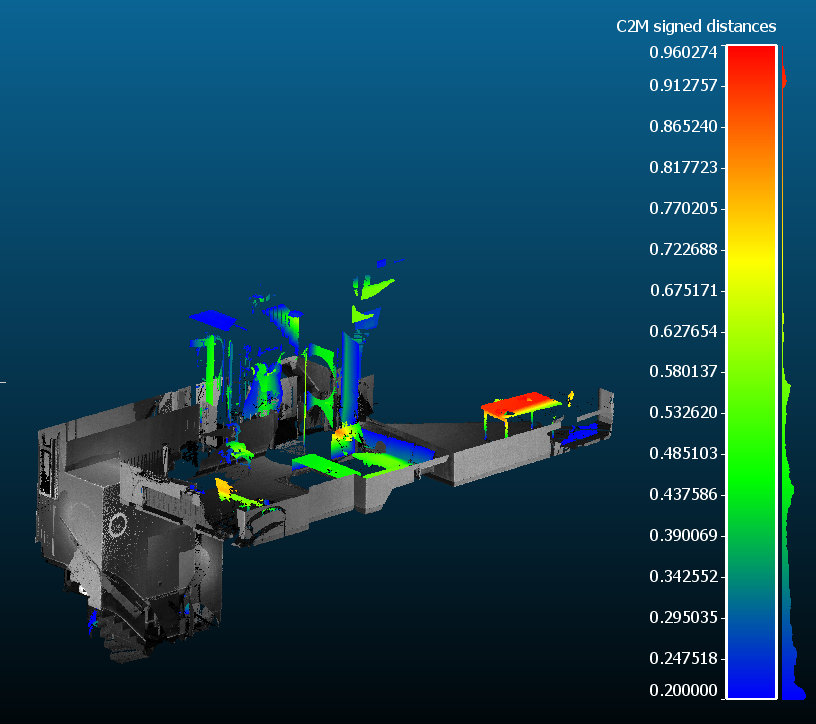
#### Afstanden puntenwolk en BIM

De volgende visualisaties in dit hoofdstuk zijn gemaakt met adres Aelbrechtsplein 2c.

In Figuur 5 is door de kleurenvisualisatie van het scalar field C2M signed distances, dat uitgerekend is in deze stap, goed zichtbaar waar de grootste afstanden liggen van de puntenwolk t.o.v. de BIM mesh. Dat is bijvoorbeeld een wasrek op de eerste verdieping, maar ook andere meubels zoals tafels en banken worden zichtbaar. Alle verschillen met een afstand kleiner dan 0.2 meter zijn uitgefilterd om te zorgen dat alleen relevante verschillen en niet onnodig veel ruis zichtbaar is. Dit zijn dus de punten die niet voorkomen in de BIM mesh en een afstand hebben van minimaal 0.2 meter.



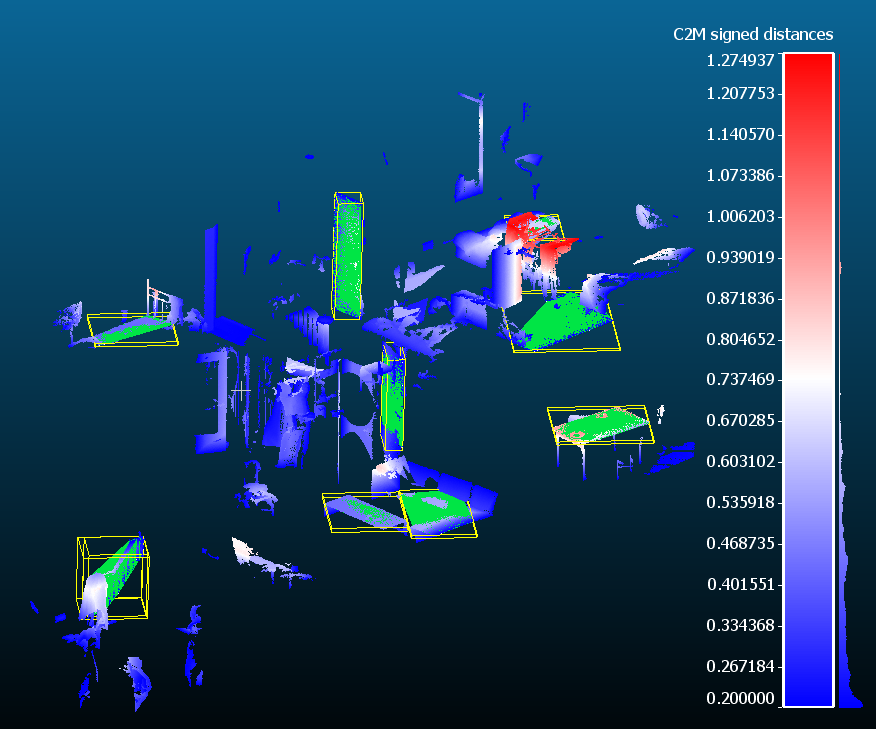
Figuur 5: Afstanden tussen puntenwolk en BIM. Rechts ter referentie ook de originele puntenwolk in grijs met de bovenste verdieping weergegeven.



Figuur 6: Afstanden tussen puntenwolk en BIM. Ter referentie ook een doorsnede van de originele puntenwolk in grijs met de onderste verdieping weergegeven.

#### Vlakkendetectie van Cloud-to-Mesh afstanden

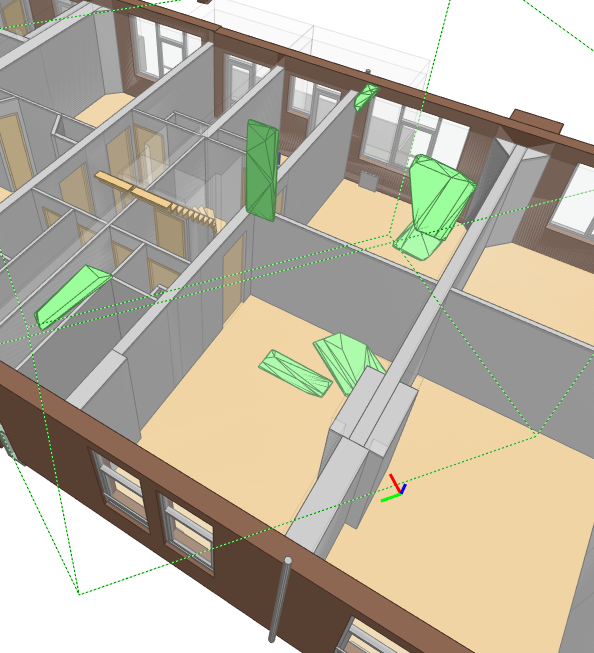
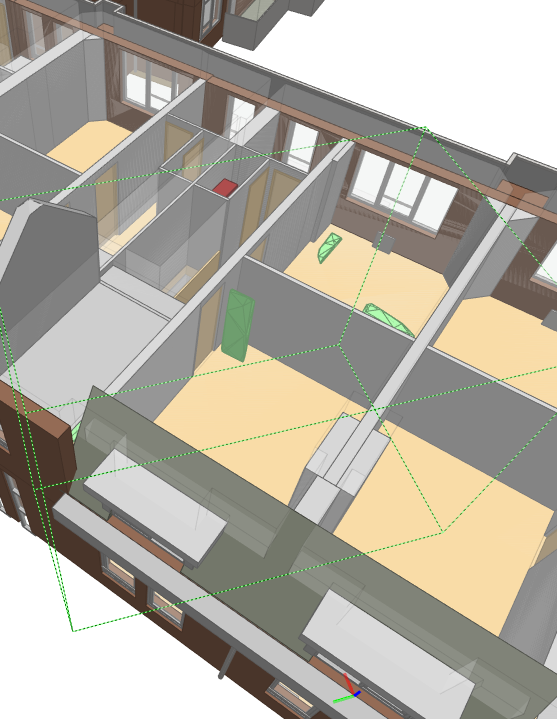
Omdat de afstanden tussen puntenwolk en BIM slechts visueel inzicht geeft en er ook veel punten zijn die geen interessante informatie bevatten, detecteren we vlakken uit deze afstanden puntenwolk. Deze vlakken zijn potentieel interessante discrepanties die we willen kunnen visualiseren samen met de BIM. Het resultaat van de vlakkendetectie is verschillende samengevoegde meshes in een OBJ-bestand. In Figuur 7 is zichtbaar in groen welke vlakken gedetecteerd worden uit de puntenwolk. Hierin zijn de tafels zichtbaar, de openstaande deuren, de reling van de trap, de zitting van de bank en ook een vlak op de grond.



Figuur 7: Vlakkendetectie van puntenwolk tot BIM afstanden. De discrepanties zijn nu gevisualiseerd in blauw t/m rood en de vlakken in groen.

#### Export als .ifc bestand

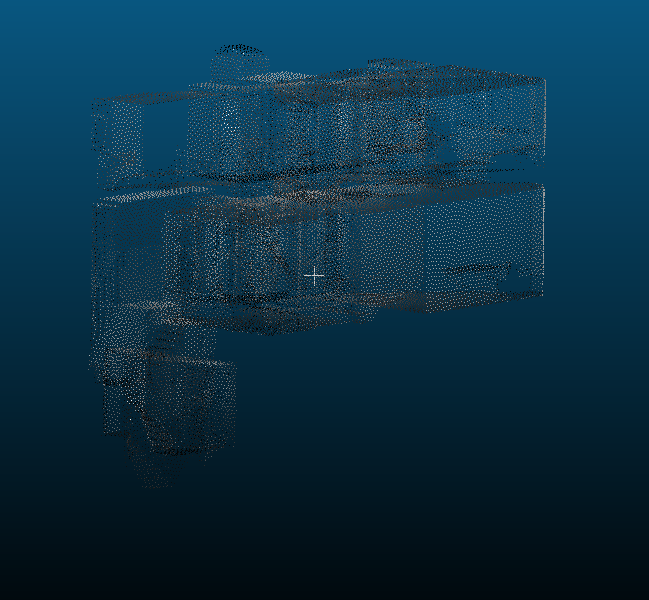
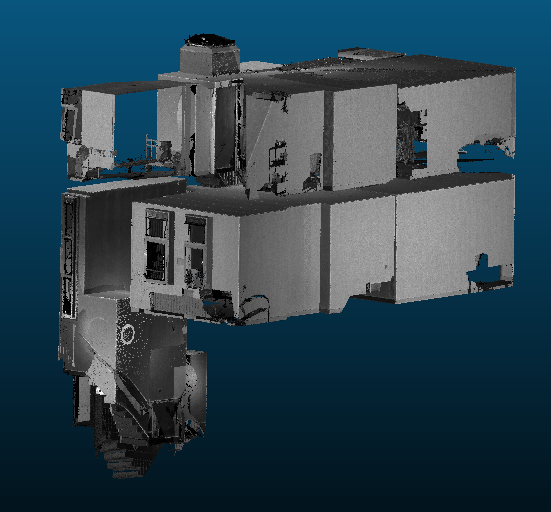
Omdat we graag de resultaten van de discrepanties samen met de BIM in een IFC-viewer kunnen laten zien worden de vlakken uit de vorige stap omgezet van OBJ-bestand naar IFC. In Figuur 8 is zichtbaar hoe dit eruit ziet gezamenlijk met de BIM. Nu we weten dat dit de deuren, tafels en bank waren kunnen we deze in de ruimtes van de BIM plaatsen en zien we dat de georeferentie hier ook in klopt.



Figuur 8: Vlakken als IFC-objecten. Links de bovenste verdieping, rechts de onderste verdieping met alle objecten in groen.

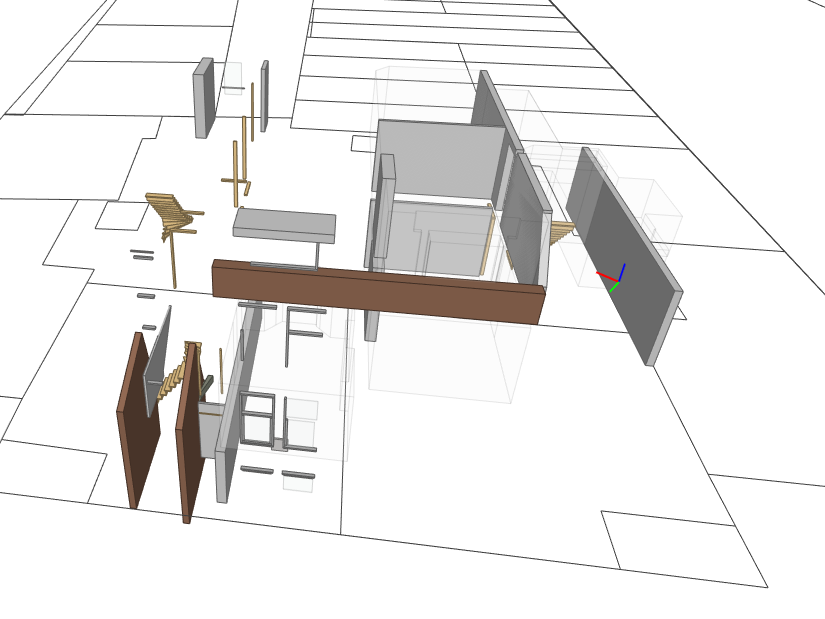
### BIM vs puntenwolk discrepanties

In de discrepanties van BIM naar puntenwolk hebben we een stuk minder tussenresultaten die hier visueel worden uitgelicht. In deze analyse zitten natuurlijk wel meerdere tussenstappen tot het eindresultaat. De belangrijkste is dat de puntenwolk hier nog verder wordt gesubsampled om het proces te versnellen. Nu duurt de analyse al circa 30 min en dat zou met de originele puntenwolk nog veel langer zijn. In Figuur 9 is het resultaat van deze subsampling met 0.1 meter distance te zien.



Figuur 9: Links de originele puntenwolk en rechts de gesubsamplede puntenwolk met 0.1 meter distance

Vervolgens zijn er meerdere stappen waarbij voor elk BIM element wordt gecheckt of deze relevant is. Alles wat niet overlapt met de puntenwolk wordt al weggegooid en vervolgens ook elk element waar zich punten in de puntenwolk bevinden. De elementen die overblijven zijn zichtbaar in Figuur 10. Hier is te zien dat er aardig wat elementen overblijven die potentieel discrepanties zijn.



Figuur 10: BIM discrepanties t.o.v. puntenwolk. Resterende IFC-objecten zijn de vlakken waar geen punten zijn gedetecteerd in de puntenwolk.

## Use cases

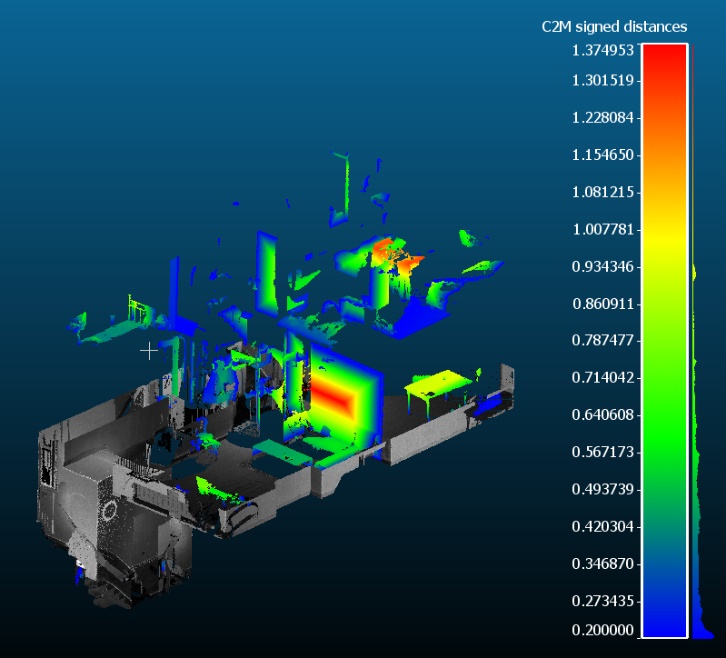
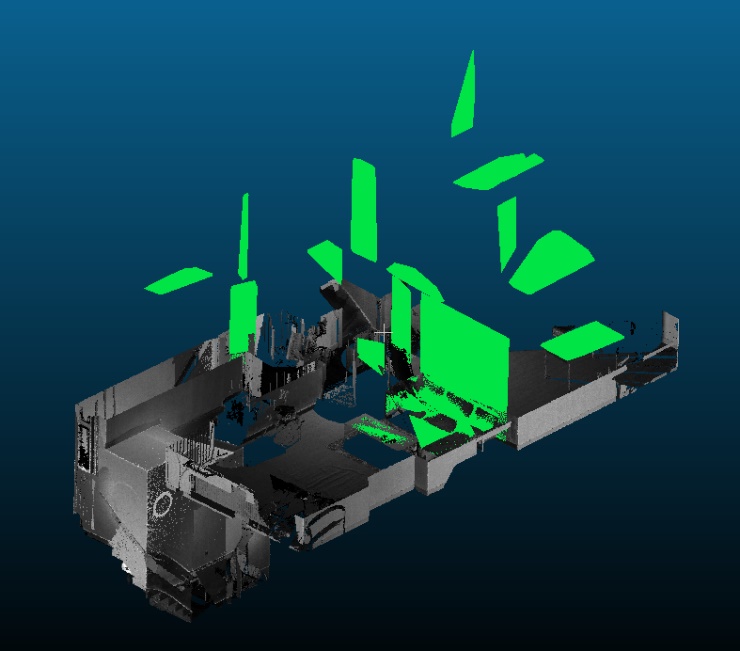
In deze paragraaf worden de resultaten van verschillende use cases uitgewerkt, die relevant zijn voor de eindgebruikers van deze methode. Hiermee laten we zien wat er wel en niet mogelijk is met het huidige model en wat voor resultaten dit oplevert. We definiëren twee situaties: een “aangebouwde muur” en een “verwijderde muur”.

De use cases zijn uitgewerkt voor het adres Aelbrechtsplein 2c, dezelfde waar de resultaten van voorgaande paragraaf van zijn weergegeven.

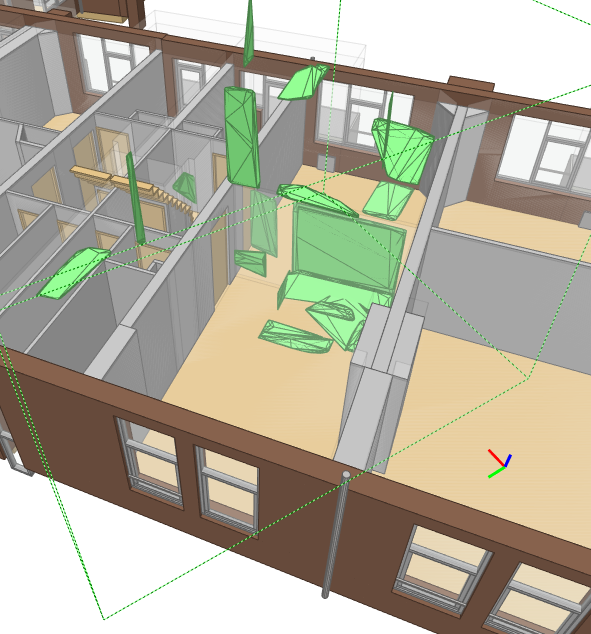
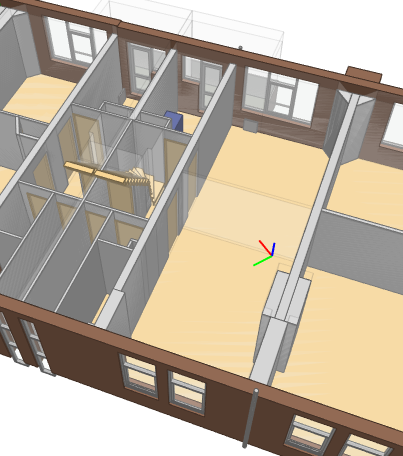
### Aangebouwde muur

Om de use case van een aangebouwde muur in huis te detecteren hebben we een muur verwijderd uit het BIM-model. Dit hebben we gedaan met de usBIM software[[11]](#footnote-12) omdat dit de enige gevonden software is waar het mogelijk is om een aanpassing te doen aan de BIM en deze daarna weer als IFC-bestand te exporteren.

Omdat de missende muur van de BIM nog wel in de puntenwolk gescand is zouden we door de puntenwolk ten opzichte van BIM-discrepanties de aangebouwde muur moeten zien. Dit is dus het resultaat van de stappen beschreven in hoofdstuk 2.2.1. De resultaten zijn gevisualiseerd in figuur 11 en figuur 12. De aangebouwde muur is zichtbaar zowel als (rood) geannoteerde punten (figuur 11 links), en als vlak als IFC-object (figuur 11 rechts). In figuur 12 links staat het input BIM-model waarin de muur is verwijderd. In figuur 12 rechts staat het resultaat in het BIM-model gevisualiseerd. Het is zichtbaar dat de muur op dezelfde plaats is gedetecteerd. Ook is het effect zichtbaar van scanschaduw van de openstaande deur in de bovenste kamer. Hierdoor is een deel van de muur gedetecteerd aan de rechterkant. Aan de linkerkant is een “deur” vlak gedetecteerd.



Figuur 11: Discrepanties puntenwolk t.o.v. BIM. Links de puntenwolk gekleurd naar afstanden. Rechts de gedetecteerde vlakken.

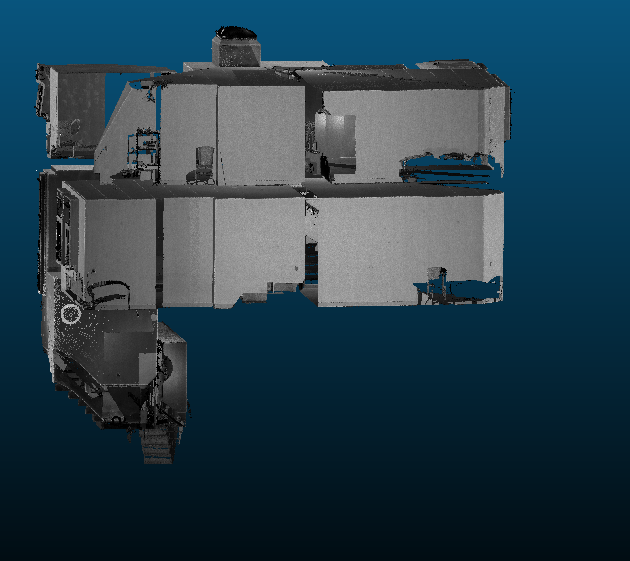
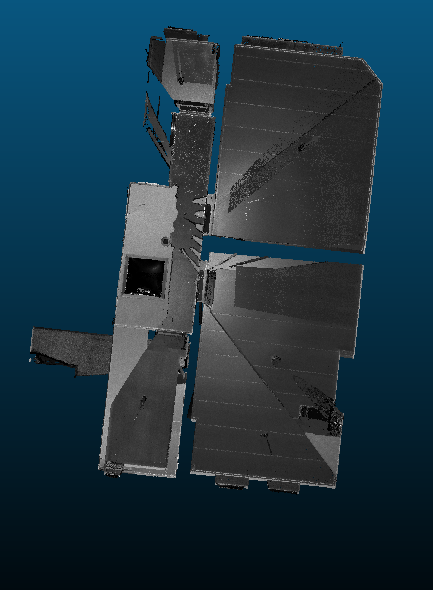


Figuur 12: Links: De BIM waar een muur uit is verwijderd. Rechts: De discrepanties puntenwolk t.o.v. BIM als IFC-vlakken.

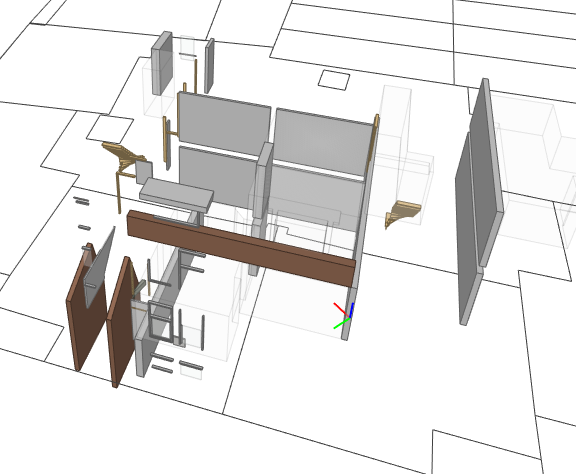
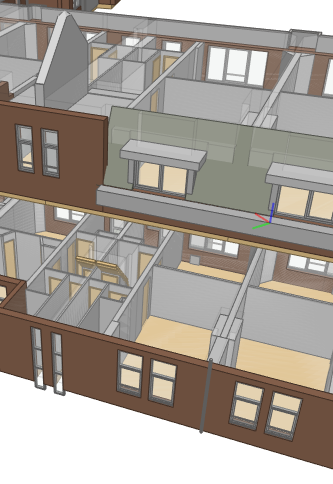
### Verwijderde muur

Om de use case van een verwijderde muur in huis te detecteren hebben we een muur verwijderd uit de puntenwolk. Dit hebben we gedaan door een selectie uit te knippen in de CloudCompare software.

Omdat de missende muur van de puntenwolk nog wel in de BIM zit zouden we door de BIM ten opzichte van puntenwolk discrepanties de verwijderde muur moeten zien. Dit is dus het resultaat van de stappen beschreven in hoofdstuk 2.2.2. De resultaten zijn gevisualiseerd in figuur 13 en figuur 14. In figuur 13 is de invoer met verwijderde muur gevisualiseerd. In figuur 14 rechts zijn de resultaten gevisualiseerd. Alle IFC-objecten die nog een punt hebben in de puntenwolk zijn overgebleven. De verwijderde muur en andere objecten worden zichtbaar in deze visualisatie.



Figuur 13: Verwijderde muur in puntenwolk. Links het bovenaanzicht, rechts het zijaanzicht. De verwijderde muur is omcirkeld.



Figuur 14: Links de originele BIM en rechts de resterende BIM-objecten van de BIM t.o.v. puntenwolk discrepanties.

## Resultaten API

In onderstaande tabel is een overzicht weergegeven van de (tussen)resultaten die uit de verwerkingsmodule komen en die via het Get-request (/api/LiDARBIM/DownloadResultsFolder) zijn op te vragen.

|  |  |  |
| --- | --- | --- |
| Stap | Outputbestandsnaam | Omschrijving |
| IFC to OBJ | InputbestandsnaamBIM.obj  InputbestandsnaamBIM.mtl | Het OBJ-bestand van de BIM, zoals omgezet vanaf een IFC-bestand. Met een bijbehorend mtl texture bestand. |
| OBJ splitsing per puntenwolk | InputbestandsnaamPuntenwolk.obj  InputbestandsnaamPuntenwolk.mtl  InputbestandsnaamPuntenwolk.las | Het OBJ-bestand van een deel van BIM, zoals deze overlapt met de bestandsnaam de bijbehorende puntenwolk. Met een bijbehorend mtl texture bestand.  Deze stap geeft ook de puntenwolk terug als .las bestand, gesubsampled met 0.01m afstand. |
| Afstanden Cloud2Mesh (puntenwolk tov BIM) | InputbestandsnaamPuntenwolk \_c2mdiscrepancies.ply | Puntenwolk met scalar field van afstanden tot de BIM, gefilterd tussen 0.2 en 10 meter. |
| Vlakkendetectie van Cloud2Mesh afstanden | InputbestandsnaamPuntenwolk \_c2mdiscrepancies\_planes\_merged.obj | OBJ-bestand met vlakken gedetecteerd van de Cloud2Mesh afstanden. |
|  | Aparte folder C2Mplanes met de losse planes die gevonden zijn | OBJ-bestanden met losse vlakken gedetecteerd van de Cloud2Mesh afstanden. |
| Export OBJ naar IFC | InputbestandsnaamPuntenwolk \_c2mdiscrepancies\_planes\_merged.ifc | IFC-bestand met de vlakken gedetecteerd van de Cloud2Mesh afstanden. |
| IFC discrepanties (BIM tov puntenwolk) | InputbestandsnaamPuntenwolk \_m2cdiscrep.ifc | IFC-bestand met BIM-objecten, waarvan er geen overlappende punten zijn gevonden in de puntenwolk |

## Vergelijking van puntenwolken verwerking met BIM

Als extra resultaat van de LiDAR met BIM-vergelijking is ook gekeken naar de resultaten van de puntenwolken verwerking zoals dit in een eerder project van CGI voor de Gemeente Den Haag is uitgevoerd.

De 3D objecten die het resultaat zijn van deze eerdere puntenwolken verwerkingsmodule zijn eerder gepresenteerd in de eindrapportage en de eindpresentatie. Echter was dit voor één test scenario uitgevoerd en is nu onderzocht of dit voor alle beschikbare inputdata automatisch mogelijk is. Het is niet mogelijk gebleken om deze analyse (vergelijking oppervlakte van LiDAR analyse en BIM-object) automatisch te doen omdat er te veel ruimtes in de puntenwolkenverwerking niet geheel gescand waren en dus niet gevonden zijn als 3D objecten.

Er is wel een handmatige vergelijking gemaakt van alle resultaten van de beschikbare puntenwolken. Het resultaat hiervan is dat de gemiddelde afwijking tussen de oppervlaktes van de ruimtes 21% is en de gemiddelde afwijking tussen de volumes van de ruimtes 23%. De totale analyse is bijgevoegd aan deze rapportage in Appendix A Vergelijking ruimtes puntenwolkenverwerking met BIM.

# Conclusie

De analyse is uitgevoerd om automatisch de discrepanties tussen een BIM-model en een puntenwolk te detecteren. In de praktijk vertegenwoordigt de BIM de "As-build" vorm van het gebouw, ofwel hoe het gebouw oorspronkelijk ontworpen was om te worden gebouwd. Aan de andere kant toont de puntenwolk de "As-is" vorm van het gebouw, ofwel hoe het gebouw in werkelijkheid is op het moment dat het werd gescand. Het doel is om de objecten te identificeren die wel aanwezig zijn in het BIM maar missen in de puntenwolk, en omgekeerd, de objecten die wel aanwezig zijn in de puntenwolk maar niet in het BIM.

De onderzoeksvraag van deze PoC was:

*“Is het mogelijk om automatisch verschillen te detecteren tussen een ingewonnen puntenwolk en een BIM-model ten behoeve van het taxatieproces van de WOZ en/of het beoordelen van een bouwvergunning?”*

Uit de resultaten getoond in hoofdstuk 3 is te zien dat het mogelijk is om een automatische vergelijking te doen tussen een puntenwolk en een BIM-model. Er is geen tussentijdse interactie nodig van de gebruiker om tot de eindresultaten te komen. De resultaten tonen visueel duidelijk de verschillen tussen beide representaties. Ook zijn deze discrepanties goed te exporteren in een 3D formaat, in dit geval IFC. Deze 3D objecten kunnen als input dienen voor de SOR-mutaties, maar zijn nog geen nieuwe officiële object registraties. De discrepanties zijn beschikbaar voor de verschillende vergelijkingen (BIM-LiDAR versus LiDAR-BIM).

De discrepanties van objecten die zich wél in de puntenwolk bevinden en niet in de BIM, dus alles wat aan de ruimtes wordt toegevoegd is duidelijk zichtbaar en accuraat. In de toekomst kan hier nog een onderscheid gemaakt worden tussen wat hiervan inrichtingselementen zijn en wat gebouwelementen zijn. Met de inrichtingselementen kan dan de functie van de ruimte geanalyseerd worden. De gebouwelementen zijn de relevante discrepanties voor de verschillende werkprocessen van taxatie en vergunningverlening en hebben dan minder last van de ‘false-positives’ van de inrichtingselementen.

Andersom, de discrepanties die zich wél in de BIM bevinden maar niet in de puntenwolk, dus alles wat aan de ruimtes is verwijderd of niet gerealiseerd is in de bouw, zijn visueel ook duidelijk zichtbaar. Hier zijn echter vaak nog te veel elementen die worden weergegeven, bijvoorbeeld doordat deze elementen van de BIM niet door de scanner zijn gescand en daarom missen in de puntenwolk. Ook zijn er elementen die slechts voor een heel klein deel in de puntenwolk extent vallen, daarom wel geanalyseerd worden maar niet overlappen met de punten van de puntenwolk en daarom ook als discrepanties worden teruggeven.

Het doel van deze onderzoeksvraag is om te onderzoeken in hoeverre deze automatische analyse bijdraagt aan de werkprocessen van de afdeling vergunningen. Gesprekken met domein experts zullen inzichtelijk maken in hoeverre de automatische analyse ook daadwerkelijk kan bijdragen aan de werkprocessen van de afdeling vergunningen. De resultaten van deze analyse slagen er echter in om "waarschuwingen" te creëren voor delen van het gebouw die verschillen vertonen tussen de "As-build" en "As-is" vormen van het gebouw. Deze waarschuwingen kunnen worden beoordeeld door een mens.

# Lessons Learned

## Puntenwolken

### Datakwaliteit

Net als in het vorige project zijn er een aantal bevindingen gedaan gerelateerd aan de inwinning van puntenwolken die voor betere kwaliteit van verwerking zorgen. Zo worden er tijdens de discrepanties tussen BIM en puntenwolk veel BIM-objecten gevonden die geen punten bevatten omdat die delen van de ruimtes niet zijn gescand. Dit zorgt voor veel extra objecten die als discrepanties tevoorschijn komen terwijl hier niet echt een relevante verandering aanwezig is. Daarom is het ook hier relevant om het proces nogmaals te doorlopen met puntenwolken van hoge kwaliteit en te toetsen of er dan nog steeds veel niet-relevante BIM-objecten gevonden worden.

### Georeferentie

De puntenwolken van Woonstad waren niet gegeorefereerd. Dit zorgt voor veel handmatig werk wat veel tijd kost. Ook zorgt een handmatige uitlijning altijd voor enige afwijking, wat bij de vervolganalyse weer kan leiden tot false positives. Wellicht kan een inwinningsapparaat met GPS-locatie het beter en zeker met RTK (Real Time Kinematics), een techniek voor precisie positiebepaling, zal dit beter worden.

## BIM

De BIM was van goede kwaliteit maar om de discrepanties nog verder te sorteren op relevantie was er meer metadata nodig. Bijvoorbeeld of een muur een dragende of een brandcompartiment is. Dit was nu niet aanwezig en daarom is er ook geen onderscheid gemaakt in het type discrepantie in het eindresultaat. Ook zijn er daardoor twee use cases uitgewerkt met een missende muur in BIM en een missende muur in puntenwolk maar niet met het brandcompartiment of de dragende muur.

## Verwerking

### Puntenwolk naar BIM discrepanties

De puntenwolk naar BIM discrepanties is de analyse geworden die uit heel veel verschillende stapjes bestaat, zoals gedetailleerd omschreven in 2.2.1. Dit had er voornamelijk mee te maken met dat het formaat van de BIM aangepast moest worden om met een software tool te kunnen werken die zowel puntenwolk als meshes aan kan. Daarnaast was de BIM ook veel groter dan de losse puntenwolken en moest deze wel opgeknipt worden.

Vervolgens kwam als feedback dat slechts een puntenwolk als visueel resultaat in de toekomst niet optimaal is voor de verschillende eindgebruikers. Ook al is hier wel veel detail zichtbaar, zijn niet duidelijk de vlakken zichtbaar die mogelijk discrepanties zijn. De vlakken detectie zorgt gelijk voor een selectie van de vele irrelevante objecten zoals banken, stoelen, lampen en personen. Ook zijn de vlakken als IFC-object te exporteren wat zorgt dat deze weer samen met de originele BIM gevisualiseerd kunnen worden.

In de toekomst is het mogelijk om de verschillende discrepanties van wat nu irrelevante objecten waren tot relevante informatie te maken door deze te classificeren in categorieën en hiermee de functie van de ruimtes te bepalen.

### BIM naar puntenwolk discrepanties

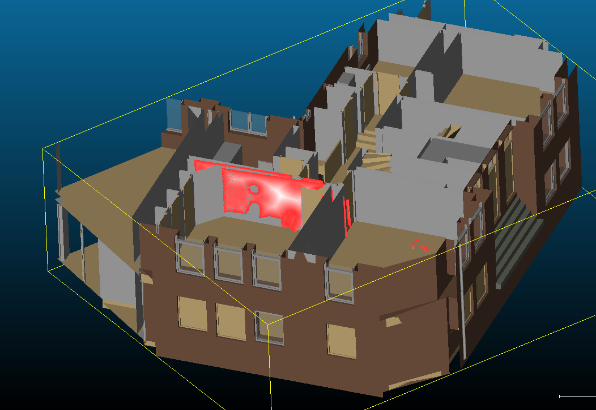
#### BIM naar puntenwolken afstanden

Voor de discrepanties van BIM naar puntenwolk hebben we als eerste dezelfde methode toegepast om ook hier de afstanden tussen BIM en puntenwolk uit te rekenen. Hiervoor moest eerst de mesh van BIM gesampled worden tot een puntenwolk. Dit is gedaan met de *Sample Mesh* tool van CloudCompare, waarin 1,000,000 punten was meegegeven als input parameter. Na deze conversie was het mogelijk de *Cloud-to-Cloud distances* tool van CloudCompare te gebruiken. Ditmaal worden de resultaten in het Scalar field *C2C absolute distances* opgeslagen, waarmee weer visueel de afstanden weergeven kunnen worden. Om het proces te versnellen zette we de maximaal zoekafstand tot het dichtstbijzijnde punt tot 1 meter.

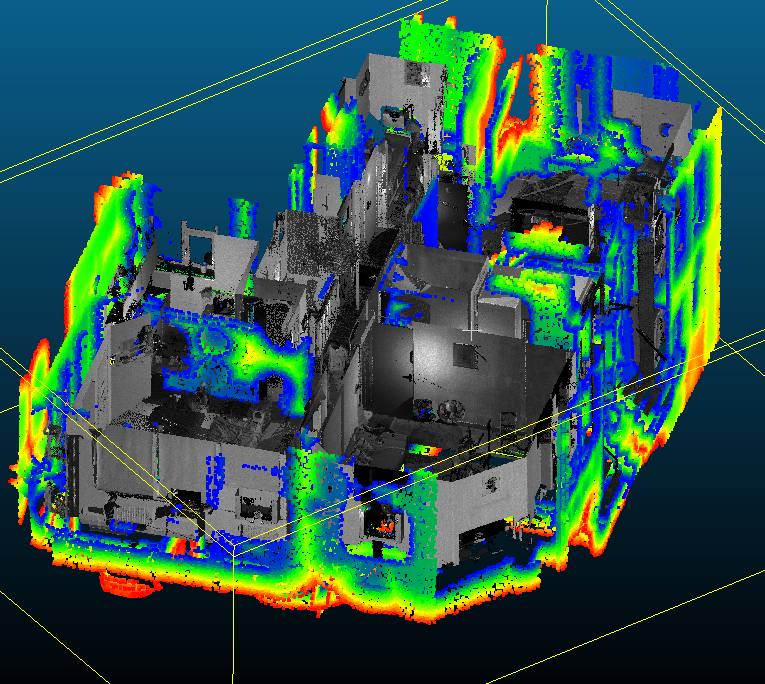
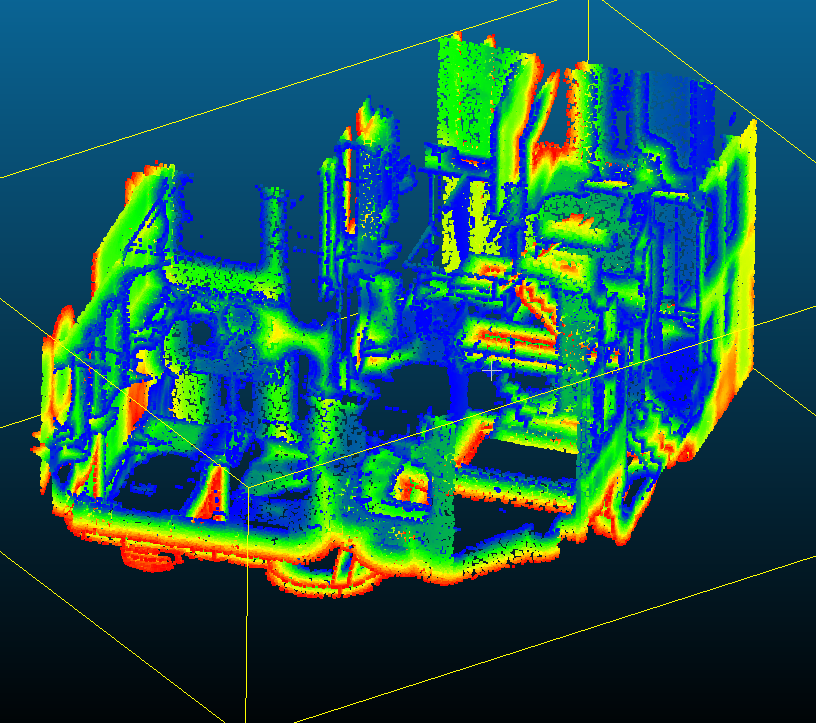
Hiermee hebben we tests gedaan van een verwijderde muur in de puntenwolk, die nog wel in de BIM staat. Hierdoor zouden deze gesamplede punten van de BIM dus een grote afstand moeten hebben tot een punt in de puntenwolk. De resultaten zijn uitgewerkt voor twee adressen en hieronder te zien.

Uiteindelijk is besloten dat de visuele representaties hiervan niet voldoende is om snel inzichtelijk te krijgen wat relevante verandering zijn. Daarom is hierna overgestapt tot de discrepanties tussen BIM en puntenwolk door middel van direct het IFC-bestand met als resultaat een IFC-bestand, zoals beschreven in paragraaf 2.2.2.

Aelbrechtskade 6b

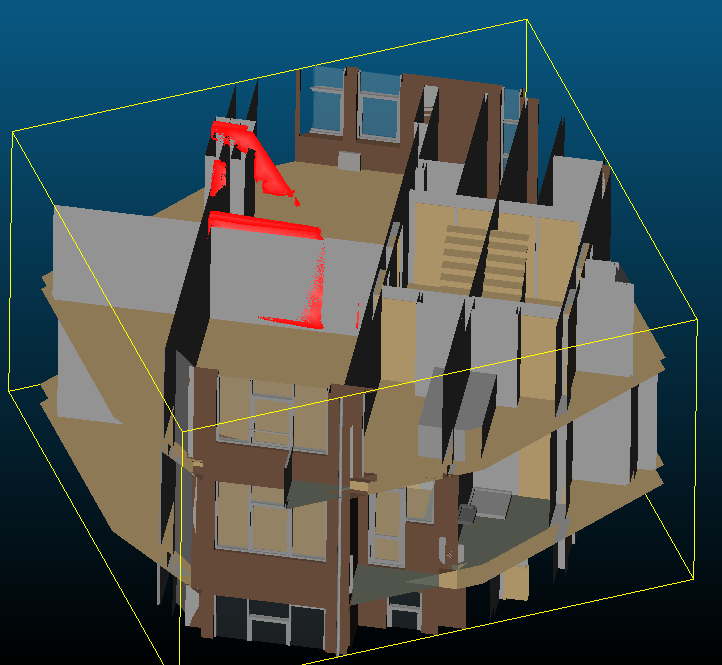
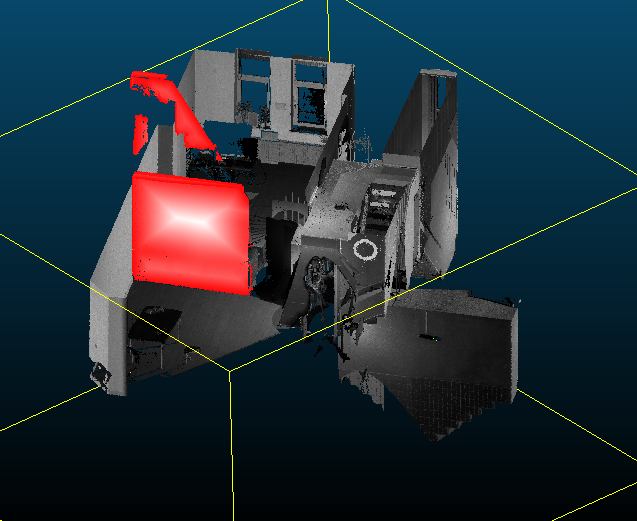


Figuur 15: In rood de muur die is verwijderd uit de puntenwolk maar nog wel in de BIM zichtbaar is.

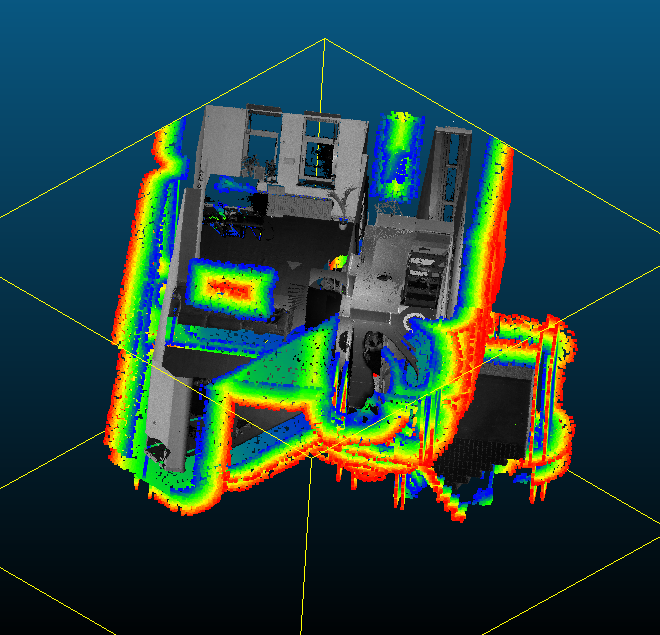


Figuur 16: Resultaten van de mesh2cloud discrepanties in kleurenschaal (blauw = 0.35 meter, rood = 1 meter afstand). Rechts ook de puntenwolk gevisualiseerd in grijs waardoor de verwijderde muur in de ruimte geplaatst kan worden.

Aelbrechtsplein 2c



Figuur 17: In rood de muur die is verwijderd in de puntenwolk (links) maar nog wel in de BIM zichtbaar is (rechts).



Figuur 18: Resultaten van de mesh2cloud discrepanties in kleurenschaal (blauw = 0.35 meter, rood = 1 meter afstand), gevisualiseerd met de puntenwolk in grijs waardoor de verwijderde muur in de ruimte geplaatst kan worden.

#### BIM naar puntenwolken IFC discrepanties

In de uiteindelijk methode voor BIM discrepanties zoals beschreven in 3.1.2 is er een afweging gedaan tussen de snelheid van de analyse en de nauwkeurigheid van de uitkomst. Hoe groter de puntdichtheid is van de puntenwolk, hoe langer het duurt om voor elk IFC-object te checken of deze wel overlappen, maar hierdoor is wel nauwkeuriger te weten of er echt geen één punt in dit IFC is opgenomen. In de huidige analyse is de subsampling van de puntenwolk op 0.1 meter gezet, waardoor er altijd 0.1 meter afstand is tussen de punten. Kleinere IFC-objecten zullen dus al snel geen punten hier tegenkomen. De analyse duurt daarmee nu circa 30 minuten voor de puntenwolken van woonstad. In de toekomst kan er afgewogen worden om de nauwkeurigheid nog iets te verbeteren en de subsampling lager te zetten en daarmee een langere runtime te accepteren. Of andersom, een sneller resultaat met minder hoge nauwkeurigheid.

# Aanbevelingen

## Methode

### Input data

Aangezien het doel is om de verschillen tussen een puntenwolk en een BIM automatisch te detecteren, is het belangrijk dat beide worden gegeorefereerd in hetzelfde coördinatensysteem. In puntenwolken is georeferentie een proces dat wordt uitgevoerd na het verzamelen van de gegevens en het wordt aanbevolen om de gegevens altijd na te bewerken zodat ze verwijzen naar het gewenste coördinatensysteem. Als ze zich in een ander coördinatensysteem bevinden, moet de puntenwolk worden getransformeerd naar het coördinatensysteem van de BIM.

Als de puntenwolk geen georeferentie heeft, moet handmatige uitlijning worden toegepast, zoals beschreven in paragraaf 2.1.3. Handmatige uitlijning kan echter door onnauwkeurigheid leiden tot een puntenwolk met een verschuiving en rotatie in vergelijking met de BIM, wat vervolgens de discrepanties zal beïnvloeden, aangezien sommige onderdelen als verschillen worden gedetecteerd vanwege de verschuiving en rotatie.

Ook kunnen nauwkeurigere resultaten worden geproduceerd als de BIM en de puntenwolk exact hetzelfde deel van het gebouw vertegenwoordigen. Anders kunnen enkele extra objecten die hier net buiten vallen als discrepanties worden gedetecteerd, zoals in de use-cases van deze analyse.

### Procesoptimalisatie

In de ontwikkeling van deze verwerkingsmodule heeft er, net als bij het LiDAR project, subsampling op de puntenwolk plaatsgevonden. Dit is een kwaliteits-performance afwegingspunt waar een best effort inschatting op is gedaan. Aan de ene kant is gebruik van alle punten van de puntenwolk goed voor de kwaliteit van de resultaten, maar aan de andere kant zorgt het voor onnodig veel berekeningen om tot dezelfde kwaliteit resultaten te komen. Subsampling tot een “redelijk” niveau is aan te raden om onnodige complexiteit te verminderen, maar een precieze analyse op impact is nog nodig.

## Implementatie

Den Haag (of een derde partij) kan de componenten gebruiken in de aangeleverde code om een pipeline te ontwikkelen. De code is zo opgezet dat ieder component een klein doel heeft om te vervullen en deze zijn geplaatst in Docker images. De images kunnen via een externe tool worden afgetrapt om zo een pipeline te bouwen. Voor opschaling kan een orchestratietool zoals Kubernetes worden gebruikt om de individuele componenten horizontaal te schalen waar nodig.

Het plaatsen van de code in containers is geen harde eis voor de analyse. Er is ook de mogelijkheid om de componenten zonder container uit te voeren en deze zelf te implementeren in de gewenste infrastructuur.

## Aansluiting op de eindgebruikers

De objecten die in het project worden gedetecteerd, vertegenwoordigen niet noodzakelijk alle echte verschillen tussen een BIM en een puntenwolk. Ze kunnen ook objecten bevatten die niet van belang zijn (zoals meubels) die niet worden onderscheiden van de belangrijke (zoals muren) in deze PoC. Ze creëren echter nuttige "alarmen" voor de delen van het gebouw waar aandacht aan moet worden besteed bij het taxatieproces van de WOZ en de beoordeling van een bouwvergunning.

In de toekomst is het mogelijk om de verschillende discrepanties van wat nu irrelevante objecten waren tot relevante informatie te maken door deze te classificeren in categorieën en hiermee de functie van de ruimtes te bepalen.

1. Vergelijking ruimtes puntenwolkenverwerking met BIM

AppendixVergelijkingRuimtes.xlsx



[**cgi.com**](https://www.cgi.com)

1. <https://www.cloudcompare.org/doc/wiki/index.php?title=FILE_I/O> [↑](#footnote-ref-2)
2. <http://20.101.41.248:8081/swagger/index.html> [↑](#footnote-ref-3)
3. <http://ifcopenshell.org/ifcconvert> [↑](#footnote-ref-4)
4. <https://laspy.readthedocs.io/en/latest/> [↑](#footnote-ref-5)
5. <https://www.cloudcompare.org/doc/wiki/index.php?title=Crop> [↑](#footnote-ref-6)
6. <https://www.cloudcompare.org/doc/wiki/index.php?title=Cloud-to-Mesh_Distance> [↑](#footnote-ref-7)
7. <http://www.open3d.org/docs/latest/tutorial/Basic/pointcloud.html#Plane-segmentation> [↑](#footnote-ref-8)
8. <http://www.open3d.org/docs/latest/tutorial/Basic/pointcloud.html#DBSCAN-clustering> [↑](#footnote-ref-9)
9. <http://www.open3d.org/docs/latest/tutorial/Basic/pointcloud.html#Convex-hull> [↑](#footnote-ref-10)
10. <https://www.freecadweb.org/> [↑](#footnote-ref-11)
11. <https://www.accasoftware.com/en/bim-management-system> [↑](#footnote-ref-12)