



# Augmented Reality voor bouw & infra

Mark Vrielink  
Afstudeerscriptie, 2012  
TGEO-LAF04-10

Versie  
**Censuur**  
2 oktober 2012



Hogeschool Utrecht

*Faculteit Natuur & Techniek*

*Instituut voor Gebouwde Omgeving*

# **Afstudeeronderzoek**

## **Geodesie, Geo-informatica**

*TGEO-AFO4-10*

Uitgevoerd voor

Witteveen+Bos Raadgevende Ingenieurs B.V., Deventer

*Sector Water, PMC Informatietechnologie, afdeling Geo-Informatie*



## Student

Mark Vrielink

studentnummer: 1574392

Oude Lochemseweg 23

mark\_vrielink@outlook.com

7245 VH EXEL

06 18488586

## Begeleiding

### Hogeschool Utrecht

Drs. F. de Vroege – *eerste begeleider* – frans.devroege@hu.nl – 06 40 235 700

Drs. E. Heere – *tweede begeleider* – elger.heere@hu.nl – 088 4818317

### Witteveen+Bos

Ing. B. Roosendaal – *GIS Consultant* – b.roosendaal@witteveenbos.nl – 0570 697268

Drs. S. Braakman – *GIS Consultant en Manager* – s.braakman@witteveenbos.nl – 0570 697268

## Geheimhouding

Met Witteveen+Bos Raadgevende Ingenieurs B.V. is afgesproken bepaalde content niet openbaar te willen delen. Daarom zijn er twee varianten van de scriptie opgemaakt.

De gecensureerde content heeft betrekking tot de:

### Programmeercode

- Paragraaf 7.3 Programmering PHP: code waarin de server-processen gedefinieerd zijn.
- Bijlage B2; Optimaliseren van datasets.
  - Bijlage “AR-Project invulsheet”; dat geautomatiseerd code exporteert voor zowel een nieuwe projectdatabase met bijbehorend serverscript.
- Bijlage B2.3 JSON Response: het afgeleide antwoord dat met de PHP code verkregen wordt.

### Afbeeldingen

- Een aantal afbeeldingen betreffen gerenderde 3D modellen die gebruikt zijn voor de productie van augments. Doordat de projecten, waarop deze modellen betrekking hebben, zich momenteel in de ontwerpfase betreffen, wordt de content hiervan niet publiekelijk gedeeld.

*Ter beoordeling geldt de originele scriptie. De tweede variant betreft een gecensureerde scriptie en mag openbaar gepubliceerd worden in de “Databank alle scripties GEO” danwel overige deposito.*

**Dit betreft de gecensureerde versie**



# VOORWOORD

Voor u ligt de scriptie van het afstudeeronderzoek “Augmented Reality voor bouw & infra” dat ik in de periode van 10 april 2012 t/m 24 augustus 2012 in opdracht van Witteveen+Bos Raadgevende Ingenieurs b.v. te Deventer, Van Twickelostraat 2, heb uitgevoerd in het kader van mijn afstuderen aan de Hogeschool Utrecht, opleiding *Geodesie, Geo-informatica voor werkenden dual*.

Graag wil ik Witteveen+Bos bedanken voor de begeleiding, verkregen kennis en feedback tijdens het afstudeeronderzoek. Van de Hogeschool Utrecht wil ik docentbegeleiders Frans de Vroege en Elger Heere bedanken voor de ondersteuning tijdens het onderzoek en de terugkoppeling op de concepten. Mijn moeder wil ik bedanken voor de grammaticale controle op de gehele scriptie.

Met speciale dank aan:

**Bert Roosendaal** (*Witteveen+Bos*) en **Steven Braakman** (*Witteveen+Bos*) voor de directe begeleiding in de gehele afstudeerperiode.

**Paul Drewes** (*Witteveen+Bos*) voor de uit te werken 3D-modellen en de daarbij verkregen feedback.

**Niels Lammers** (*Witteveen+Bos*) en **Gerben Aaltink** (*MA-IT*) voor de ondersteuning in PHP-programmering en SQL.

**Harmen Kampinga** (*Witteveen+Bos*) voor de hulp bij het onderzoek naar bruikbare conversiemogelijkheden vanuit ArcGIS.

# SAMENVATTING

Deze scriptie is geschreven in het kader van het afstudeeronderzoek ter afsluiting van de opleiding Geodesie, Geo-informatica voor werkenden dual aan de Hogeschool Utrecht. De visualisatietechniek “augmented reality” is *(met name in de bouw- & infrabranche)* een relatief onbekende methode om bijvoorbeeld toekomstige bouw- en infraprojecten mee te presenteren. Om er achter te komen hoe en óf deze techniek een bijdrage in deze branche kan leveren, is dit onderzoek uitgevoerd.

Het doel van dit onderzoek is om een product in de vorm van augmented reality te realiseren, dat voor een breed, dan wel besloten publiek toegankelijk gemaakt kan worden. Dit product zal aantonen dat augmented reality, als zijnde een visualisatietechniek, een grote bijdrage kan leveren met name in de vorm van tijdsbesparing voor toekomstige bouwprojecten in de ontwerpfase.

De hoofdvraag van de onderzoek luidt dan ook:

*In hoeverre kan de visualisatietechniek “augmented reality” in de buitenruimte worden toegepast?*

De bestaande mobiele telefoonapplicatie “Layar” staat hierin centraal. Zo wordt onder andere:

- Layar vergeleken met andere mobiele telefoonapplicaties;
- onderzocht welke mobiele apparatuur geschikt is voor het uitvoeren van de Layarapplicatie;
- zelf een server met serverscript in combinatie met een SQL-database ontworpen en gerealiseerd, die augmented realitymodellen serveert naar de Layar browser;
- onderzocht op welk schaalniveau een augment gegeneraliseerd moet worden, zodat de best haalbare weergave gewaarborgd blijft en het mobiele apparaat de maximale grafische prestaties kan leveren;
- een protocol geschreven, dat alle modeldefinities, database-instellingen en bijbehorende processen beschrijft om tot een werkend augment te komen. Het protocol beschrijft tevens hoe een bestaande database gemuteerd en uitgebreid kan worden.

## Conclusies en resultaten

In de scriptie worden alle bovengenoemde onderdelen behandeld. De belangrijkste resultaten hiervan betreffen een goed werkende Layarserver, waarbij een aantal processen voor het vullen van de SQL-database geautomatiseerd verlopen en een workflow waarin alle stappen van ontwerp tot en met publicatie voor een augment en Layarserver worden beschreven. Deze en andere resultaten en producten worden uiteindelijk in de praktijk getest en beschreven in diverse casussen.

# INHOUDSOPGAVE

<b>VOORWOORD .....</b>	<b>5</b>
<b>SAMENVATTING .....</b>	<b>6</b>
BEGRIPPENLIJST .....	9
<b>1. INLEIDING .....</b>	<b>11</b>
1.1 STELLING .....	11
1.2 DOELSTELLING .....	11
1.3 AANPAK .....	12
1.4 LEESWIJZER .....	13
2.1 OVER WITTEVEEN+BOS .....	14
<b>2. WITTEVEEN+BOS .....</b>	<b>14</b>
2.2 SECTOR WATER .....	14
2.3 PMC INFORMATIETECHNOLOGIE .....	14
2.4 GROEP GEO-INFORMATIE .....	14
<b>3. HISTORIE VAN AUGMENTED REALITY .....</b>	<b>15</b>
<b>4. INVENTARIS AR-SOFTWARE .....</b>	<b>17</b>
4.1 VERKLARING SOFTWARE-EIGENSCHAPPEN .....	17
4.2 INVENTARISATIE AR-SOFTWARE .....	18
4.3 CONCLUSIE INVENTARIS AR-SOFTWARE .....	18
4.4 LAYAR VERSUS ADOBE FLASH PRO .....	19
4.5 BEELDHERKENNING DIVERSE AR-SOFTWARE .....	20
<b>5. INVENTARIS MOBIELE APPARATUUR .....</b>	<b>21</b>
5.1 VOORWAARDEN BESTURINGSSYSTEEM .....	21
5.2 VOORWAARDEN HARDWARE .....	21
5.3 BEREKENING PERFORMANCE MOBIELE APPARATUUR .....	22
5.4 CONCLUSIES INVENTARIS MOBIELE APPARATUUR .....	22
<b>6. INVENTARIS SUPPORT &amp; DIENSTEN .....</b>	<b>23</b>
<b>7. INRICHTEN LAYARSERVER .....</b>	<b>24</b>
7.1 ALGEMENE WERKING VAN LAYAR .....	24
7.2 DATABASE ONTWERP SQL .....	26
7.3 PROGRAMMERING PHP .....	29
7.4 RESULTAAT .....	35
<b>8. GENERALISATIE 3D-MODELLEN .....</b>	<b>36</b>
WAAROM GENERALISEREN? .....	36
8.1 WAVEFRONT 'S OBJ .....	36
8.2 WAVEFRONT 'S MTL .....	38
8.3 GESTUURDE GENERALISATIE .....	39
8.4 CONCLUSIE GENERALISATIE .....	39
<b>9. BEELDHERKENNINGSTECHNIEK .....</b>	<b>40</b>
9.1 DIGITALE BEELDBEWERKING / BEELDHERKENNING .....	41
9.2 ORIËTERING APPARAAT & AUGMENT .....	44
9.3 CONCLUSIE BEELDHERKENNING .....	47
<b>10. CASUS NAUWKEURIGHEID .....</b>	<b>48</b>
<b>11. CASUS RIOLERING .....</b>	<b>51</b>

<b>12. CONCLUSIES .....</b>	<b>54</b>
BRONVERMELDING.....	58
<b>BIJLAGEN .....</b>	<b>61</b>
B1(A) INVENTARIS BRUIKBARE MOBIELE APPARATUUR.....	62
B1(B) GRAFISCHE KWALITEIT MOBIELE APPARATUUR.....	64
<b>B2 OPTIMALISEREN VAN DATASETS .....</b>	<b>65</b>
B2.1 EXCEL INVULBLAD.....	65
B2.2 TABEL EN ATTRIBUUTDEFINITIES .....	66
B2.3(A) JSON .....	70
B2.3(B) JSON RESPONSE .....	71
<b>B3 GENEREREN &amp; PUBLICEREN .....</b>	<b>73</b>
B3.1 ONTWERPSOFTWARE .....	74
B3.2 ONTWERPVOORWAARDEN .....	76
B3.3 LAYAR 3D MODEL CONVERTER.....	77
B3.4 REGISTRATIE EN PUBLICATIE .....	79



# Begrippenlijst

## MESHSMOOTH

AutoCAD-commando

Converteert AutoCAD-entiteiten als 3D-solids, 3D-surfaces, polygonmeshes, polyfacemeshes, regions en gesloten 2D- / 3D-polylines naar een mesh-model. Dit meshmodel kan omgezet worden naar 3D-faces.

## 3D-face

AutoCAD-entiteit

Vlak bestaande uit 3 of 4 vertices. "Meestal" het product als gevolg van een conversieslag op een volumiek- of triangulatiemodel. Een 3D-face kan tevens direct getekend worden.

## 3D-solids

AutoCAD-entiteit

3-dimensionale objecten met een volume.

## Accelerometer

Detecteert verplaatsing en versnelling.

## ActionScript

Codetaal dat in Adobe Flash Pro wordt gebruikt om onder andere AR-software mee te ontwikkelen.

## Adobe Flash Pro

Adobe Flash is een programma waarmee vectorafbeeldingen / animaties gemaakt kunnen worden. Veelal wordt Flash gebruikt om websites mee aan te kleden. Om Flash-content weer te geven dienen plugins geïnstalleerd te worden.

## Android

OpenSource besturingssysteem van Google.

Android is het meest toegepaste besturingssysteem voor smartphones en tabletcomputers.

## Android Market

Online applicatiewinkel voor apparaten die gebruik maken van het Google Android besturingssysteem.

## API

Application Programming Interface

Een software-interface tussen een programmeertaal en een applicatie dat toegang biedt tot het bestandssysteem, netwerkbronnen etc.

## Apple Appstore

Online applicatiewinkel voor apparaten die gebruik maken van het Apple iOS besturingssysteem.

## AR

Afkorting voor "Augmented Reality".

## AR-media

Augmented reality softwarepakket op basis beeldherkenning.

## Augmentdefinitie

Zowel het model, als alle parameters waarmee het augment gedefinieerd wordt (*geometrische eigenschappen, kleur, textuur, belichting*).

## Augmented Reality (AR)

(*toegevoegde realiteit*) Is een visualisatietechniek waarbij virtuele objecten, die geregistreerd zijn in de echte wereld, op zijn werkelijke positie weergegeven worden. Deze objecten kunnen bekeken worden met apparaten voorzien van camera en beeldscherm. Het apparaat dient zijn eigen positie en oriëntatie ten opzichte van het augment te kunnen bepalen.

## Augments

Synoniem voor augmented realitymodellen. Dit betreft het computergegenereerde model dat in het camerabeeld geprojecteerd wordt.

## Beeldherkenning

Methodiek van relatieve positiebepaling. Het waargenomen beeld wordt herkend met een sterk overeenkomend beeld in de bibliotheek. Het te projecteren model is met de publicatie gerefereerd aan en ten opzichte van dit beeld.

## Besturingssysteem

Samenhangend geheel van systeemprogramma's, waarmee alle programma's op het apparaat worden aangestuurd en uitgevoerd.

## Binair

Zwart/wit, 1/0 (*tiny-integer*), aan / uit, ja / nee.

## Biw

Brief Information Widget, een (*pop-up*)venster in een applicatie met informatie over de betreffende augment.

## GBKN

Grootschalige Basiskaart Nederland.

## Gyroscoop

Component dat de rotatie om de X- én Y-as detecteert.

## HMD

Head mounted display. Een op het hoofd gedragen beeldscherm.

## HSL

Hue Saturation Value, kleurenstelsel waarbij de kleuren van donker naar licht zijn gedefinieerd.

**HTML(5)**

HyperText Markup Language. De taal waarmee hoofdzakelijk websites zijn opgemaakt.

**HUD**

Head-Up Display. Een systeem waarbij informatie rechtstreeks geprojecteerd wordt in het gezichtsveld.

**Indexeren (Layar)**

Indexeren houdt in dat Layar de fysieke bestanden (*L3D, PNG, MP3 etc.*) opslaat in zijn eigen digitale geheugen (*cachen*). Het gevolg hiervan is dat het augment beter vindbaar wordt, maar het kan niet meer live verwijderd noch gemuteerd worden. Het wel of niet indexeren heeft géén gevolgen voor eventuele wijzingen in de SQL-database.

**iOS**

Besturingssysteem voor de Apple-producten iPad, iPhone en iPod.

**JSON**

JavaScript Object Notation:

Betreft een licht data-uitwisselformaat dat relatief simpel te lezen / schrijven is en wordt ondersteund door nagenoeg alle moderne programmeertalen.

**Labels (Layar)**

Tekstuele data die als augment weergegeven worden. In Layar zijn alle iconen voorzien van een label.

**Latitude**

Breedtegraad van een polair coördinatenstelsel.

**Layar**

Een service dat gebruikt kan worden om geserveerde augments mee weer te geven. Hiervoor dient wél een Layarserver gebouwd te worden.

**Layarserver**

Een server die speciaal is ingericht om de mobiele telefoonapplicatie "Layar" te serveren van augments. De webserver is opgebouwd uit een PHP-script in combinatie met een SQL-database.

**Longitude**

Lengtegraad van een polair coördinatenstelsel.

**Magnetometer**

Detecteert de rotatie om de Z-as ten opzichte van het noorden.

**Orbit-model**

Is een digitaal 3D-model dat alleen vanaf de buitenkant goed bekeken kan worden. Een synoniem voor "orbit-model" is "buitenste schil" (*van het model*).

**PHP**

PHP Hypertext Preprocessor is een voor de gebruiker onzichtbare programmeertaal dat veelal wordt toegepast in websites.

**Positiebepaling**

Methodiek voor het verkrijgen van de apparaatpositie ten opzichte van het te projecteren beeld.

**Processor**

(*ook wel CPU*) Voert rekenkundige en logische bewerkingen uit. De bewerkingen en resultaten vloeien voort uit de input van de programmeertaal.

**RAM**

Random Access Memory: werkgeheugen. Een geheugen van een apparaat dat gevuld wordt met data van een programma dat op dat moment wordt uitgevoerd.

**Reflectorzicht**

Een kijker waarin een afbeelding wordt geprojecteerd. Meestal betreft de afbeelding een punt of kruisdraad, doordat het reflector-zicht vaak fungeert als hulpstuk voor diverse wapens.

**Right-hand-rule**

De zichtbare zijde van een 3D-face is getekend volgens de right-hand-rule. De vertexpunten van het vlak zijn gezien vanaf de zichtbare zijde tegen de klok in getekend.

**SDK**

Software Development Kit: een verzameling van tools dat gebruikt wordt voor de (*door*)ontwikkeling van een computerprogramma.

**Verwerkingssnelheid**

Prestatie van de processor (*in bytes per seconde*).

**WGS84**

World Geodetic System 1984: het geodetische stelsel dat voornamelijk gebruikt wordt door GPS.

# 1. INLEIDING

## 1.1 Stelling

Bij Witteveen+Bos is er de overtuiging dat augmented reality een waardevolle toevoeging kan leveren voor projecten in de buitenruimte. Augmented reality (*toegevoegde realiteit*) is een visualisatietechniek waarmee virtuele objecten worden geprojecteerd in een realtime gefilmd beeld. Deze virtuele objecten zijn gerefereerd aan “iets” in de echte wereld. In dit afstudeeronderzoek betreft “iets” de geografische ligging.

Tot op heden wordt augmented reality niet tot nauwelijks in de bouw- en infrasector toegepast. Vanuit Witteveen+Bos is aangegeven deze techniek te willen onderzoeken en bij positieve resultaten, het commercieel in te zetten.

Wanneer dit gebeurt, dient het publiek zelfvoorzienend te zijn in de te gebruiken apparatuur (*smartphone of tabletcomputer*), zodat met behulp van een mobiele telefoonapplicatie de augmented realitymodellen (*hierna te noemen augments*) bekeken kunnen worden.

## 1.2 Doelstelling

Het vormen van een product / communicatiemiddel waardoor gebruikers op de geografische planlocatie, door middel van de visualisatietechniek “augmented reality”, inzicht krijgen in de grootte en vormgeving van een project in de ontwerpfase.

Hierdoor kunnen gebruikers (*opdrachtgevers, overheden en overige betrokkenen*) al vanaf de eerste conceptontwerpen mee oordelen en feedback geven over het project. De acceptatie van het ontwerp door de betrokkenen zal hierdoor sneller gerealiseerd worden, waardoor mogelijk de uitvoering ook vervroegd kan worden.

Om tot de doelstelling te komen, worden in de scriptie de volgende onderzoeksvragen beantwoord:

- In hoeverre kan de visualisatietechniek “augmented reality” in de buitenruimte worden toegepast?
- Wat is het huidige marktaanbod voor augmented realitysoftware? Wat zijn de onderlinge verschillen? Is Layar nog steeds de meest geschikte keuze?
- Welke mobiele apparatuur (smartphones / tabletcomputers) is geschikt voor het gebruik van augmented reality? Welke apparaten presteren hierbij het beste?
- Hoe dient een eigen Layarserver ingericht te worden om zelf augments te kunnen serveren?
- Welke partijen kunnen eventueel benaderd worden voor Layar ontwikkelaarssupport?
- Wat is voor de mobiele apparatuur van nu de best haalbare weergave van augments? Hoe kan deze weergave gerealiseerd worden?  
Hoe kunnen zware (gedetailleerde) modellen goed gegeneraliseerd worden?
- (Hoe) kunnen de nauwkeurigheidsproblemen in de plaatsbepaling geminimaliseerd worden?
- Voor welke toepassingen kan Layar als medium ingezet worden?

## 1.3 Aanpak

Om de doelstelling te bereiken is het onderzoek opgedeeld in vier fasen die bestaan uit een literatuurstudie, inventarisaties, onderzoeksfase met casussen en de afrondingsfase. Het volledige plan van aanpak is te vinden in de bijlagen. Onderstaand volgt een korte samenvatting hiervan.

### Literatuurstudie

Om kennis te vergaren van de visualisatietechniek “augmented reality”, de mobiele telefoonapplicatie Layar, PHP programmering, SQL en de techniek “beeldherkenning” is allereerst een literatuurstudie uitgevoerd. De benaderde literatuur is terug te vinden in de bijlagen onder “Bronvermelding / Bibliografie” en wordt (*wanneer gebruikt*) verwezen bij de alinea.

### Inventarisaties

In deze fase worden inventarisaties verricht waarmee een overzicht gevormd wordt van:

- het aanbod van mobiele apparatuur met een bijhorende prestatie-index;
- de aangeboden software met betrekking tot augmented reality, waarbij onderscheid wordt gemaakt in het type af te beelden augments en de verkrijging van de apparaatpositie en oriëntering;
- bedrijven die ondersteuning aan Layarontwikkelaars kunnen geven.

### Onderzoeks- en analysefase in combinatie met casussen

In de onderzoeksfase wordt een augmented realityserver ontworpen, waarbij een eigen serverscript met een bijhorende SQL-database gevormd wordt. Ook wordt onderzocht waar tijdens de productie van augmented realitymodellen op gelet dient te worden, alvorens deze juist en representatief fungeren in de Layarapplicatie. De onderzoeksfase is grotendeels uitgevoerd in combinatie met diverse casussen.

### Afrondingsfase

Aan de hand van de resultaten en bevindingen van de voorgenoemde fasen zijn producten gevormd. Deze producten betreffen:

- inventarissen van mobiele apparatuur, augmented realitysoftware en Layarsupport;
- augmentserver (*eigen ontwerp*);
- protocol voor het opzetten van de augmentserver;
- protocol voor het genereren en publiceren (*maken van augmented realitymodellen incl. de hosting daarvan*).



## 1.4 Leeswijzer

### Scriptie

De afstudeerscriptie start met een beschrijving van het afstudeerbedrijf “Witteveen+Bos” (*hoofdstuk 2*).

Vervolgens wordt in hoofdstuk 3 kort beschreven hoe augmented reality ontdekt en verder ontwikkeld is tot de visualisatietechniek die wij tot de dag van vandaag kennen.

De hoofdstukken 4, 5 en 6 betreffen de inventarissen van mobiele apparatuur, augmented realitysoftware en bedrijven die ondersteuning kunnen leveren voor Layar-ontwikkelaars.

Hoofdstuk 7 beschrijft het grootste onderdeel van de scriptie, namelijk het ontwerpen van een serverscript en een SQL-database. Het serverscript beantwoordt bevestigingen die vanuit de mobiele telefoonapplicatie Layar gesteld worden. De beantwoording wordt gevormd door tabelattributen in de SQL-database, waarna deze in JSON-formaat verzonden worden naar het mobiele apparaat.

In hoofdstuk 8 wordt omschreven hoe de augmented realitymodellen zijn opgebouwd én wanneer de server kiest voor een vereenvoudigde weergave van een augment.

Een alternatieve manier om augments te positioneren is beeldherkenning. In hoofdstuk 9 wordt beschreven hoe met deze techniek de apparaatpositie ten opzichte van de augmentpositie bepaald kan worden.

De hoofdstukken 10 en 11 betreffen casussen. De Layarserver wordt hierbij in de praktijk getest met diverse augmented realitymodellen onder verschillende omstandigheden.

Alle onderzoeksvragen in het plan van aanpak worden in het kort beantwoord in hoofdstuk 12 (conclusies). Voor de volledige beantwoording wordt in hoofdstuk 12 verwezen naar de bijbehorende onderdelen in de scriptie.

### Bijlagen

Ter completering van de scriptie zijn er bijlagen toegevoegd. In de bovengenoemde scriptieonderdelen wordt hiernaar verwezen. De bijlagen betreffen:

- de inventaris van bruikbare mobiele apparatuur;
- een uitleg van de databasestructuur (*alle tabellen en attributen worden beschreven*);
- JSON-response: het antwoord dat gevormd wordt door het bevestigen van het PHP-script;
- een workflow voor het maken en publiceren van augmented realitymodellen.

## 2. WITTEVEEN+BOS

### 2.1 Over Witteveen+Bos

In 1946 is Witteveen+Bos opgericht als zijnde een “Stedebouwkundig advies- en ingenieursbureau” en is inmiddels gegroeid tot een toonaangevend ingenieursbureau met circa 900 werknemers dat voor advies en ontwerp met name in de sectoren water, infrastructuur, milieu, ruimte en bouw actief zijn.

Witteveen+Bos werkt wereldwijd aan projecten en heeft zich naast Nederland gevestigd in onder andere België, Kazachstan, Indonesië, Rusland, Letland en Vietnam.

Witteveen+Bos kent een platte organisatiestructuur (*4 lagen*). Vanaf directioniveau omlaag gekeken worden de diverse bedrijfsonderdelen ingedeeld in sectoren, vervolgens in product-markt-combinaties (*PMC*), gevolgd door de groepen in de *PMC*'s.

### 2.2 Sector Water

Voor Witteveen+Bos omvat het werkgebied water de gehele waterketen en het gehele watersysteem. De specialisten in de gehele sector water adviseren over waterwinning, de bereiding van drinkwater, riolering, procesautomatisering, behandeling en hergebruik van afvalwater, watermanagement en waterbouw.

### 2.3 PMC Informatietechnologie

Informatietechnologie wordt toegepast in projecten binnen alle werkkerreinen van Witteveen+Bos. De *PMC Informatietechnologie* richt zich op de automatisering van technische processen.

### 2.4 Groep Geo-informatie

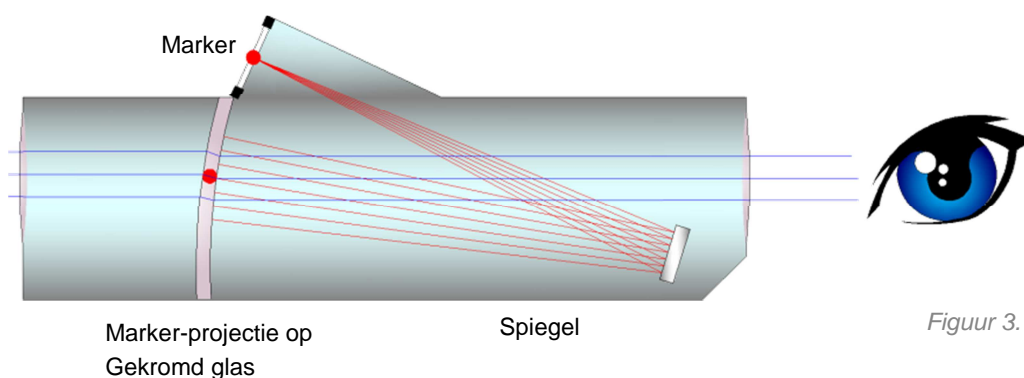
De groep Geo-informatie is verantwoordelijk voor een gecontroleerde informatievoorziening. Door het uitvoeren van ruimtelijke analyses op aangeleverde data (*leggers, ontwerpen*) kunnen zij deze controleren en hierover een advies uitspreken. Deze rekenprocessen zijn vaak van een dergelijk grote omvang, dat de GIS-specialisten ter plekke een geautomatiseerde oplossing bedenken en programmeren. Hiernaast is de groep Geo-informatie ook verantwoordelijk voor het databeheer, de kartografische opmaak van (*thematische*) kaarten en systeemontwikkeling.

Om een nog bredere informatievoorziening te zijn voor interne en externe klanten wordt het afstudeeronderzoek “Augmented Reality voor Bouw & Infra” voor en bij de groep Geo-informatie uitgevoerd.

### 3. HISTORIE VAN AUGMENTED REALITY

#### 1900 Head up display (HUD), Sir Howard Grubb

De eerste “met augmented reality vergelijkbare toepassing” betreft het reflector-zicht / reflex-zicht (zie *figuur 3.1*). Nog steeds wordt dit instrument als hulpmiddel bevestigd op diverse wapens, waarbij een marker (*kruisdraad of punt*) geprojecteerd wordt in het zicht van de kijker. Ook wordt deze techniek toegepast in gevechtsvliegtuigen, waarbij het raam in de cockpit als projectievlak fungeert. De relatie met augmented reality betreft, dat als de gebruiker zijn/haar oogpunt verandert ten opzichte van de kijker, de geprojecteerde marker stil staat in het te bekijken gebied. Deze augmented realitytechniek valt onder de categorie “head up display” (HUD).



*Figuur 3.1*

#### 1968 Head mounted display (HMD), I. Sutherland

De realisatie van de eerste bril waarop digitale informatie geprojecteerd wordt op het glas van de bril. Het is het eerste systeem dat de naam augmented reality mag dragen doordat het geprojecteerde beeld in het werkelijke beeld, afhankelijk is van digitale input en deze gestuurd wordt doordat de gebruiker zich verplaatst.

#### 1992 De term “augmented reality”, T. Caudell en D. Mizell

De term “augmented reality” (AR) wordt ingevoerd. Deze term refereert naar een visualisatietechniek, waarbij door computer verkregen beelden fungeren als een laag (*overlay*) bovenop (*het beeld van*) de echte wereld.

#### 1993 Navigatiesysteem voor blinden en slechthzienden, J. Loomis, R. Golledge en R. Klatzky

Hoewel dit navigatiesysteem gestuurd werd door GPS, gaf het géén virtuele weergave. De augmented reality in dit systeem betreft geluid. De gebruiker wordt op locatie gestuurd door gesproken instructies. Het geluid wordt door middel van een ruimtelijk model geregeld (*vergelijkbaar met een surround systeem*), zodat de gebruiker niet alleen weet dat er “iets” in de buurt is, maar ook in welke richting.

#### 1996 Augmented reality gepositioneerd door 2D matrix markers, J. Rekimoto

Een marker waaraan een model gedefinieerd is, wordt herkend in het camerabeeld. In real-life wordt de augment in het beeld gepositioneerd ten opzichte van deze marker. In hoofdstuk 9. Beeldherkenningstechniek wordt deze techniek van augmented reality nader omschreven.

**1997 1e mobiele apparatuur voor augment, S. Feiner**

Niet kijkend naar de prestatieverschillen is het apparaat zeer te vergelijken met het head mounted displaysysteem dat I. Sutherland in 1968 heeft gerealiseerd. Het verschil is de positiebepaling en het feit dat het apparaat mobiel (*voorzien van een accu*) is.

**1997 Eerste mobiele telefoon met camera, Philippe Kahn**

**1999 Worldboard concept, Jim Sphorer**

Worldboard is het idee van Jim Sphorer om via internet geogerefereerde informatie door middel van draadloze communicatie te voorzien aan draagbare mobiele apparaten.

**1999 Eerste mobiele telefoon met GPS, Benefon Esc! NT2002**

**2000 GPS voor iedereen**

Bij het Amerikaanse NAVSTAR-GPS (*NAVigation Satellite Time And Ranging – Global Positioning System*) is vanaf 1 mei 2000 de Selective Availability (*opzettelijk gegenereerde fout*) op nul gezet, zodat ook burgers gebruik kunnen maken van een nauwkeurig GPS-signaal. Deze fout werd met name in de Golfoorlog ingezet, zodat de vijand hier géén gebruik van kon maken. In 2007 is door het Amerikaanse ministerie van defensie aangegeven, toekomstige GPS-III satellieten niet meer van een Selective Availability optie te voorzien.

**2001 Augmented realityserver, Jürgen Freund**

De eerste augmented realityserver. De PDA ontvangt via een draadloze verbinding met een server, digitale informatie die in het camerabeeld geprojecteerd wordt.

**2001 Interactieve augmented reality (multi-user), G. Reitmayr, D. Schmalstieg**

Hierbij kan een spel in augmented reality tussen verschillende gebruikers worden gespeeld. De 3D augments zijn dynamisch en kunnen door de gebruikers verplaatst worden.

**Tot heden**

Vanaf 2001 tot heden zijn er géén ontdekkingen van betekenis gedaan. Wel zijn de bestaande technieken verfijnd, doordat tegenwoordig de prestaties van bruikbare apparatuur een stuk hoger liggen én voor iedereen toegankelijk zijn.

Tegenwoordig zijn prestaties van bruikbare apparaten hoog genoeg om vloeiend augments op weer te geven. De apparaten zelf zijn veelal smartphones of tabletcomputers die bij een groot (*snelgroeiend*) publiek in bezit zijn.

De ontwikkelingen van de tegenwoordige augmented reality richt zich dan ook veelal op entertainment voor een groot publiek (*games, navigatie, animatie etc.*). De techniek die beschreven wordt in deze scriptie betreft vooral locatiegebonden augmented reality.

*Bron: (Christian Doppler Laboratory, (n.d.)), (Wikipedia.org, 2012)*



## 4. INVENTARIS AR-SOFTWARE

*Hoewel het gebruik van Layar al door Witteveen+Bos was vastgesteld is er alsnog een inventarisatie geweest van het aanbod van augmented reality softwarepakketten. Deze inventaris is opgesteld om vast te stellen dat de goede keuze wordt gemaakt, maar ook om te ondervinden of er géén technieken over het hoofd worden gezien. De resultaten zoals vermeld in tabel 4.1 zijn beperkt gehouden tot de onderstaande eigenschappen:*

- Besturingssysteem / platform
- Methodiek van positiebepaling / oriëntatie
- Visualisatiemogelijkheden

### 4.1 Verklaring software-eigenschappen

Besturingssysteem: samenhangend geheel van systeemprogramma's, waarmee alle programma's op het apparaat mee worden aangestuurd en uitgevoerd.

Positiebepaling: methodiek voor het verkrijgen van de apparaatpositie ten opzichte van het te projecteren beeld.

Beeldherkenning: methodiek van relatieve positiebepaling. Het waargenomen beeld wordt herkend met een sterk overeenkomend beeld in de bibliotheek. Het te projecteren model is met de publicatie gerefereerd aan en ten opzichte van dit beeld.

GPS: in de tabel in §4.2 van dit hoofdstuk betreft "GPS": de verzamelnaam voor de bepaling van absolute positie en oriëntatie door middel van GPS in combinatie met een gyroscoop, magnetometer (*kompas*) en accelerometer (*verplaatsings- / versnellingsmeter*).

#### **Weergavecontent**

Labels: tekstuele data waaronder te verstaan: hyperlinks naar e-mail, websites, (*telefoon*)nummers, documenten/downloads.

2D: afbeeldingen of andere "platte" modellen die absoluut of relatief geogerefereerd zijn.

3D: driedimensionale augments die absoluut of relatief geogerefereerd zijn.

## 4.2 Inventarisatie AR-software

Met het inventariseren van de software voor augmented reality, is geconcludeerd dat het pakket “Layar” uniek is, wanneer men 3D-augments wil projecteren door middel van GPS-positionering. Echter is AR-media met zijn tijd meegegaan, sinds 24 september 2012 (*betreft tevens de afrondingsfase van deze scriptie*) kan ook AR-media 3D-augments projecteren door gebruik te maken van GPS-positionering. In de scriptie wordt met deze laatste ontwikkeling géén rekening gehouden.

In de onderstaande tabel ziet u de resultaten van de volledige inventaris “augmented realitysoftware voor mobiele apparatuur”.

Software	Platform	Positiebepaling	
		Beeldherkenning	GPS
Layar	iOS / Android / BlackBerry 7 OS / Symbian	labels / 2D	labels / 2D / 3D
Junaio SDK	iOS / Android	labels / 2D / 3D	-
Lumar	Symbian	labels / 2D / 3D	-
Yelp	iOS	-	labels
Adobe Flash Pro	Android, Symbian, BlackBerry 7 OS	labels / 2D / 3D	labels / 2D / 3D
Wikitude	alle smartphones	labels / 2D / 3D	labels
Aurasma	iOS / Android	3D	-
Sticktu	iOS / Android	2D	-
String	iOS	2D / 3D	-
Qualcomm	iOS / Android	-	labels
Total Immersion	iOS / Android	2D / 3D	labels
AR-media	iOS / Android	Labels / 2D / 3D	sinds 24 september '12

Tabel 4.1

## 4.3 Conclusie inventaris AR-software

Bij Witteveen+Bos is een voorwaarde dat de applicatie op het mobiele apparaat een absolute positie en oriëntatie moet kunnen bepalen door middel van GPS in combinatie met een gyroscoop, magnetometer (*kompas*) en accelerometer verplaatsing / versnellingsmeter.

Een tweede voorwaarde is dat driedimensionale modellen weergegeven kunnen worden wanneer gebruik wordt gemaakt van absolute positiebepaling.

Als gevolg van de bovengenoemde eisen blijven er twee opties open, namelijk Layar en Adobe Flash Pro (*zie paragraaf A.4 Layar versus Adobe Flash Pro*).

Wanneer positiebepaling door middel van GPS géén mogelijkheid is en de positionering alleen gedaan wordt op basis van beeldherkenning, dan is de applicatie AR-media de beste oplossing. AR-media heeft in tegenstelling tot de andere beeldherkenningssoftware, de algoritmes zo ingesteld dat het zelfs zware (*gedetailleerde*) modellen goed en zonder schokken op een mobiel apparaat weer kan geven.

## 4.4 Layar versus Adobe Flash Pro

De verschillen tussen de software “Layar” en “Adobe Flash Pro” zitten grotendeels in de:

- ontwikkeling en onderhoud;
- kosten aanschaf;
- compatibiliteit;
- commercie;
- JSON blijft, Flash verdwijnt.



versus



### 4.4.1 Ontwikkeling en onderhoud

Het nadeel van Adobe Flash Pro is dat de applicatie om augments weer te geven door de gebruiker zelf gemaakt dient te worden. ActionScript: de codetaal die in Adobe Flash Pro gebruikt dient te worden, wordt als zeer complex ervaren. Daar bovenop is de verwachting dat HTML5 een goede vervanger wordt voor datgene wat nu in Flash wordt weergegeven. HTML(5) is opensource en de internetstandaard voor het weergeven van websites en wordt door alle webbrowsers (*ook smartphones*) ondersteund. Flash zal naar verwachting hierdoor binnen niet al te lange tijd ophouden te bestaan.

Wanneer zelf een applicatie wordt gemaakt, moet deze ook zelf onderhouden worden. Dat wil zeggen: wanneer het besturingssysteem van de mobiele apparatuur verandert door een update of upgrade is het waarschijnlijk dat de huidige versie van de eigen applicatie niet, dan wel slechter fungeert.

Wanneer Layar wordt gebruikt, kan er worden teruggevallen worden de specialisten van Layar die continu bezig om de applicatie te verbeteren en te onderhouden.

### 4.4.2 Compatibiliteit

Het grootste nadeel van Adobe Flash Pro is dat de modellen weergegeven worden als flash-content. Het besturingssysteem van Apple, iOS ondersteunt géén flash-content. Aangezien Apple iOS na Google Android in mobiele apparatuur het meest toegepaste besturingssysteem betreft, is het gewenst dat (*net als bij Layar*) de augmented realitysoftware op tenminste deze beide besturingssystemen fungeert.

### 4.4.3 Commercie

Wanneer er een eigen applicatie geschreven wordt, heeft dit het voordeel dat de gehele inhoud zelf bepaald kan worden. Eigen interface, eigen content en de garantie dat er géén ongewenste reclame wordt vertoond.

Het nadeel van een eigen applicatie neemt wel met zich mee dat deze (*zeker vlak na publicatie*) onbekend en moeilijk vindbaar is in de Android Market of Apple Appstore.

#### 4.4.4 Interface

De werking van Layar is vergelijkbaar met die van een internetbrowser. In Layar wordt eerst naar een bibliotheek met augments gezocht. Wanneer deze geselecteerd is, kan met de GIS-functie “zoeken op locatie” of beeldherkenning de modellen worden weergegeven. Bij een applicatie gemaakt met Adobe Flash Pro, zou alle content (*augments*) in de applicatie zelf verwerkt moeten worden.

### 4.5 Beeldherkenning diverse AR-software

Voor alle software waar beeldherkenning zorgt voor de modelpositionering, geldt dat het te herkennen beeld een sterk contrasterend patroon betreft. Het beeld wordt omgezet naar een binair beeld (*zwart-wit*) waarop de software zoekt naar patronen die gedefinieerd zijn in een database.

Layar kan augments zoeken door een beeld te scannen. Tenminste een gedeelte van hetzelfde beeld als waaraan de augment is gelinkt dient in de database gevonden te worden.

Het projecteren van augments door middel van beeldherkenning is bij Layar alleen mogelijk wanneer er een 95% match in het beeld is gevonden en de camera uiterst stil wordt gehouden. De augments betreffen in dit geval labels, afbeeldingen of andere platte media waar acties (*hyperlinks, audio, video*) aan gekoppeld zijn, maar géén 3D modellen.

Doordat Layar géén 3D modellen ondersteunt in haar beeldherkenningstechniek, geeft de beeldherkenningstechniek van Layar géén meerwaarde voor datgeen (*het versterken of verkrijgen van de augmentpositie ten tijde van een slecht GPS-signaal*), dat voor Witteveen+Bos gewenst is. *Bron: (Layar, 2011)*

De AR-software die in de inventaris het beste naar voren komt qua beeldherkenningstechniek betreft AR-media. Deze software positioneert de augment met een hoge nauwkeurigheid relatief ten opzichte van de marker waaraan de augment gedefinieerd is. Zelfs zware (*gedetailleerde*) modellen worden in een hoge frequentie (*ca. 30 frames/seconde*) gerendeerd, wanneer de gebruiker zich ten opzichte van de marker verplaatst.

*In hoofdstuk 9 wordt de techniek “beeldherkenning” concreet beschreven.*



## 5. INVENTARIS MOBIELE APPARATUUR

Er is voor gekozen een inventarisatie te doen op mobiele apparatuur die geschikt is voor het uitvoeren van de Layarapplicatie. Zo wordt inzichtelijk hoe groot de doelgroep zou kunnen worden wanneer voor Layar gekozen wordt. De mobiele apparatuur moet minimaal voorzien zijn van een aantal verplichte hardwarecomponenten (§ 5.2) én moet fungeren op een bepaald type besturingssysteem (§5.1).

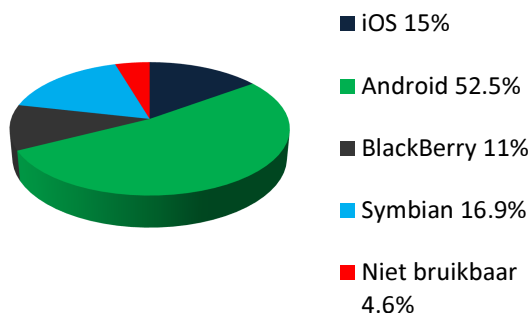
### 5.1 Voorwaarden besturingssysteem

Layar is een applicatie die ondersteund wordt in de volgende besturingssystemen:

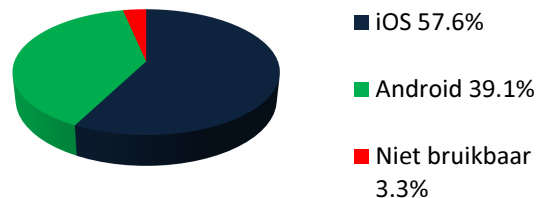
- Android (Google);
- 7 OS (BlackBerry);
- iOS (Apple);
- Symbian (Nokia).

Onderstaand de grafieken met daarin cijfers van het vierde kwartaal van 2011 over de verdeeldheid van besturingssystemen voor smartphones en tabletcomputers.

Besturingssysteem smartphones



Besturingssysteem tabletcomputers



Bron: Gartner inc. cijfers van het vierde kwartaal 2011

### 5.2 Voorwaarden hardware

Met Layar is het mogelijk augments in het scherm te projecteren mits de apparatuur voorzien is van alle van de onderstaande componenten:

- |                                  |  |
|----------------------------------|--|
| ▪ Camera                         | Verkrijgen van het werkelijke beeld                            |
| ▪ Beeldscherm                    | Weer te geven beeld ( <i>real-life + augment</i> )             |
| ▪ GPS antenne                    | Positiebepaling in X,Y,Z                                       |
| ▪ Magnetometer ( <i>kompas</i> ) | Rotatie ten opzichte van het noorden ( $\Omega$ )              |
| ▪ Gyroscop                       | Rotatiehoeken ( <i>kantel, helling</i> ) ( $\varphi, \kappa$ ) |
| ▪ Accelerometer                  | Verplaatsing en versnelling ( $m/s^2$ )                        |

Bron: (Layar, 2012)

### 5.3 Berekening performance mobiele apparatuur

Van processoren is bekend dat de maximale rekenkracht wordt bepaald door de bus-frequentie (*niet de kloksnelheid*) van de processorunit en het aantal bits dat de chip heeft. De verwerkingssnelheid kan als volgt berekend worden:

Het aantal bits van een processor bepaalt het aantal cyclussen per frequentie:

$$8 \text{ bits} \times n \text{ Hz} \times 4 \text{ cyclussen} \times \frac{1 \text{ bit}}{8 \text{ bytes}} = \frac{\text{bytes}}{\text{seconde}}$$

$$32 \text{ bits} \times n \text{ Hz} \times 6 \text{ cyclussen} \times \frac{1 \text{ bit}}{8 \text{ bytes}} = \frac{\text{bytes}}{\text{seconde}}$$

$$64 \text{ bits} \times n \text{ Hz} \times 9 \text{ cyclussen} \times \frac{1 \text{ bit}}{8 \text{ bytes}} = \frac{\text{bytes}}{\text{seconde}}$$

Vergelijking 5.1

Dit betreft de hoogste haalbare prestatie van de processor. De effectieve rekenkracht is afhankelijk van andere (*altijd lager scorende*) componenten als grafisch-, RAM- en opslaggeheugen.

Om een presentatie-indicatie te vormen van de inventaris, is een vergelijking gemaakt met de hardwarecomponenten: "processor" en werkgeheugen (*RAM*). De resultaten van de inventarisatie bruikbare mobiele apparatuur gesorteerd op verwerkingssnelheid (*van hoog naar naar*) zijn te vinden in de bijlagen: B1(a) en B1(b).

### 5.4 Conclusies inventaris mobiele apparatuur

- Uit deze inventaris is gebleken dat de Apple iPad 4G het snelst presterende apparaat is waarop de Layaapplicatie uitgevoerd kan worden.
- Wereldwijd voldoet 95,4% van alle telefoons en 96,7 % van alle tabletcomputers aan de vereiste besturingssystemen (*dit zijn ook apparaten zonder de vereiste hardwarecomponenten*).

## 6. INVENTARIS SUPPORT & DIENSTEN

Hoewel Layar uniek is in het projecteren van augments in combinatie met een positionering door middel van GPS, is wereldwijd gezien het support-aanbod nihil.

Organisatie	Land	Specialiteit
Talking Dog Studios	Canada	Beeldherkenning
Augmented Reality BlackSuit Labs	Canada	Websites
Infinity Layar	Israël, Singapore	Layar platform
Atware Inc.	Japan	Programmeren, Ontwikkeling Layar
Nakano AI System Co. Ltd.	Japan	Websites
DnL Productions Inc.	Korea, Spanje	Beeldherkenning
Clockwork	Nederland	Websites
Menno Bieringa	Nederland	Website, 3D, Beeldherkenning
Zaphyrion	Nederland	Websites, Internetapplicaties
DPI Animation House	Nederland	Websites, Beeldherkenning, Animatie
One Shoe Mobile	Nederland	Websites
Innovatia – Emerging Technologies	Spanje	Animatie
SIGNO, Ingeniería del Territorio	Spanje	CMS, Websites
B_cultura	Spanje	Marketing, Kunst, Website
bigBlgBang	Verenigde Staten	Websites, AR-marketing
DataVision	Verenigde Staten	Beeldherkenning, Animatie
Second Site	Verenigde Staten	Websites
Different Signal	Zweden	3D, Beeldherkenning

Tabel 6.1

*De inhoud van de tabel betreffen bedrijven en personen die Layar-geautoriseerd zijn om support en diensten te verlenen. De tabel is gevormd op d.d. 27 april 2012, de organisatienamen en bijhorende standplaatsen zijn vermeld op [layar.com](http://layar.com), de specialiteiten op de website van de betreffende organisatie. Bron: (Layar, 2012)*

Doordat Witteveen+Bos geïnteresseerd is in het onderdeel “3D”, zouden (*wanneer support gewenst is*) de organisaties “Menno Bieringa” en “Different Signal” benaderd kunnen worden.

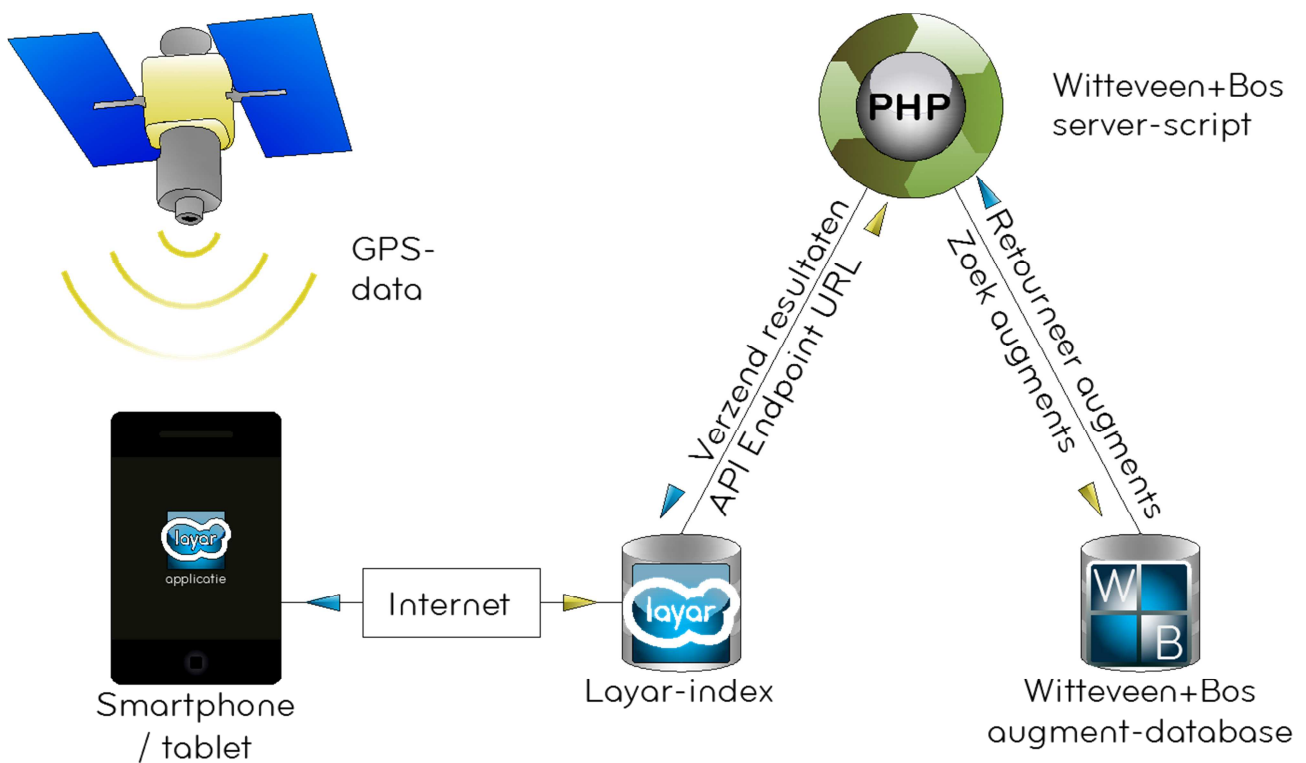
De overige organisaties zijn (*nagenoeg allemaal*) veel sterker georiënteerd op het onderdeel Layar Vision (*Beeldherkenning voor drukwerk*), omdat commercieel gezien met Layar Vision een veel groter publiek bereikt kan worden. De afstudeeropdracht is afgebakend voor “augmented reality voor de bouw & infra”, waar Layar Vision voor de projectie van geogerefereerde 3D modellen géén meerwaarde in betekent.

## 7. INRICHTEN LAYARSERVER

### 7.1 Algemene werking van Layar

Om een Layarserver te realiseren, dient duidelijk te zijn hoe de communicatiestromen verlopen, welke informatie beschikbaar is, en hoe deze gebruikt kan worden. Onderstaand (in figuur 7.1) een overzicht van de communicatiestromen.

Wanneer de Layarapplicatie gestart is, zendt Layar (*smartphone / tabletcomputer*), wanneer de geografische locatie (*GPS-data*) bekend is, een vraag uit om de lokaal gelegen augments te zoeken. Deze augments dienen bij Layar geregistreerd te zijn (*Layarindex*) om gevonden te kunnen worden. Het PHP-script regelt uiteindelijk welke augments uit de augmentdatabase weergegeven mogen worden en vanaf welk oogpunt de modellen geprojecteerd moeten worden op de display van het mobiele apparaat.



Figuur 7.1

Om een PHP-serverscript te programmeren dient de vraagstelling van de Layarapplicatie duidelijk te zijn. Hieronder een voorbeeld van een vraagstelling die door het mobiele apparaat uitgezonden wordt:

```
http://server.domein.nl/layar/api.php?lang=en&countryCode=NL&lon=6.41738891602&userId=6f85d06929d160a7c8a3cc1ab4b54b87db99f74b&version=6.2&radius=5000&lat=52.1756160474&layerName=wblayar1&accuracy=20
```

Deze vraagstelling is opgebouwd uit de volgende componenten:

De internetlocatie van het te benaderen PHP script:

```
http://server.domein.nl/layar/api.php?
```

De taal van de gebruiker. De Layaapplicatie wordt alleen in het Engels ondersteund, deze waarde betreft dan ook altijd:

```
lang=en&
```

Het land van waaruit de betreffende augment opgevraagd wordt:

```
&countryCode=NL
```

De longitude oftewel de lengtegraad in het coördinatenstelsel WGS84:

```
&lon=6.41738891602
```

De bij Laya geregistreeerde gebruiker:

```
&userId=6f85d06929d160a7c8a3cc1ab4b54b87db99f74b
```

Versienummer van de Layaapplicatie

```
&version=6.2&
```

De zoekradius in meters ten opzichte van de positie van het mobiele apparaat:

```
&radius=5000
```

De latitude oftewel de breedtegraad in het coördinatenstelsel WGS84:

```
&lat=52.1756160474
```

De naam van de weer te geven layer:

```
&layerName=wblayar1
```

De door de apparaat berekende nauwkeurigheid van de positiebepaling in meters:

```
accuracy=20
```

Dit zijn voldoende gegevens om met behulp van een PHP-script bevestigingen en berekeningen te doen. *Bron: (Laya, 2011)*

## 7.2 Database ontwerp SQL

Layar is een programma dat alleen augments kan visualiseren, de fysieke opslag van deze modellen dient elders ondergebracht te worden. Hiervoor is een database met bijbehorende programmeercode samengesteld en toegepast.

De Layarapplicatie krijgt toegang tot de database door middel van een API Endpoint URL, dit betreft een PHP gecodeerd script dat verwijst naar de SQL-database. Om dit script te schrijven dient eerst bekend te zijn hoe de database is opgebouwd.

API: Application Programming Interface. Een software interface tussen een programmeertaal en een applicatie, die een transparante toegang biedt tot het bestandsysteem, netwerkbronnen, etc.

### 7.2.1 Algemene instellingen

Opstellen van voorwaarden voor het genereren van de SQL-database.

Tenzij gedefinieerd, worden er standaard géén nulwaarden aan records toegekend. Sleutelkolommen worden achteraf / na het genereren van de tabellen toegekend.

```
SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
SET FOREIGN KEY CHECKS = 0;
```

*Dit is ter voorkoming dat koppelingen tussen de tabellen onderling niet gemaakt kunnen worden: Een onderliggende "child-table" moet vanaf de "parent-table" gekoppeld worden, andersom is niet toegestaan.*

### 7.2.2 POI, Places of Interest

Als eerste wordt de POI-tabel aangemaakt, deze tabel betreft de hoofdtabel waaraan alle komende tabellen direct of indirect aan gekoppeld worden.

De belangrijkste informatie hierin betreffen de coördinaten (lat, lon) en de sleutelattributen naar de overige tabellen.

De overige attributen betreffen metadata, waarmee het model in de Layarapplicatie opgezocht kan worden.

```
CREATE TABLE IF NOT EXISTS `POI` (
  `id` varchar(255) NOT NULL,
  `footnote` varchar(150) DEFAULT NULL,
  `title` varchar(150) NOT NULL,
  `lat` decimal(13,10) DEFAULT NULL,
  `lon` decimal(13,10) DEFAULT NULL,
  `imageUrl` varchar(255) DEFAULT NULL,
  `description` varchar(150) DEFAULT NULL,
  `biwStyle` enum('classic','collapsed') DEFAULT
'classic',
  `doNotIndex` tinyint(1) DEFAULT '0',
  `showSmallBiw` tinyint(1) DEFAULT '1',
  `showBiwOnClick` tinyint(1) DEFAULT '1',
  `poiType` enum('geo','vision') NOT NULL DEFAULT 'geo',
  `iconID` int(11) DEFAULT NULL,
  `objectID` int(11) DEFAULT NULL,
  `transformID` int(11) DEFAULT NULL,
  `layerID` int(11) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `iconID` (`iconID`),
  KEY `objectID` (`objectID`),
  KEY `transformID` (`transformID`),
  KEY `layerID` (`layerID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

### 7.2.3 Layer

De “Layer”-tabel regelt het herladen van een model.

Zoals hiernaast is beschreven:

*“Wanneer men zich meer dan 6 m of langer dan 15 sec niet ( $\leq 6$  m) verplaatst, dan wordt het model opnieuw gepositioneerd ten opzichte van de actuele GPS-waarneming van het mobiele apparaat.”*

```
CREATE TABLE IF NOT EXISTS `Layer` (
  `layer` varchar(255) NOT NULL,
  `refreshInterval` int(10) DEFAULT '15',
  `refreshDistance` int(10) DEFAULT '6',
  `fullRefresh` tinyint(1) DEFAULT '1',
  `showMessage` varchar(255) DEFAULT NULL,
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `biwStyle` enum('classic','collapsed') DEFAULT 'classic',
  PRIMARY KEY (`id`),
  UNIQUE KEY `layer` (`layer`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=6 ;
```

### 7.2.4 POIAction

In deze tabel kan een augment voorzien worden van diverse media die ondersteund worden door het mobiele apparaat betreffende: audio, video, telefoonnummer, website, e-mailadres etc.

1 record in de “POIAction” tabel betreft 1 type content met één bijbehorend bestand of tekst, dat aan meerdere augments gekoppeld kan worden.

```
CREATE TABLE IF NOT EXISTS `POIAction` (
  `poiID` varchar(255) NOT NULL,
  `label` varchar(30) NOT NULL,
  `uri` varchar(255) NOT NULL,
  `autoTriggerRange` int(10) DEFAULT NULL,
  `autoTriggerOnly` tinyint(1) DEFAULT '0',
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `contentType` varchar(255) DEFAULT 'application/vnd.layar.internal',
  `method` enum('GET','POST') DEFAULT 'GET',
  `activityType` int(2) DEFAULT NULL,
  `params` varchar(255) DEFAULT NULL,
  `closeBiw` tinyint(1) DEFAULT '0',
  `showActivity` tinyint(1) DEFAULT '1',
  `activityMessage` varchar(255) DEFAULT NULL,
  `autoTrigger` tinyint(1) NOT NULL DEFAULT '0',
  PRIMARY KEY (`id`),
  KEY `poiID` (`poiID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=7 ;
```

### 7.2.5 Icon

Tabel met een verwijzing naar een afbeelding dat fungeert als pictogram voor de betreffende augment.

```
CREATE TABLE IF NOT EXISTS `Icon` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `url` varchar(255) DEFAULT NULL,
  `type` int(11) DEFAULT '0',
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=3 ;
```

### 7.2.6 Object

Tabel met de verwijzingen naar de fysieke bestanden die fungeren als augments.

```
CREATE TABLE IF NOT EXISTS `Object` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `contentType` varchar(255) NOT NULL,
  `url` varchar(255) NOT NULL,
  `reducedURL` varchar(255) DEFAULT NULL,
  `size` float(15,5) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=6 ;
```

### 7.2.7 Transform

Mocht een augment in het veld niet goed gepositioneerd zijn, dan kan de plaatsing en rotatie in deze tabel gemuteerd worden.

```
CREATE TABLE IF NOT EXISTS `Transform` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `rel` tinyint(1) DEFAULT '0',
  `angle` decimal(5,2) DEFAULT '0.00',
  `rotate_x` decimal(2,1) DEFAULT '0.0',
  `rotate_y` decimal(2,1) DEFAULT '0.0',
  `rotate_z` decimal(2,1) DEFAULT '0.0',
  `translate_x` decimal(5,1) DEFAULT '0.0',
  `translate_y` decimal(5,1) DEFAULT '0.0',
  `translate_z` decimal(5,1) DEFAULT '0.0',
  `scale` decimal(12,2) NOT NULL DEFAULT '1.00',
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=4 ;
```



### 7.2.8 Koppelingen

Als laatste worden alle tabellen met elkaar gekoppeld.

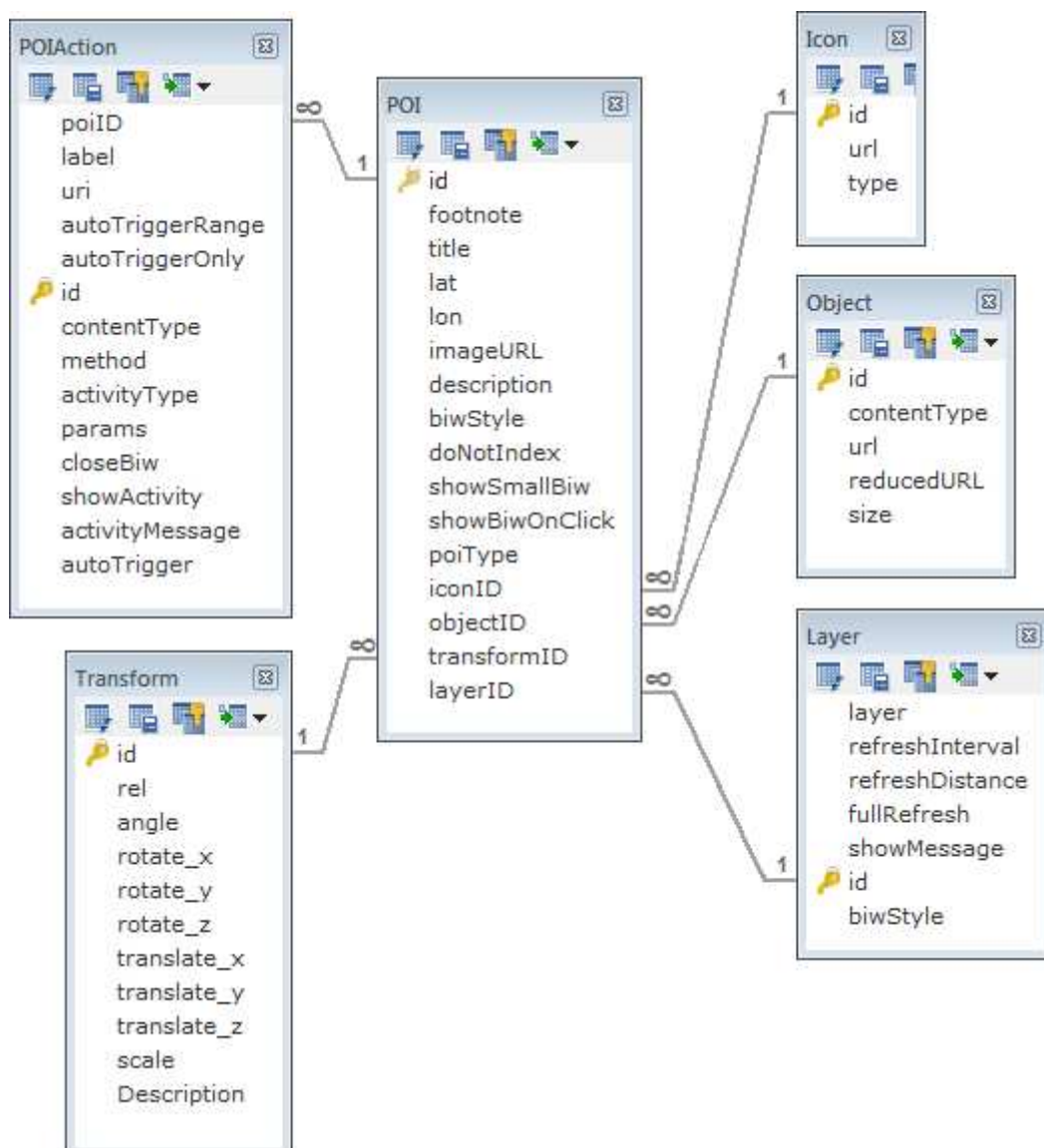
Principe SQL-database, bron: (Din, 1994)

```
ALTER TABLE `POI`
  ADD CONSTRAINT `POI_ibfk_8` FOREIGN KEY (`layerID`)
  REFERENCES `Layer` (`id`),
  ADD CONSTRAINT `POI_ibfk_12` FOREIGN KEY (`iconID`)
  REFERENCES `Icon` (`id`),
  ADD CONSTRAINT `POI_ibfk_11` FOREIGN KEY (`objectID`)
  REFERENCES `Object` (`id`),
  ADD CONSTRAINT `POI_ibfk_7` FOREIGN KEY (`transformID`)
  REFERENCES `Transform` (`id`);

ALTER TABLE `POIAction`
  ADD CONSTRAINT `POIAction_ibfk_1` FOREIGN KEY (`poiID`)
  REFERENCES `POI` (`id`);
```

### 7.2.9 Resultaat

Wanneer alle codeblokken in de paragrafen 7.2.1 t/m 7.2.8 in één reeks als SQL-query in een database wordt geïmporteerd, dan is de volgende databasestructuur zoals in figuur 7.2 weergegeven, gevormd.



Figuur 7.2

## 7.3 Programmering PHP

In deze paragraaf wordt het proces beschreven hoe de vraagstelling en beantwoording van de applicatie Layaar vanuit het mobiele apparaat naar de server verloopt. De code hiervoor is geschreven in PHP. Het script is zo geschreven dat het alleen augments retourneert binnen een door de gebruiker opgegeven straal met een maximum van vijftig.

PHP; PHP Hypertext Preprocessor is een voor de gebruiker onzichtbare codetaal. Het script wordt direct door een computer benaderd waarnaar de server bevragingen doet op de SQL-database. Bij Layaar heet deze directe link een “API Endpoint URL” (zie § 7.1 Voorbereiding).

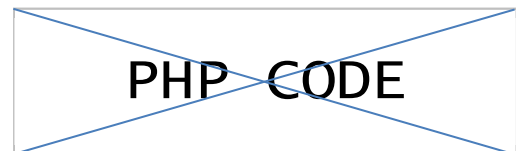
Met de input van de gebruiker en de gevonden resultaten in de SQL database worden berekeningen gevoerd, waarop de gebruiker alleen het resultaat hiervan als antwoord retour krijgt van de server.

In dit geval betreffen de computers mobiele apparaten (*smartphones en tabletcomputers*), waarop de applicatie Layaar uitgevoerd kan worden.

In de volgende paragrafen wordt beschreven hoe het PHP-script is samengesteld en wat de werking betreft. De code is zelf geschreven, maar bevat diverse codeblokken, afkomstig van Xuan Wang ([Layaar.com](http://Layaar.com)).

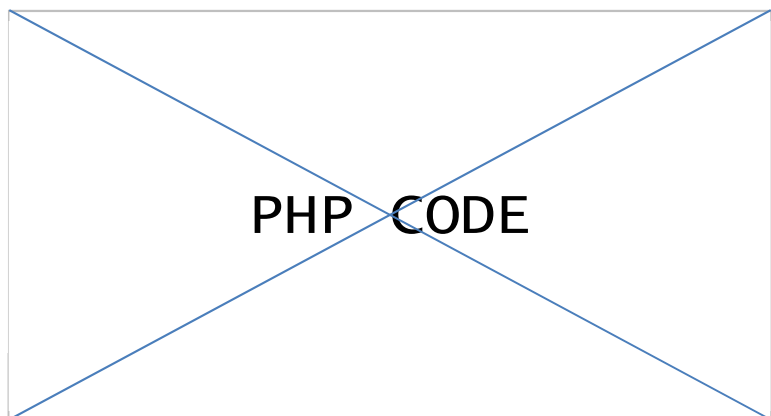
### 7.3.1 API Endpoint URL

Om toegang met de SQL-database te kunnen krijgen zijn de toegangsgegevens hiervan vereist. Hiernaast worden de toegangsgegevens gedefinieerd.



### 7.3.2 Controle

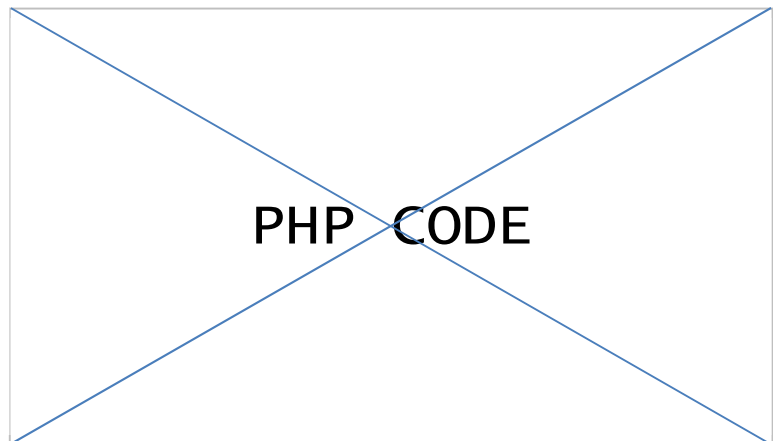
Alle benodigde parameters worden door het script gecontroleerd en binnengehaald in een geassocieerde matrix.



### 7.3.3 Verbinding SQL-database

Er wordt verbinding gemaakt met de database met behulp van de gedefinieerde toegangsgegevens.

De database kan nu via dit PHP-script door derden benaderd worden zonder dat zij de database kunnen muteren noch inzicht hebben in de toegangsgegevens.



### 7.3.4 Data ordening

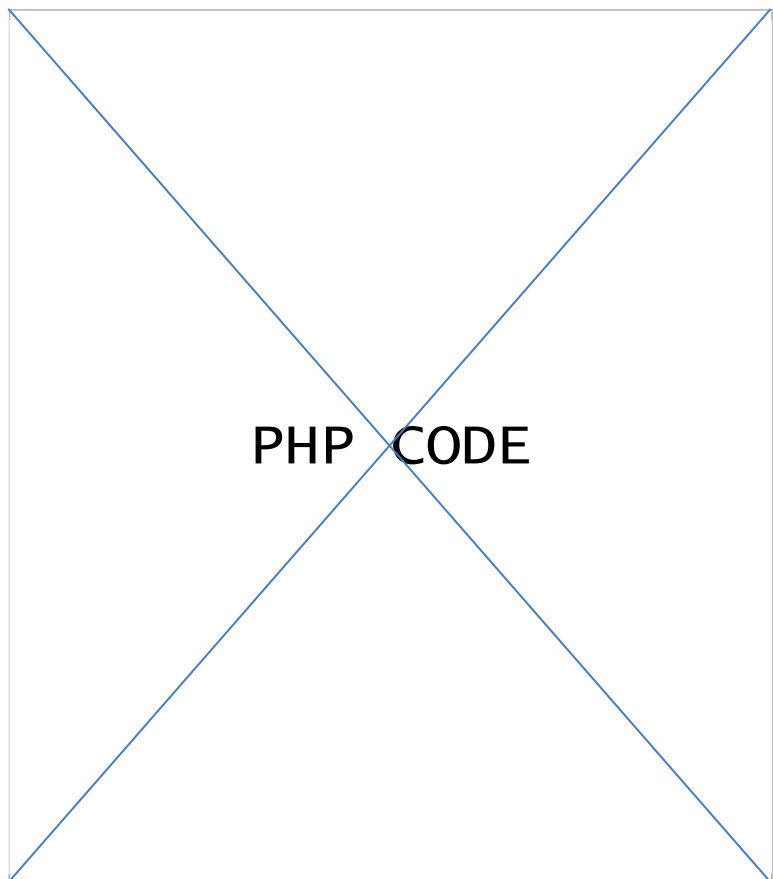
In dit deel wordt de ontvangen code gereed gemaakt om hiermee berekeningen uit te kunnen voeren.

De gegevens, verkregen van het mobiele apparaat, worden getransponeerd van rij naar kolommen.

(*Decimale*) getallen worden ten behoeve van berekeningen omgezet naar integrale getallen.

Alle waarden worden in een nieuwe matrix geschreven.

Kleine integrale waarden (0 en 1) naar waar of niet waar:  
0 = FALSE, 1 = TRUE



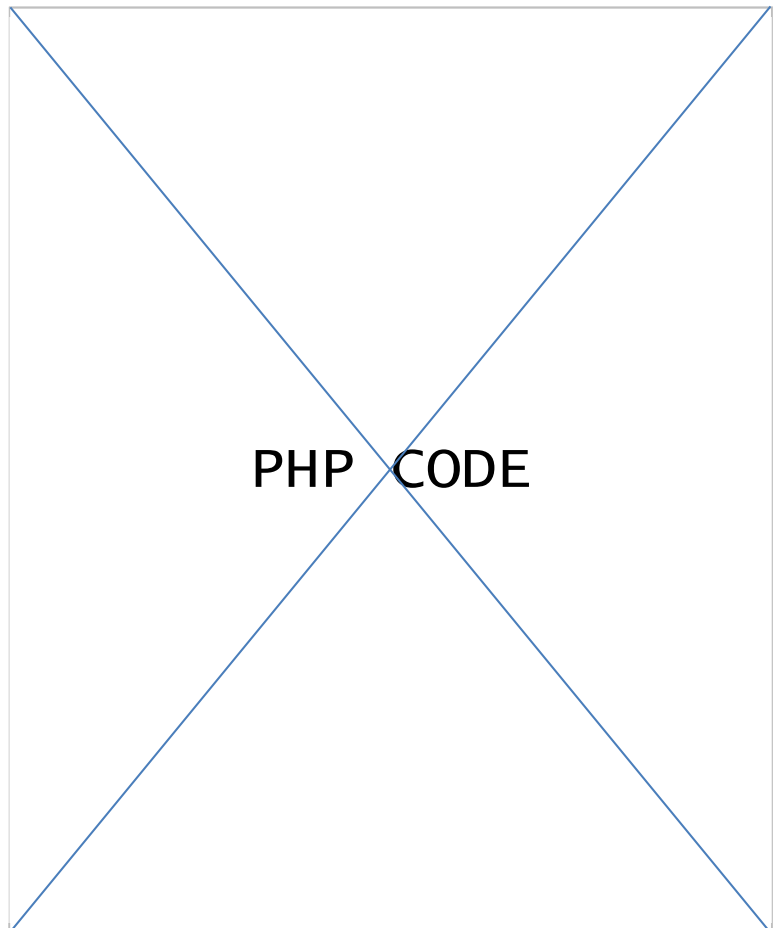
Voor dit PHP-script maakt het niet uit in welke volgorde er door de tabellen gezocht moet worden. In de vraagstelling van het mobiele apparaat naar de API Endpoint URL wordt al aangegeven onder welke groepnaam (*variabele*) de augment(s) geregistreerd dienen te zijn. Wanneer meerdere augments in dezelfde groep voorkomen, kunnen meerdere augments tegelijk in het scherm geprojecteerd worden.

### 7.3.5 Toevoegen diverse media

Met behulp van SQL wordt in de database gezocht naar diverse media in de tabel "POIAction".

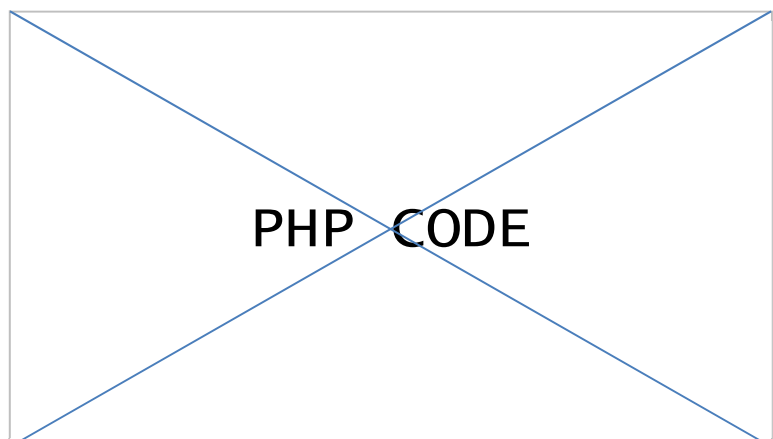
De resultaten worden alleen opgenomen en klaargezet wanneer deze voldoen aan de onderstaande controle:

Het geregistreerde type media "activityType" moet overeenkomen met de daadwerkelijke media.



### 7.3.6 Toevoegen iconen

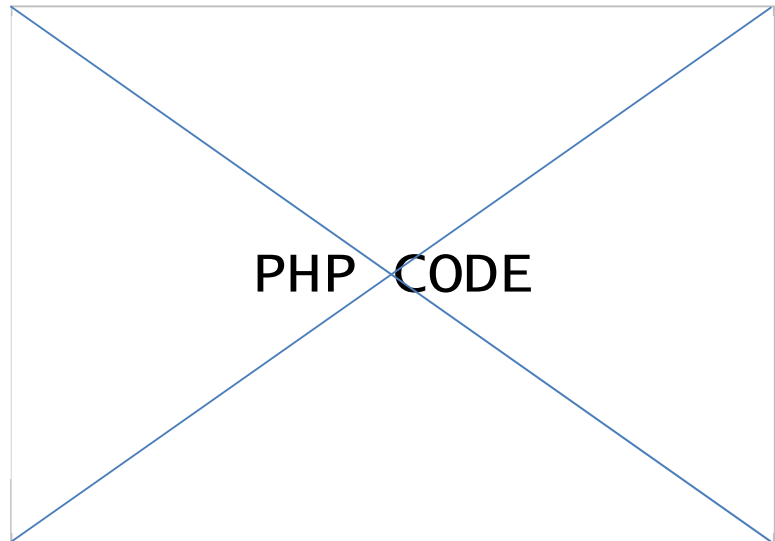
Een voor Layar verplicht onderdeel is dat de POI voorzien moet zijn van een icoon.



### 7.3.7 Toevoegen augment

Het model dat in het gefilmde beeld weergegeven dient te worden (*de augment*) wordt in de SQL-database opgezocht en klaargezet.

Een augment is niet per definitie een 3D-model, het zou ook een geogerefereerde afbeelding kunnen betreffen. Of in de simpelste vorm een tekstueel label.

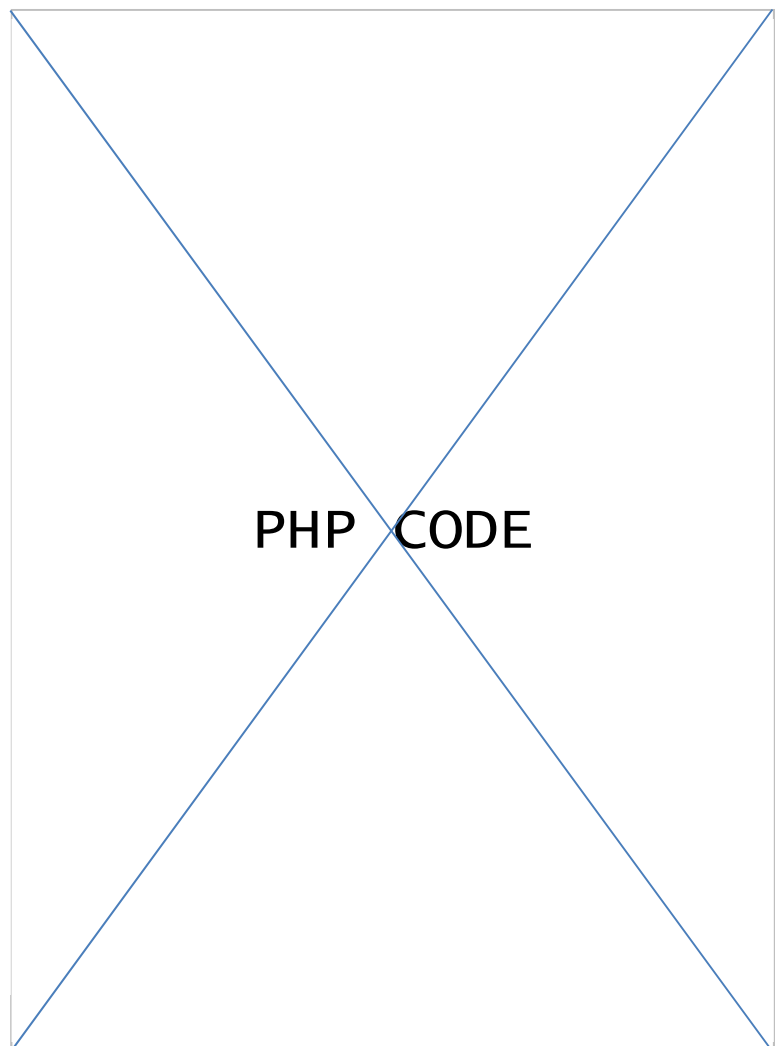


### 7.3.8 Inpassing augment

Wanneer een augment in de database opgenomen is, is de grootte, rotatie en positionering al gedefinieerd.

Met behulp van deze tabel kan de augment getransformeerd worden, wanneer deze in het veld niet goed blijkt te liggen.

Deze transformatie kan in het veld worden uitgevoerd, door via een internetbrowser in te loggen op de database.



### 7.3.9. POI's in zoekbereik

In dit deel van de code wordt de onderlinge afstand tussen het mobiele apparaat en de omliggende augments berekend:

waarbij

en  $lon_n = \text{Lengtegraad}$

De overige voorkomende getallen zijn de onderlinge schaalfactoren tussen de gebruikte eenheden.

Conversie van graden naar radialen  $1^\circ = \frac{1^\circ \times \pi}{180} \approx 0.0175^{rad}$

$$\frac{1^\circ}{60} = 1^{minuut} = 1^{nautische\ mijl} \approx 1.1515^{engelse\ mijl}$$

$$1.1515 \times 1.609344 = 1.853159616^{kilometer}$$

$$1^\circ = 60^{minuten} \times 1.853159616^{kilometer} = 111.189577^{kilometer}$$

Vergelijking 7.1

In het GPS-systeem wordt aangenomen dat de aarde perfect rond is, de straal is gedefinieerd op 6370.7 kilometer.

$$R_{aarde} = \frac{1}{2} \times \frac{omtrek}{\pi} = \frac{1}{2} \times \frac{(360^\circ \times 111.189577_{Km})}{\pi} = 6370.693488_{Km}$$

De onderlinge afstand "mobiel apparaat, augment" kan uit de coördinaten worden berekend.

$$\Delta_{GPS,Augment} = \left( \left( \left( \arccos \left( \sin \left( \frac{lat_1 \times \pi}{180} \right) \times \sin \left( \frac{lat \times \pi}{180} \right) + \cos \left( \frac{lat_2 \times \pi}{180} \right) \times \cos \left( \frac{lat \times \pi}{180} \right) \times \cos \left( (long - lon) \times \frac{\pi}{180} \right) \right) \right) \times \frac{180}{\pi} \right) \times 60 \times 1.1515 * 1.609344 \times 1000 \right)$$

Vergelijking 7.2

Bron: (Meridianworlddata.com, 1998-2007)

De gegevens van de POI's die binnen het zoekbereik voorkomen (*tot een maximum van 50*) worden verzameld en gebonden in een stringtekst.

De resultaten worden tijdelijk in een aparte matrix genaamd "rawPois" opgeslagen.

Als voorwaarde en controle op correcte informatievoorziening kan de functie in § 7.3.9 alleen worden uitgevoerd bij POI's, waarbij alle attributen in de database kloppen.



PHP CODE

### 7.3.10 Verzamelen resultaten

Door middel van de bewerking in § 7.3.9 zijn nu alle POI's in de database voorzien van een afstands-attribuut oftewel de radius.

Alle te verzenden data worden geplaatst in de matrix "\$response".

Wanneer er géén POI's bestaan, dan krijgt de gebruiker hiervan een melding.

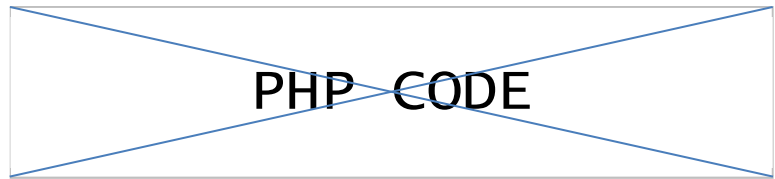


PHP CODE



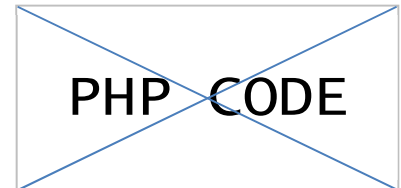
### 7.3.11 Verzenden JSON antwoord

De matrix “\$response” waarin alle te verzenden informatie verzameld is, wordt geconverteerd naar het JSON-formaat. Dit is het enige formaat dat door de Layaapplicatie ondersteund wordt.



*Bron PHP-programmering: (the PHP Documentation Group, 1997 - 2012)*

Voordat het JSON-bericht het mobiele apparaat bereikt, wordt deze gevalideerd door Laya. De controleparameters worden aan het JSON-bericht toegevoegd.

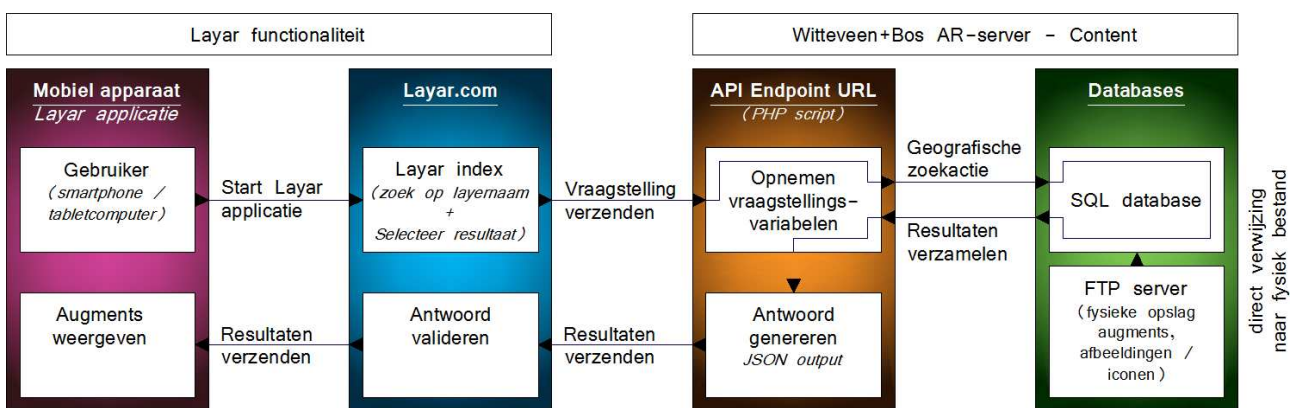


Als er een errorCode 0 gegeven wordt, zijn er géén fouten geconcludeerd. Andere mogelijke foutcodes betreffen:

ErrorCode	Omschrijving
0	Géén fouten, augments worden weergegeven
20	Géén POI's binnen zoekradius, géén weergaven augments
30	Kan geen verbinding maken met de database.

Tabel 7.1

De uiteindelijke communicatiestroom tussen het mobiele apparaat, de Layaindex en de eigen augmentsserver als in geheel hoofdstuk 7 beschreven is weergegeven in onderstaand organogram (zie figuur 7.3). In de casus “Nauwkeurigheid” wordt de werking van de server in de praktijk getest.



Figuur 7.3

## 7.4 Resultaat

Het resultaat van dit onderdeel is een werkende Laya-server met een eigen productieomgeving dat deels geautomatiseerd verloopt. In de volgende hoofdstukken wordt beschreven waaraan de augments moeten voldoen en hoe het generalisatieproces door de server geregeld wordt. In bijlage B3 vindt u het gehele productieproces voor het produceren van een augment. In de casussen “Nauwkeurigheid” en “Riolering” wordt de Laya-server in de praktijk getest.

## 8. GENERALISATIE 3D-MODELLEN

### Waarom generaliseren?

Mobiele apparatuur als PDA's, Smartphones en tabletcomputers hebben beduidend minder rekenkracht dan een laptop of desktopcomputer. Met het ontwerp van de augments moet hiermee rekening worden gehouden. De augments dienen zo schoon mogelijk gevormd te worden, zodat een augment, dat een klein geheugen (*grootte in bytes*) in beslag neemt, toch snel en correct in de Layarapplicatie weergegeven wordt.

De Layarapplicatie is zo gemaakt dat de weergave van het betreffende model altijd goed (*in de zin van detaillering*) weergegeven wordt. Afhankelijk van de complexiteit van het model en de prestaties van de mobiele apparatuur is het bepalend of het augment vloeiend mee loopt met de gebruiker wanneer hij zich in het terrein verplaatst.

Om een correcte weergave te waarborgen, dient in het ontwerp van de augment rekening gehouden te worden met het detail waarin het augment opgebouwd wordt. Layar adviseert modellen die maximaal uit 7000 faces bestaan *bron: (Layar, 2011)*. Een face is een tweedimensionaal vlak in een driedimensionale ruimte bestaande uit minimaal drie tot maximaal vier vertexpunten.

Het uiteindelijke bestandsformaat dat door de Layarapplicatie weergegeven kan worden betreft het \*.L3D formaat dat alleen gerealiseerd kan worden door een OBJ / MTL-bestand te converteren.

### 8.1 Wavefront 's OBJ

Het OBJ/MTL-formaat is een relatief oud (*sinds 1984*) en simpel formaat om een driedimensionaal model mee te definiëren. Het voordeel hiervan is, dat het model zowel met een viewer als in tekstuele vorm gelezen kan worden. De entiteiten die in OBJ ondersteund worden betreffen: punten (*points*), lijnen (*lines*) en vlakken (*faces*). De Layar3DModelConverter converteert uiteindelijk het OBJ/MTL-bestand naar een Layar-eigen L3D-bestand dat alleen de content "faces" ondersteunt.

Het nadeel is dat voor Layar de modeldefinitie alleen maar kan bestaan uit tweedimensionale vlakken bestaande uit drie of vier vertexpunten in een driedimensionale ruimte. Dit betekent ook dat er géén gecurvde objecten in gedefinieerd kunnen worden.

R	VN	OBJ content-code
1		mtllib kubusmodel.mtl
2		o kubus
3	1	v 2.000000 0.000000 -2.000000
4	2	v 2.000000 0.000000 1.000000
5	3	v -1.000000 0.000000 1.000000
6	4	v 4.000000 -1.000000 -2.000000
7	5	v 2.000000 3.000000 -2.000000
8	6	v 2.000000 3.000000 1.000000
9	7	v -1.000000 3.000000 1.000000
10	8	v -1.000000 3.000000 -2.000000

Tabel 8.1

*Het gehele model wordt als in de derde kolom van de hiernaast staande tabel 8.1 gedefinieerd. De kolom "R" staat voor regelnummer, de kolom "VN" voor het vertexnummer. In tabel 8.2 staat "FN" voor het facenummer.*

In het OBJ-bestand wordt in de eerste regel verwezen naar het bijbehorende, tevens gelijknamige MTL-bestand. Het MTL-bestand beschrijft de opmaak van de te vormen faces (*zie § 8.2 Wavefront 's MTL*).

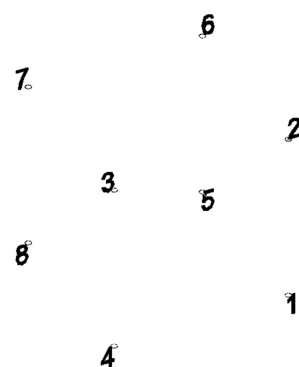
*Bron: (Murray & Van Ryper, 1996-05)*

In regel 2 wordt het eerste voorkomende object benoemd, waarnaar alle vertexpunten worden gedefinieerd. De definitie van een vertexpunt begint met een “v” gevolgd door zijn coördinaten. De vertexpunten worden genummerd in de volgorde als hoe deze in het OBJ-bestand geschreven staan (zie kolom “VN”). In figuur 8.1 ziet u een afbeelding van de in tabel 8.1 gedefinieerde vertexpunten.

### 8.1.1 Right-hand-rule (face-vorming)

In OBJ wordt een face gedefinieerd uit vertexpunten. De volgorde waarin de vertexpunten de face vormen is bepalend voor de definitie van de voor- en achterkant van de face. De achterkant van een face is namelijk onzichtbaar. De zijde die volgens de right-hand-rule getekend is, is zichtbaar in de Layaapplicatie.

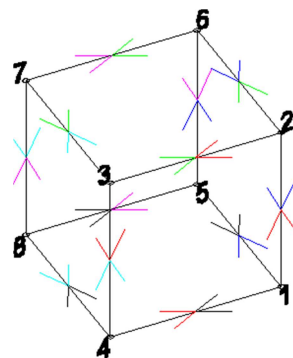
Om een van-buitenaf-gesloten kubus te vormen van de punten in tabel 8.1 (zie ook figuur 8.1), dienen de vertexpunten op de juiste manier met elkaar verbonden te worden. In OBJ-code ziet de face-vorming eruit als in tabel 8.2 omschreven. In figuur 8.2 ziet u een schematische tekening van de face-vorming.



Figuur 8.1

R	FN	OBJ content-code
111		usemtl hout
112		s off
113	11	F 1 2 3 4
114	22	F 5 8 7 6
115	33	F 1 5 6 2
116	44	F 2 6 7 3
117	55	F 3 7 8 4
118	66	F 5 1 4 8

Tabel 8.2

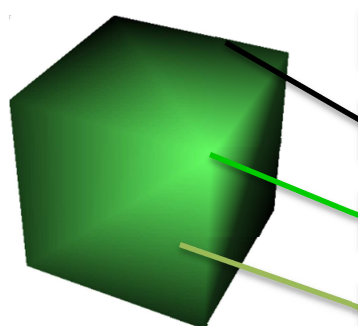


Figuur 8.2

Voor een eventueel volgend object wordt de OBJ-code als in bovenstaand proces herhaald vanaf regel 2 in tabel 8.1.

## 8.2 Wavefront 's MTL

Bij het OBJ-bestand hoort het gelijknamig MTL-bestand. Deze beschrijft de opmaak van de faces in kleur (*diffuse-, omgevings- en reflectiekleur*), zie figuur 8.3.



Omschrijving	MTL code	Waarde
Omgeving	Ka	R,G,B
Reflectie	Ks	R,G,B
Diffuus	Kd	R,G,B
Glans	Ns	percentage
Refractie	Ni	percentage
Transparantie	d	percentage
Lichtsterkte	illum	percentage
Texturen	map_<url>	maplocatie

Figuur 8.3

De RGB-waarden betreft het percentage van de maximale kleursterkte gedeeld door 100. Groen betreft in deze RGB-notering de waarde **0.0 1.0 0.0**.

In tabel 8.3 staat de MTL-code voor de opmaak van figuur 8.3.

Texturen zijn afbeeldingen, deze kunnen niet geschreven worden in de MTL-code. In regel 9 wordt verwezen naar een textuur.

Texturen zijn optioneel. Wanneer texturen toegevoegd worden, dienen de afbeeldingen zich te bevinden in een gelijknamige onderliggende map ten opzichte van de OBJ- en MTL-maplocatie.

R	MTL content-code
1	newmtl Material_1
2	Ns 96.078431
3	Ka 0.000000 0.000000 0.000000
4	Kd 0.640000 0.640000 0.640000
5	Ks 0.500000 0.500000 0.500000
6	Ni 1.000000
7	d 1.000000
8	illum 2
9	map_kubus/textuur.png

Tabel 8.3

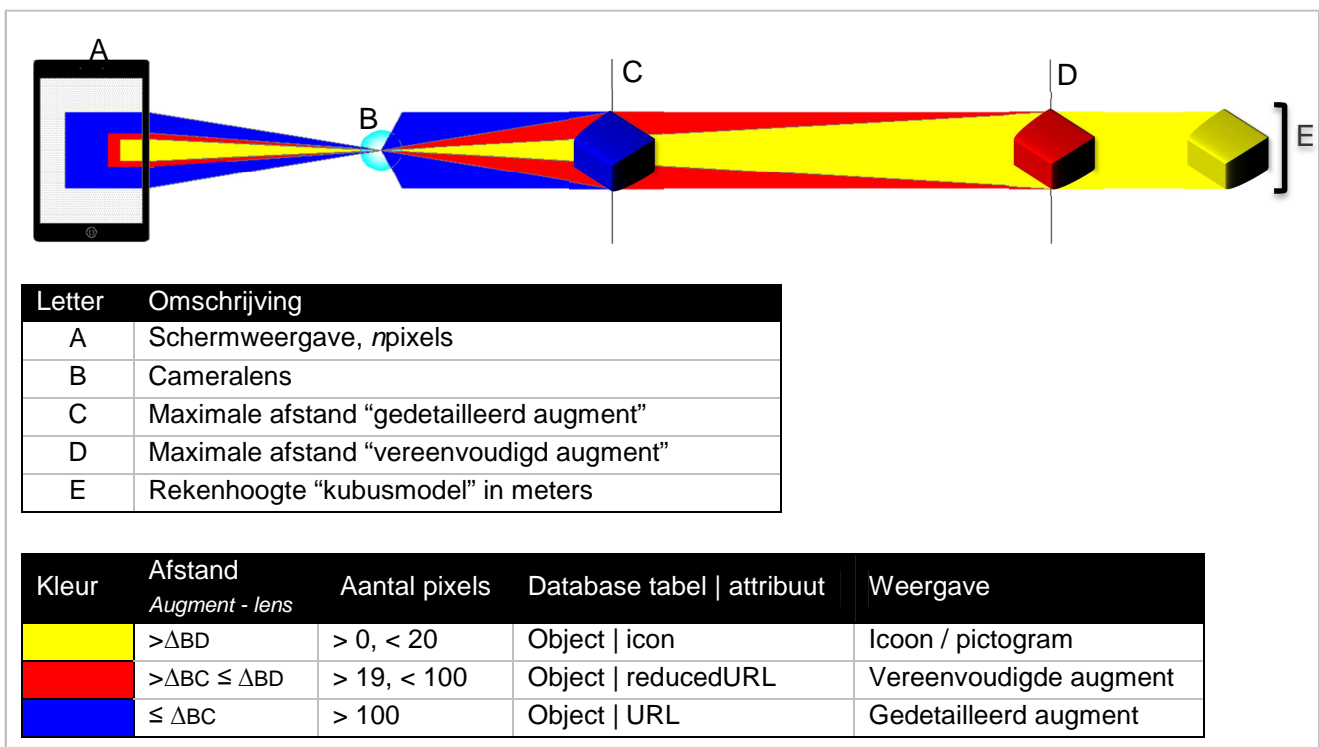
### 8.3 Gesteuurde generalisatie

Eén augment kan in de Layarapplicatie drie verschillende modellen bevatten, namelijk; 1. een icoon; 2. een vereenvoudigd 3D-model, en 3. een gedetailleerd 3D-model. De keuze van weergave is gerelateerd aan het weer te geven aantal pixels in het beeldscherm van het mobiele apparaat.

$$nPixels = \frac{E}{\frac{afstand}{\frac{1}{300}}} \quad \text{Vergelijking 8.1}$$

$$E = [Object | size] \times [Object | scale] \quad \text{Vergelijking 8.2}$$

De rekenhoogte hoeft géén relatie te hebben met het model van de augment. De rekenhoogte wordt bepaald aan de tekstuele attributen in de database. De berekening van het aantal pixels wordt altijd herleid aan het kubusmodelprincipe (zie figuur 8.4). Dit houdt voor de berekening in dat: de verhouding “nPixels” tussen “Object | size”, “Object | scale” en de afstand tussen apparaat en augment altijd gelijk is.



Figuur 8.4

### 8.4 Conclusie generalisatie

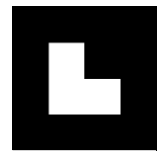
Het generaliseren van een augment wordt op twee manieren geregeld. Als eerste dienen er twee zo schoon mogelijke (*orbit*)modellen uit te tekening. Eén variant met detail en één kale variant. Wanneer de gedetailleerdste variant te zwaar wordt voor de Layarapplicatie wisselt deze automatisch naar de (*lichtere*) kale variant.

## 9. BEELDHERKENNINGSTECHNIEK

In het plan van aanpak staat beschreven dat dit hoofdstuk zou gaan over mogelijkheden om een betere positiebepaling van de augments te realiseren. De verwachting was dat de positionering van augments versterkt kon worden door het herkennen van patronen in het gefilmde beeld. Dit blijkt een verkeerd uitgangspunt geweest te zijn, doordat Layar deze optie niet biedt. De combinatie van de positionering van augments door middel van GPS en een versterking door middel van beeldherkenning blijkt tot heden in Layar niet mogelijk te zijn. Het hoofdstuk Beeldherkenningstechniek wordt hierdoor als volgt ingevuld:

Het onderdeel beeldherkenningstechniek beperkt zich tot de herkenning van lineaire afbeeldingen in een driedimensionale ruimte. Voor beeldherkenning dient de afbeelding als grondslag voor de positionering van de augments, waardoor componenten als GPS-antenne, gyroscoop, magnetometer en accelerometer overbodig worden.

De lineaire afbeeldingen zijn markers die relatief simpele binaire afbeeldingen betreffen. Deze kunnen goed en snel door een mobiel apparaat geïnterpreteerd worden (zie *figuur 9.1*).



*Figuur 9.1*

Dit hoofdstuk beschrijft dan ook de beeldherkenningstechniek achter het identificeren van de marker en het proces voor de positiebepaling van het apparaat ten opzichte van het te projecteren augment.

## 9.1 Digitale beeldbewerking / beeldherkenning

Voor beeldherkenning in augmented reality, is het zeer belangrijk dat een marker goed en snel herkend kan worden in groot zoekbereik en onder diverse transformaties.

### 9.1.1 Sobel algoritme (Matlab)

De randherkenning zelf wordt gedaan door middel van het Sobel-algoritme. Het Sobel-algoritme scant elke pixel in het beeld twee keer; één keer in de x-richting en één keer in de y-richting. Onderstaand de gehele procedure geschreven als zijnde Matlab-programma:

Bron: (Angel, 2010)

```
A=imread('C:\Bestandslocatie\afbeelding.jpg');
B=rgb2gray(A);

C=double(B);

for i=1:size(C,1)-2
    for j=1:size(C,2)-2

        Gx=((2*C(i+2,j+1)+C(i+2,j)+C(i+2,j+2))-(2*C(i,j+1)+C(i,j)+C(i,j+2)));
        Gy=((2*C(i+1,j+2)+C(i,j+2)+C(i+2,j+2))-(2*C(i+1,j)+C(i,j)+C(i+2,j)));

        B(i,j)=sqrt(Gx.^2+Gy.^2);

    end
end
figure,imshow(B); title('Sobel gradiënt');

Thresh=100;
B=max(B,Thresh);
B(B==round(Thresh))=0;

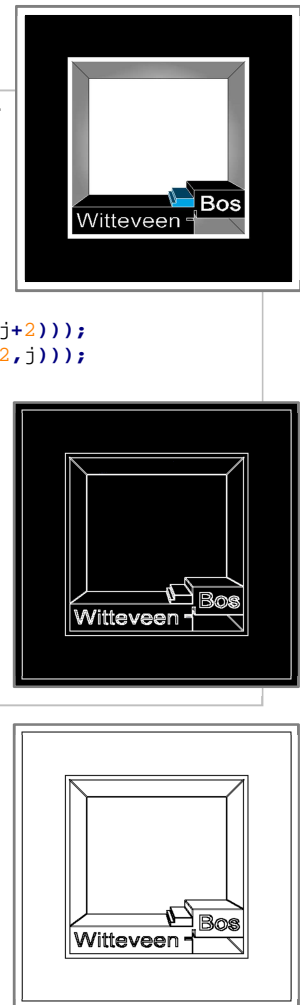
B=uint8(B);
figure,imshow(~B);title('Gevonden randen in afbeelding');
```

Matlab-programma 1

In bovenstaand proces (*Matlab-programma 1*) wordt als eerste het kleurenbeeld omgezet naar een zwart-wit beeld met grijsintinten. In een zwart-wit beeld met grijsintinten kan met een veel kortere programmacode een Sobel-randdetectie over het beeld uitgevoerd worden, doordat maar op één waarde een analyse uitgevoerd hoeft te worden.

Een zwart-wit beeld met grijsintinten kent de eigenschap dat alle waarden in R, G en B (0 t/m 255) van een grijsintint gelijk betreffen ( $R=G=B$ ). De waarde "helderheid" betreft de gemiddelde waarde van een RGB-kleur.

Vervolgens wordt het thresholding-proces uitgevoerd, waardoor alleen de in het beeld gevonden randen / lijnen overblijven in een binair beeld.



Figuur 9.2

#### R,G,B (grijsintinten)

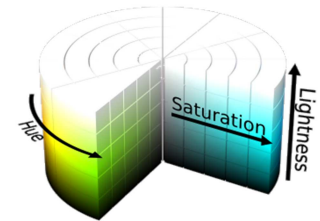
000,000,000  
025,025,025  
050,050,050  
075,075,075  
100,100,100  
125,125,125  
150,150,150  
175,175,175  
200,200,200  
225,225,225  
250,250,250



### 9.1.2 Proces thresholding

Anders dan “Matlab-programma 1” doet vermoeden is het voor het thresholding-proces niet noodzakelijk het beeld vooraf naar een zwart-wit beeld met grijsntinten om te zetten (*thresholding doet dit zelf ook*).

Het gefilmde beeld wordt binnengehaald in het RGB-stelsel, waarna de software de helderheid van de kleuren bepaalt volgens het HSL-stelsel (zie *figuur 9.3*). Dit gebeurt door middel van de onderstaande formule:



Figuur 9.3

$$L = \frac{1}{2}(M + m) \quad \text{Vergelijking 9.1}$$

Waarbij:

$L = Luma / \text{Lightness}$

Gemiddelde helderheidsdimensie

$M = \text{Max}(R, G, B)$

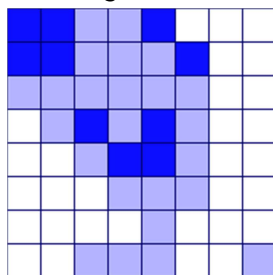
Grootste RGB-waarde / meest heldere kleur

$m = \text{Min}(R, G, B)$

Kleinste RGB-waarde / meest donkere kleur

Doordat de grootste, kleinste en dus ook de gemiddelde helderheidsdimensies bekend zijn, kan het beeld opgedeeld worden in grijsntinten. Het thresholding-principe deelt grijsntinten altijd op in drie categorieën. Deze categorieën worden evenredig bepaald aan de hand van de grootste en kleinste luma / helderheidswaarde (zie *figuur 9.4*).

Ontvangen beeld



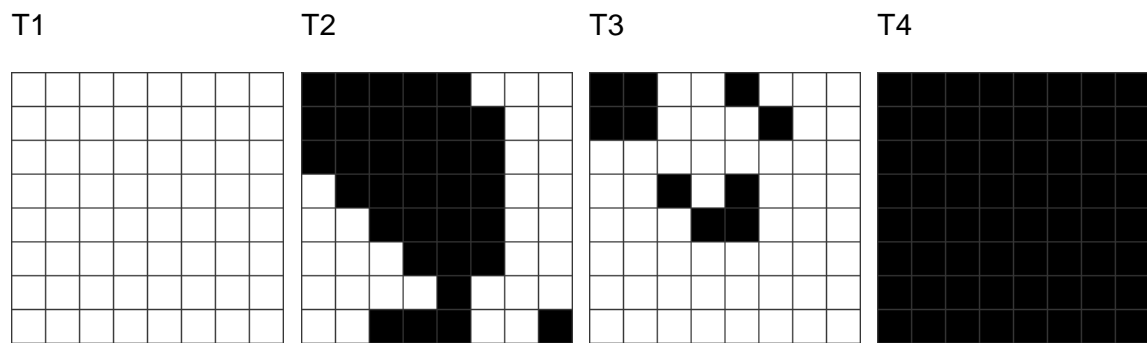
→ Conversie naar grijsntinten

1	1	2	2	1	3	3	3
1	1	2	2	2	1	3	3
2	2	2	2	2	2	3	3
3	2	1	2	1	2	3	3
3	3	2	1	1	2	3	3
3	3	3	2	2	2	3	3
3	3	3	3	2	3	3	3
3	3	2	2	2	3	3	2

Figuur 9.4

Vervolgens worden er vier analyses uitgevoerd op het in grijstinten verkregen beeld. Wanneer het resultaat van één van de vier analyses (*of een combinatie hiervan*) zodanig overeenkomt met de gedefinieerde marker, dan kan het augment relatief ten opzichte van deze marker geprojecteerd worden.

In de onderstaande afbeelding (*figuur 9.5*) het resultaat van de thresholding-analyse op het beeld van figuur 9.4. *Bron: (Frontier Vision)*



Figuur 9.5

## 9.2 Oriëntering apparaat & augment

Om een augment met behulp van een marker te projecteren, moet de geometrische relatie tussen de camera en de marker bepaald worden. De marker betreft een tweedimensionale afbeelding ( $p' = (x', y')$ ) in een driedimensionale ruimte  $p = (x, y, z)$ . De volgende parameters dienen bekend te zijn alvorens een projectie van een augment in de apparaatdisplay gedaan kan worden:

- Perspectiefprojectie;
- cameraparameters (*lengte / pixel*);
- transformatieparameters (*rotatiematrix  $R$  en een translatievector  $T$* ) waarin de geometrische relatie tussen posities van de camera en marker wordt gedefinieerd.

### 9.2.1 Perspectiefprojectie

De perspectiefprojectie transformeert het zichtvolume in het canoniek zichtvolume. Dat wil zeggen: objecten die in de augmented reality dichtbij gepositioneerd zijn, worden in verhouding groter getoond dan verder gelegen objecten. Hetzelfde principe geldt voor de beeldherkenning. In de database zijn de afmetingen van de gefilmde marker (zie figuur 9.6) gedefinieerd, waardoor het apparaat kan berekenen op welke afstand de marker zich ten opzichte van het apparaat bevindt.

Om de perspectiefprojectie ( $\lambda$ ) te bepalen wordt de afstand ( $Z$ ) gedeeld door de cameraconstante ( $f = \Delta OP$ ), die tevens de schaalfactor (*afmeting en resolutie*) naar het digitale display bevat.

Deze afstand kan bepaald worden doordat afmeting van de marker in de augmentdefinitie bekend is.

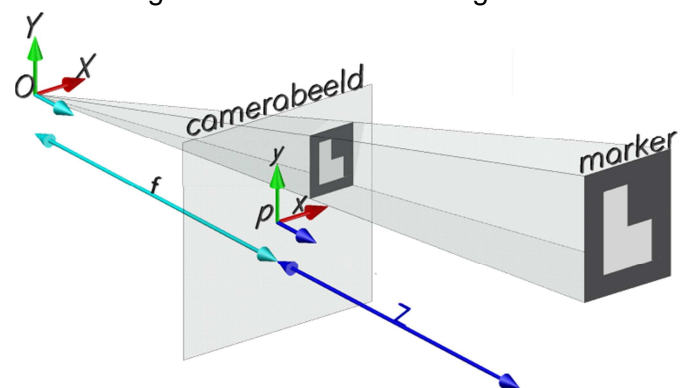
$$\lambda = Z/f$$

$$\lambda \times \begin{bmatrix} x \\ y \\ f \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

De projectiematrix voor een marker (*plat vlak in een driedimensionale ruimte*) kan hierdoor als volgt gedefinieerd worden:

$$\begin{bmatrix} x \\ y \\ f \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Vergelijking 9.2



Figuur 9.6

### 9.2.2 Interne parameters van de camera

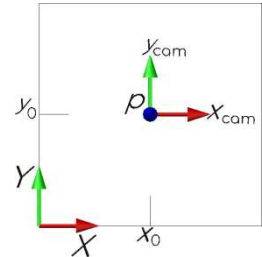
Doordat de perspectiefprojectie en zo ook de onderlinge afstand ( $\Delta_{apparaat,marker}$ ) bekend is, kan berekend worden wat de pixelgrootte moet worden.

$$K_x x_{cam} = x - x_0$$

$$K_y y_{cam} = y - y_0$$

$$Pixelgrootte = \frac{1}{f} \times \begin{bmatrix} fK_x & 0 & a_0 \\ 0 & fK_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_{cam} \\ y_{cam} \\ f \end{bmatrix} = K \times \begin{bmatrix} x_{cam} \\ y_{cam} \\ f \end{bmatrix}$$

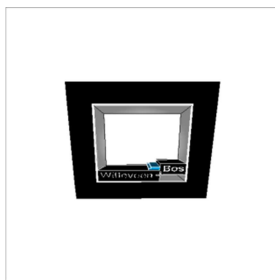
Vergelijking 9.3



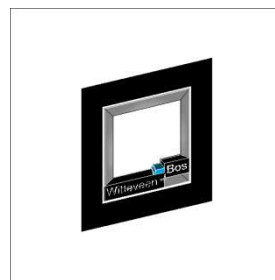
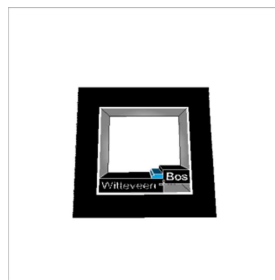
Figuur 9.7

### 9.2.3 Transformatieparameters

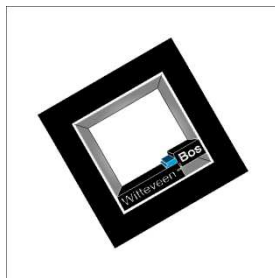
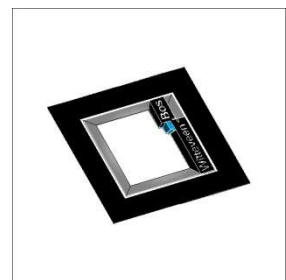
Vervolgens worden de transformatieparameters berekend. Hiervoor is een model nodig om alle mogelijke transformaties van een tweedimensionale afbeelding (*marker*) in een driedimensionale ruimte te berekenen. Deze transformaties gaan gepaard met de volgende eigenschappen:



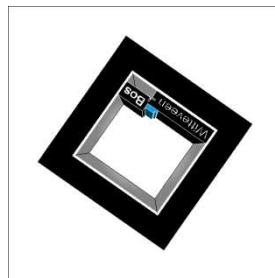
Centrale projectie



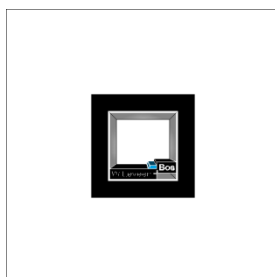
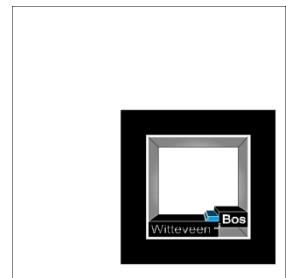
Parallele hoeken



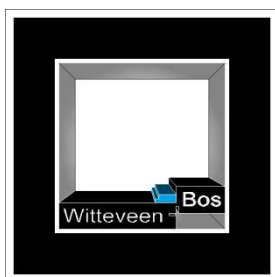
Rotatie



Verplaatsing



Verschaling



Figuur 9.8

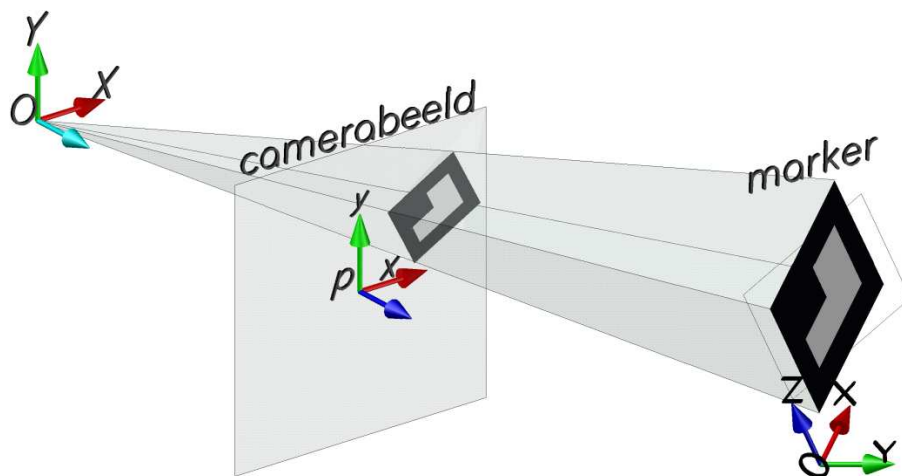
Doordat de marker bekend is in grootte en oriëntering, kan de relatieve positie ten opzichte van de camera met de onderstaande formule hieruit berekend worden.

$$\begin{bmatrix} X_{cam} \\ Y_{cam} \\ Z_{cam} \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad \text{Vergelijking 9.4}$$

$$\begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta \times \cos\psi & -\cos\theta \times \sin\psi + \sin\phi \times \sin\theta \times \cos\psi & \sin\phi \times \sin\psi + \cos\phi \times \sin\theta \times \cos\psi & v_x \\ \cos\theta \times \sin\psi & \cos\theta \times \cos\psi + \sin\phi \times \sin\theta \times \sin\psi & -\sin\phi \times \cos\psi + \cos\phi \times \sin\theta \times \sin\psi & v_y \\ -\sin\theta & \sin\phi \times \cos\theta & \cos\phi \times \cos\theta & v_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Doordat de camerapositie en oriëntatie ten opzichte van de marker bekend is, kan het augment ten opzichte van de marker geprojecteerd worden. Wanneer de gehele berekening van beeldherkenning, plaatsbepaling en oriëntering van de camera tot aan het doorgeven van de gerenderde augment 60x per seconde (60 Hz) uitgevoerd kan worden, zal het augment in de belevenis van de gebruiker in realtime worden weergegeven.

Projectie van 2D afbeelding in een 3D ruimte

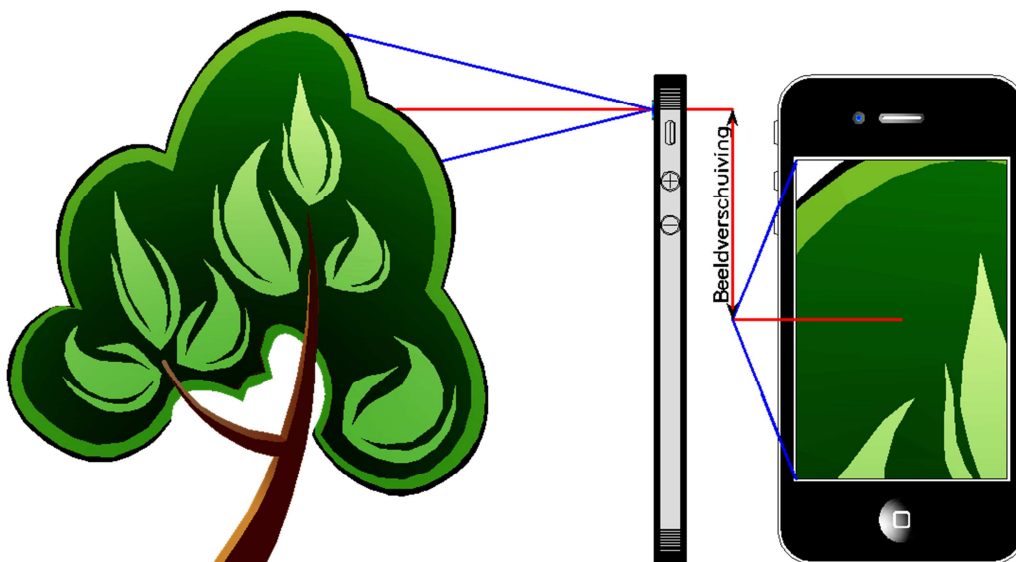


Figuur 9.9

### 9.2.4 Beeldverschuiving

Hoewel het augment correct gepositioneerd wordt in het waargenomen beeld, kan het storend werken dat het beeld in de display verschoven is ten opzichte van de werkelijkheid. De camera's van bruikbare smartphones en tabletcomputers (zie *bijlage B1(a) Inventaris mobiele apparatuur*) zijn nooit gecentreerd in het toestel geplaatst, wat leidt tot een parallelle verplaatsing van het beeld in de werkelijkheid.

Hoe verder de camera geplaatst is van het displaymiddenpunt, hoe groter de parallelle verplaatsing van het gefilmde beeld. Daarnaast geldt dat (*doordat in perspectief gekeken wordt*) het effect versterkt wanneer een terrein van zeer dichtbij wordt gefilmd. In de onderstaande afbeelding (*figuur 9.10*) een geïllustreerde uitleg betreffende het fenomeen "beeldverschuiving".



Figuur 9.10

De storing "beeldverschuiving" kan alleen verholpen worden door de camera exact gecentreerd te plaatsen ten opzichte van het scherm waarop het camerabeeld wordt weergegeven.

## 9.3 Conclusie beeldherkenning

De techniek beeldherkenning gaat gepaard met een zeer sterke apparaatoriëntatie. Met beeldherkenning is het apparaat niet afhankelijk van hardwarecomponenten als een gyroscoop, kompas, gps-antenne noch accelerometer. Deze waarnemingen hoeven en kunnen daarom niet berekend worden, met als gevolg dat de augmentafbeeldingen met 60 frames per seconden berekend en gerenderd kunnen worden.

# 10. CASUS NAUWKEURIGHEID

In deze casus is de zelf gebouwde server met de mobiele telefoonapplicatie “Layar” getest op nauwkeurigheid in diverse omstandigheden. Het model dat hiervoor is gebruikt betreft het ontwerp van het Noord-Zuidlijn station Vijzelgracht te Amsterdam. Het augmented realitymodel dat hiervan is gemaakt is echter niet in Amsterdam, maar op verschillende locaties onder diverse omstandigheden in Deventer getest.

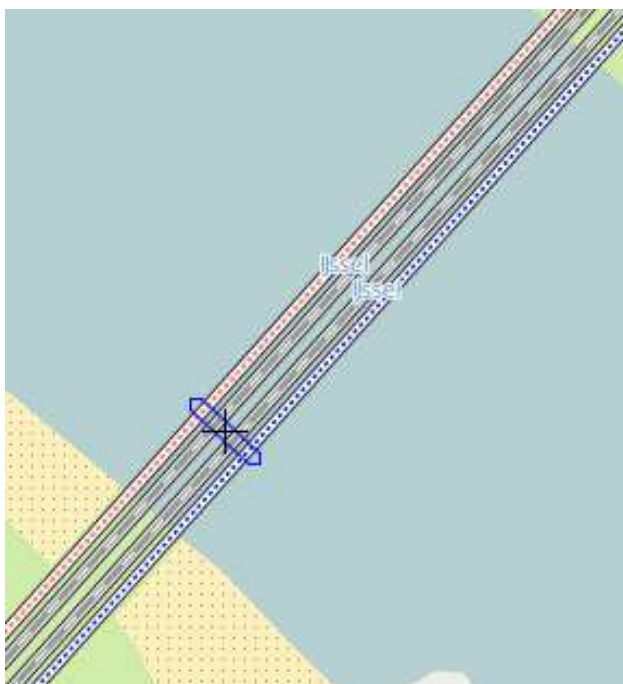
De apparaten, waar beschikking over is, zijn de Apple iPhone 4 en de Apple iPad 2. De werking van de Layarapplicatie is onder de volgende omstandigheden getest:

- augments op diverse afstanden (*controle op projectiegrootte*);
- oriëntatieverstoring (*kompas, gyroscoop*);
- nauwkeurigheid in XY-positionering (*in open terrein en tussen bebouwing*).

*De hoogtepositionering (Z) is relatief getest, doordat bekend is dat Layar gebruik maakt van het relatieve hoogteverschil tussen de gebruiker en het augment. Dit hoogteverschil wordt per augment gedefinieerd in de SQL-database en kan daardoor ook in het veld worden aangepast.*

## Controle op projectiegrootte

Om een controle op de projectiegrootte uit te voeren, is een augment gemaakt van een object dat zich al in het terrein bevindt. Er is voor gekozen om een augment te maken van het landhoofd van de spoorbrug in Deventer. Het model bestaat uit de contouren van het landhoofd die afkomstig zijn uit de GBKN, deze contouren zijn in de hoogte opgetrokken zodat het 3D-model gevormd kan worden (*zie de figuren 10.1 en 10.2*).



Figuur 10.1



Figuur 10.2



## Oriëntatie

Hoewel de projectiegrootte van het augment geregeld wordt door de met GPS-berekende positie van de gebruiker (*afstand tot het augment*), is de oriëntatierichting afhankelijk van de waarnemingen van de gyroscoop en het kompas.

De augmentpositie in het camerabeeld kan gemanipuleerd worden door het apparaat snel te roteren. Het augment zal zich vertraagd positioneren ten opzichte van de werkelijke rotatiesnelheid, waardoor de gebruiker een screenshot kan maken van een ideaalprojectie. Deze storende eigenschap lijkt veroorzaakt te worden door de Layarapplicatie zelf, aangezien andere gyroscoop- en kompasgebruikende applicaties op de iPhone / iPad sneller en accurater reageren wanneer het apparaat wordt geroteerd. Wanneer in een klein gebied meerdere augments gedefinieerd zijn, wordt deze vertraging groter. De Layarapplicatie wordt overbelast met computergegenereerde beelden, waardoor deze niet meer optimaal presteert.

Naast deze vertraging van positionering zijn tenminste de iPhone / iPad storingsgevoelig in het kompas voor nabije ( $\pm 1$  meter) elektronische apparaten (*andere apparaten zijn niet getest*). De maximaal waargenomen fout, veroorzaakt door een naburig elektronisch apparaat, bedraagt een hoekverdraaiing van ca.  $\angle 20^\circ$ .

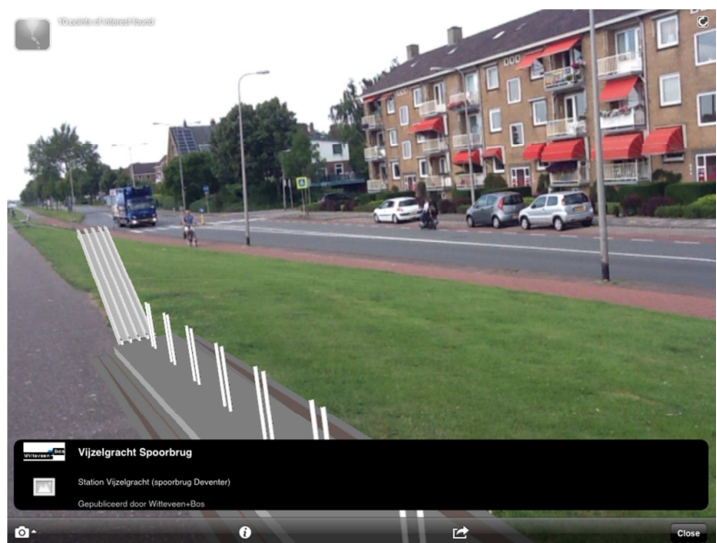
## XY-positionering in open terrein

Ten behoeve van de casus "Nauwkeurigheid" is een versimpeld 3D-ontwerp van Station Vijzelgracht te Amsterdam omgezet naar een augment.

In de figuren 10.3 en 10.4 wordt het augmented realitymodel gepositioneerd in een open terrein (*géén bomen noch gebouwen die het GPS-sigitaal beïnvloeden*). Hoewel het model juist gepositioneerd is, is er een slecht besef van de grootte en hoogte van het model. Ondergrondse augmented realitymodellen hebben zonder een duidelijke referentie aan het terrein, niet het juiste effect. Dit wordt vooral veroorzaakt door kleine gyroscopische en kompasstoringen. Doordat de pilaren in het augment niet parallel lopen met bijvoorbeeld de lantaarnpalen in het gefilmde beeld, kan men in dit geval maar moeilijk inschatten of het model op het voetpad ligt, of zoals bedoeld, in het hart van de rijbaan (*zie figuur 10.3*).



Figuur 10.3



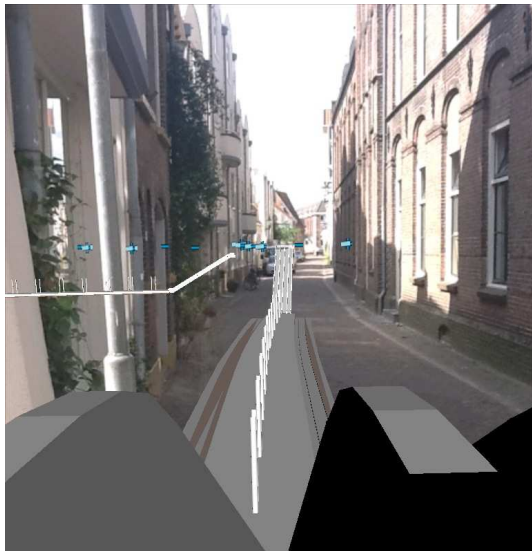
Figuur 10.4

In de casus “Ondergrondse infra” wordt het rioleringsmodel voorzien van een fictief hoogtevlak dat correspondeert met de werkelijke verhardingscontouren.

### XY-positionering tussen bebouwing

De XY-positionering is tevens getest in smalle straten tussen bebouwing. Het doel hiervan is om te bepalen of de positionering negatief wordt beïnvloed in vergelijking met het positioneren in open terrein. Het model is daarom in verschillende straten (*oriëntaties*) geplaatst om een zo objectief mogelijk beeld te krijgen van eventuele positioneringsfouten in de GPS-waarnemingen.

Tegen alle verwachtingen in vertoonden de augments van Station Vijzelgracht in alle geteste omstandigheden een vergelijkbaar gedrag. De positionering was in alle gevallen van een vergelijkbare kwaliteit, de oriëntatie uit de gyroscop- en kompaswaarnemingen daarentegen gaf tussen de bebouwing wel grotere fouten.



Wanneer men de screenshot in figuur 10.5 vergelijkt met de screenshot in figuur 10.4 kan geconcludeerd worden dat zowel de positionering als de diepteligging van ondergrondse augments beter door de gebruiker ingeschat kan worden in stedelijk gebied (*tussen bebouwing*).

Figuur 10.5

### Conclusie casus nauwkeurigheid

Uit de casus “Nauwkeurigheid” kan geconcludeerd worden dat Layarapplicatie prima gebruikt kan worden voor conceptuele ontwerpen waarbij de bewustwording van het design belangrijker is dan de geografisch juiste inpassing.

- De projectiegrootte van het augment in het beeld correspondeert met de grootte van de ontwerp-tekening.
- De oriëntatie in  $\Omega$ ,  $\varphi$ , en  $\kappa$  is zeer storingsgevoelig.
- De hoogtepositionering wordt relatief bepaald. Zolang men hiermee rekening houdt, hoeft dit géén belemmering te vormen. In de volgende casus (riolering), wordt een alternatieve oplossing gegeven om alsnog tot een juiste hoogtepositionering te komen.

# 11. CASUS RIOLERING

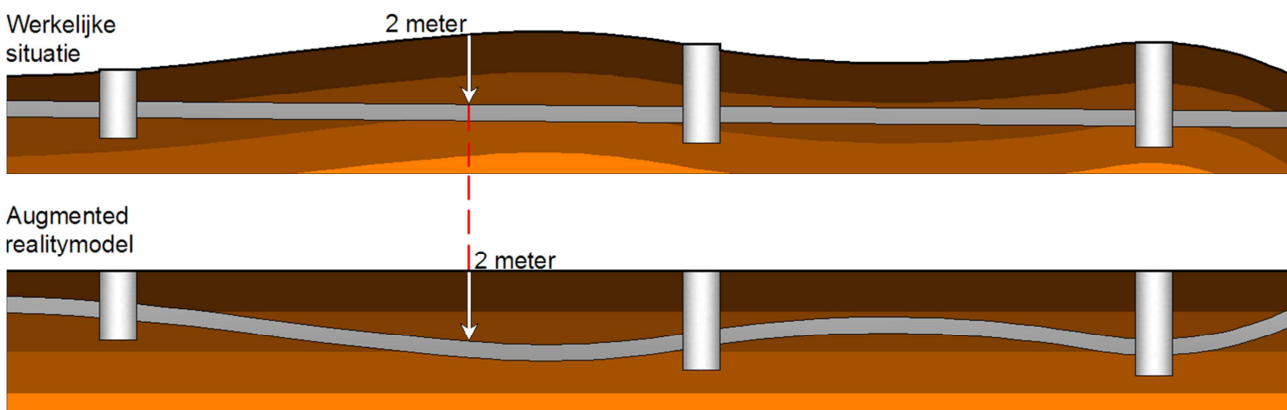
## 3D-augment van een rioleringsstelsel

Voor de casus riolering is gekozen om een bestaand rioleringsstelsel te gebruiken door deze uit te tekenen en te vergelijken met het werkelijke riool. Tevens wordt er bekeken of augmented reality toegepast kan worden in de vorm van een beheertool voor riolering.

Om een augment van een rioleringsstelsel te maken, dient men rekening te houden met de positioneringstechniek die Layar biedt. De hoogte (Z) wordt altijd relatief ten opzichte van de gebruiker bepaald. Voor kleine augments vormt dit geen probleem, maar als een groot model (bijvoorbeeld een rioleringsstelsel) ingepast wordt, dient men rekening te houden met de volgende eigenschappen:

- Een rioleringsstelsel beslaat een groot gebied;
- ondanks alle bewerkingen is Nederland niet vlak;
- Layar maakt gebruik van een relatieve hoogtepositionering.

Het augmented realitymodel, dat voor deze casus gemaakt is, is daarom niet NAP-gerelateerd, maar ten opzichte van de maaiveldhoogte. Het maaiveld fungeert als hoogtereferentie. Dit betekent dat het model geometrisch vervormd moet worden wil deze in het gehele gebied op de juiste hoogte in het camerabeeld geprojecteerd worden. In figuur 11.1 wordt zo'n vervorming van de werkelijke situatie naar het augmented realitymodel uitgebeeld.

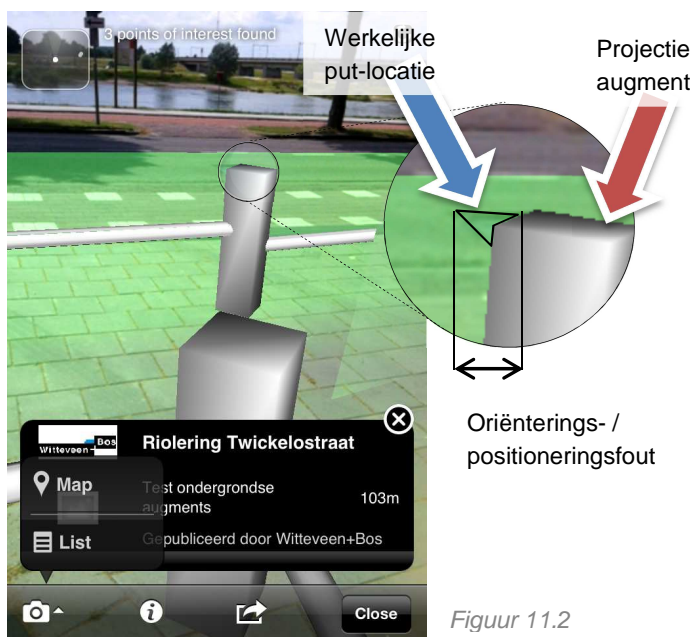


Figuur 11.1

Zoals in figuur 11.1 het augmented realitymodel wordt vervormd, klopt voor de gebruiker de diepte. Wanneer het landschap op korte afstanden grote hoogteverschillen bevat, dan wordt de vervorming van de rioolstrengen extremer. Hoewel de projectiediepte van het augment op de plaats van de gebruiker wel klopt, zal het model, bekeken van grotere afstanden, er grillig en onrealistisch uitzien.

Ten tijde van de productie van het rioleringsaugment kwam het besef van de hoeveelheid arbeid die hierachter verscholen zit. Een dagproductie voor het 3D-modelleren, inclusief het toepassen van de vervorming hierop, komt uit op ongeveer 50 rioolstrengen. Dit proces zou met aanpassingen in de software geautomatiseerd kunnen worden, wanneer men in bezit is van een gedetailleerd hoogtemodel van het maaiveld.

Het augmented realitymodel van het rioleringsstelsel is volgens de vervorming in figuur 11.1 ontworpen. Ter versterking van de oriëntatie is er een transparant vlak aan toegevoegd dat de verhardingscontouren symboliseert.



Figuur 11.2

De augments worden in de hoogte geprojecteerd op -2.00 meter ten opzichte van het mobiele apparaat.

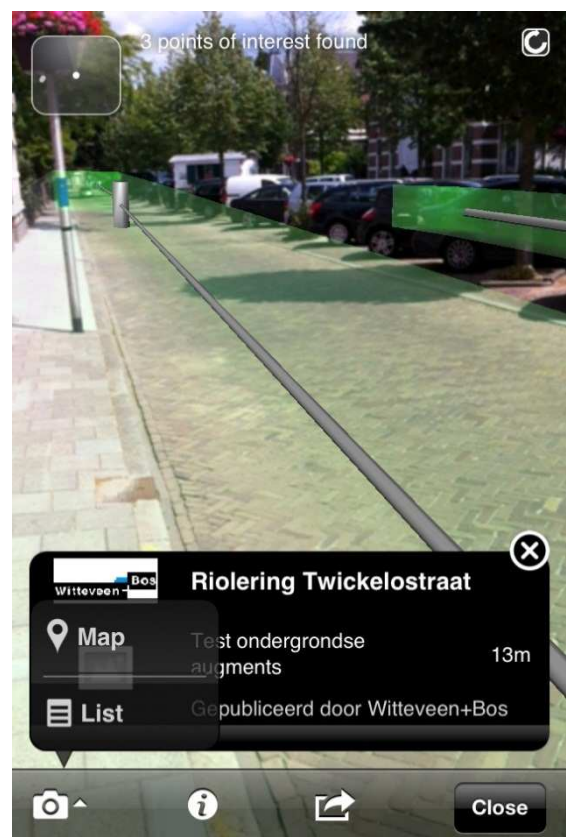
Het transparante groene vlak symboliseert het maaiveld.

De positionering van het augment is in dit geval optimaal.

In figuur 11.3 ziet u een screenshot waarin het effect van de vervorming duidelijk aanwezig is. In de werkelijke situatie loopt het terrein af, waardoor het augmented realitymodel boven de weg zweeft.

Geconcludeerd kan worden dat het wel of niet vervormen van een augmented realitymodel afhankelijk is van de toepassing.

- Dient een model dat een groot gebied beslaat op locatie, hoogtegetrouw te zijn, dan moet het model worden vervormd.
- Wanneer de geometrische relaties over grote afstanden dienen te kloppen en de hoogtepositionering een minder belangrijke rol speelt, dan is de beste keuze het model niet te vervormen.



Figuur 11.3

## Rioleringsstelsel in de vorm van informatielabels

Aan het einde van de uitvoering van de casus riolering, is het rioolstelsel niet als 3D-model, maar in de vorm van labels en icoonafbeeldingen opgezet. Dit systeem kan relatief snel opgezet worden, wanneer deze data beschikbaar zijn in tabelvorm.

De inhoud van de labels wordt gevormd door attribuutdata in de SQL-database, waardoor de mogelijkheid geschepd wordt deze data in het veld te muteren.

Grafisch blijkt dit niet de oplossing. Alle informatie binnen het opgegeven zoekbereik wordt weergegeven, met als gevolg dat het beeld vol raakt met labels en iconen. Layar heeft de eigenschap een informatiekader weer te geven van het augment waarop het mobiele apparaat gericht staat. Bij teveel labels in het scherm kan de gebruiker niet meer inschatten bij welk label het op dat moment weergegeven informatiekader hoort.

## Conclusie casus riolering

Uit de beide varianten op de casus riolering kunnen de volgende conclusies geconcludeerd worden:

- Het opzoeken van riolering werkt wanneer gebruik wordt gemaakt van een 3D-augment dat gemodelleerd is volgens het principe in figuur 11.1. Bij het opsporen van riolering is alleen de positionering in de directe omgeving van de gebruiker van belang. Dat dit ook daadwerkelijk op deze manier toegepast gaat worden, is een onrealistische gedachte wanneer men kijkt naar de hoeveelheid werk dat hiermee gepaard gaat.
- Het in het veld muteren van tabelattributen in de SQL-database is mogelijk. De werkelijke toepassing daarentegen zou praktisch gezien alleen werken als de zee van labels en icoonafbeeldingen gecombineerd worden met een 3D-augment van het rioleringsstelsel. Alleen dan kan de gebruiker een duidelijk onderscheid maken in de labels en het bijbehorende rioolcomponent.
- Beide systemen zijn alleen bruikbaar in combinatie met een 3D-augment, waardoor de oplossingen te kostbaar zijn om deze werkelijk toe te gaan passen.



## 12. CONCLUSIES

In dit hoofdstuk worden alle onderzoeksvragen beantwoord. De belangrijkste feiten worden hierin genoemd, de volledige uitgeschreven conclusies, bevindingen en antwoorden worden per vraag verwezen naar het bijbehorende hoofdstuk in de scriptie of in de bijlagen.

### Onderzoeksvragen

- V 1 In hoeverre kan de visualisatietechniek “augmented reality” in de buitenruimte worden toegepast?
- A 1. Augmented reality (*Layar*) is een zeer geschikt medium om een publiek op locatie een goede “impressie” te geven van een nieuw te realiseren plan. Er dient wel rekening gehouden te worden met een beperkte nauwkeurigheid in positionering van het augment.
  - A 1.a Zo is Layar geschikt voor 3D-modellen tot 7000 faces. Dit zijn nagenoeg altijd objectgerichte modellen als kantoorgebouwen, civieltechnische kunstwerken en ondergrondse infrastructuur (*praktisch allemaal orthogonale modellen*).
  - A 1.b Layar is minder geschikt voor terreingerichte modellen. Denk hierbij aan landschapsarchitectuur, glooiende terreinen en meanderende rivieren. Deze gedetailleerde triangulatiemodellen overschrijden nagenoeg altijd de veilige acceptatiegrens van 7000 faces en worden daardoor op veel apparaten niet correct, dan wel gepaard met storingen of ruis, weergegeven.
- V 2 Wat is het huidige marktaanbod voor augmented realitysoftware? Wat zijn de onderlinge verschillen? Is Layar nog steeds de meest geschikte keuze?
- A 2.a Layar is de enige augmented realitysoftware waarbij 3D-augments geografisch gepositioneerd worden doordat het mobiele apparaat zijn eigen positie en oriëntatie bepaalt uit GPS, kompas en gyroscopische waarnemingen. Voor de projectie van geogerefereerde augments is Layar daarom de enige geschikte keuze.
  - A 2.b De augmented realitysoftware, die fungeert op basis van beeldherkenning, is steekproefsgewijs vergeleken doordat alle varianten, die geschikt zijn voor mobiele apparatuur, betaalde softwarepakketten zijn. AR-media kan gratis getest worden en is als zeer gebruiksvriendelijk ervaren. Ook in de bijbehorende mobiele telefoonapplicatie scoort AR-media (*zelfs bij zware / gedetailleerde modellen*) goed op de grafische aspecten.

De gehele beantwoording is te vinden in hoofdstuk 4: Augmented Realitysoftware.

V 3 Welke mobiele apparatuur (*smartphones / tabletcomputers*) is geschikt voor het gebruik van augmented reality? Welke apparaten presteren hierbij het beste?

A 3. Uit de inventaris is gebleken dat de apparaten Apple The New iPad 4G en Apple iPad 2 3G het hoogste scoren wat betreft verwerkingssnelheid. The New iPad 4G scoort daarnaast ook het hoogst op de grootte van de schermresolutie en de pixeldichtheid (*264 pixels per inch*).

Het nadeel van alle bruikbare Appleproducten is dat de displays van glas gemaakt zijn. Bij helder en zonnig weer fungeert het glas als een spiegel, waardoor datgene dat afgebeeld wordt slecht zichtbaar is voor de gebruiker.

De voorwaarden die voor de inventarisatie zijn opgesteld, zijn te vinden in hoofdstuk 5: Inventaris mobiele apparatuur.

De volledige inventaris "mobiele apparatuur" is te vinden in bijlage B1(a).

V 4 Hoe dient een eigen Layaarserver ingericht te worden om zelf augments te kunnen serveren?

A 4. Voor het bouwen van een eigen Layaarserver dient een code (*PHP*) geprogrammeerd te worden dat een SQL-database bevraagt. De bestandslocatie van dit PHP-script moet gelinkt worden als zijnde API Endpoint URL bij de website van Layaar.

De SQL-query die de SQL-database aanmaakt, wordt beschreven in hoofdstuk 7, § 7.2. Een beschrijving van de tabellen met de bijbehorende attributen is gegeven in bijlage B7(a).

Het PHP-script dat benaderd wordt door de mobiele telefoonapplicatie Layaar en bevestigingen doet op de SQL-database, wordt beschreven in hoofdstuk 7, § 7.3.

De structuur van het JSON-antwoord, dat gevormd wordt door het PHP-script, is te vinden in de bijlage B7(b).

V 5 Welke partijen kunnen eventueel benaderd worden voor Layaar ontwikkelaarssupport?

A 5. Wanneer support bij de ontwikkeling van een eigen augment-server gewenst is, zouden de organisaties "Menno Bieringa" en "Different Signal" benaderd kunnen worden. Deze organisaties bieden als enige ondersteuning voor het ontwikkelen van een 3D-augmentserver.

De gehele lijst "Support & Diensten" is te vinden in hoofdstuk 6: Inventaris Support & Diensten.

V 6 Wat is voor de mobiele apparatuur van nu de best haalbare weergave van augments? Hoe kan deze weergave gerealiseerd worden?

A 6. Een augment wordt weergegeven als hoe deze gedefinieerd is. De externe invloeden van licht (*lichtintensiteit en omgevingskleur*) kunnen gedefinieerd worden in de modeldefinities. De in het model gedefinieerde materialen zullen in de augment realistischer ogen wanneer de belichtingskleuren en intensiteit kloppen.

Houd er wel rekening mee dat, wanneer het weertype verandert (*bewolkt of zonnig, licht of donker*), de externe invloeden van de belichting ook veranderen.

V 7 Hoe kunnen zware (*gedetailleerde*) modellen goed gegeneraliseerd worden?

A 7.a De Layaapplicatie is van zichzelf zo geprogrammeerd dat, wanneer het mobiele apparaat moeilijkheden ondervangt doordat het model te complex is om weer te geven, deze in de SQL-database gaat zoeken naar een gegeneraliseerd (*reducedURL*) model dan wel icoon.

A.7.b Wanneer het mobiele apparaat geen moeilijkheden ondervangt, wordt de generalisatie bepaald aan de afstand tussen de gebruikerspositie en de augmentpositie. In hoofdstuk 8, § 8.3 wordt uitgelegd welke regels gelden voor deze generalisatie.

V 8. (*Hoe*) kunnen de nauwkeurigheidsproblemen in de plaatsbepaling geminimaliseerd worden?

A 8. Wanneer gebruik wordt gemaakt van de Layaapplicatie is de gebruiker in de oriëntering volledig afhankelijk van de hardwarecomponenten die zijn / haar smartphone of tabletcomputer bezit.

A 8.a De oriëntering in  $\Omega$ ,  $\varphi$ ,  $\kappa$  wordt verkregen door de gyroscoop en het digitaal kompas. De oriënteringsfouten kunnen alleen met behulp van beeldherkenning gecorrigeerd worden. Helaas biedt Laya géén mogelijkheid in een combinatie van beeldherkenning en positionering door middel van GPS.

Andere mobiele telefoonapplicaties positioneren de augment relatief (*niet geogerefereerd*) ten opzichte van een vooraf gedefinieerde marker. Dit geeft een sterke apparaatoriëntering.

Het toekomstideaal is echter de markers te vervangen door het beeld van de werkelijke situatie buiten, oftewel een herkenning van de natuurlijke patronen in het terrein.



A 8.b De positionering in X,Y,Z die verkregen wordt door waarnemingen van de GPS-antenne zou verbeterd kunnen worden door een communicatiestroom te vormen met een GPS-referentiestation (*bijvoorbeeld 06-GPS*). Hierdoor ontvangt het mobiele apparaat corrigerende data, waarmee grove positioneringsfouten in zijn eigen waarnemingen opgeheven worden. Echter is dit géén realistische oplossing zolang de tarieven voor het gebruik van GPS-referentienetwerken zo hoog zijn.

V 9 Voor welke toepassingen kan Layar als medium ingezet worden?

A 9.a De meest voor de hand liggende toepassing is het weergeven van conceptuele ontwerpen op de geografische ligging hiervan.

A 9.b Een mogelijke toepassing, die niet in dit onderzoek is onderzocht, is het toepassen van Layar als zijnde visualisatietechniek voor realtime data, zoals monitoringsgegevens.

Het is programmeerbaar de SQL-database in realtime te laten updaten. Het bijbehorende PHP-script zou zodanig aangepast kunnen worden dat het weer te geven icoon afhankelijk is van een attribuutwaarde in de SQL-database. Door het categoriseren van deformatiegroottes wordt het visuele icoon veranderd.

V 10 Welke toevoeging geeft Layar vergeleken met een bestekstekening?

A 10 Met behulp van Layar wordt een 3D-ontwerp geprojecteerd als zijnde een augmented realitymodel. De positionering, oriëntering, grootte en het uiterlijk van het ontwerp kan hierdoor objectief benaderd worden. Een groot voordeel in tegenstelling tot een bestekstekening is, dat ook gebruikers die geen bestekstekening kunnen lezen, een goed beeld krijgen van het ontwerp.

E 1 In hoofdstuk 4 is een inventarisatie gedaan betreffende augmented realitysoftware. Ten tijde van de afronding van deze scriptie is gebleken dat AR-media doorontwikkeld is en dat deze sinds 24 september 2012 net als Layar, geogerefereerde augmented realitymodellen ondersteunt. Deze mogelijkheid, gecombineerd met de overige voordelen van AR-media, maakt AR-media qua geogerefereerde 3D-augments op dit moment een beter alternatief dan Layar.

## Bronvermelding

### Bibliografie

- Phone Finder - search for a phone by feature - GSMArena.com. ((n.d.)). (Phone Finder)  
Retrieved april 18, 2012, from GSMArena.com:  
<http://www.gsmarena.com/search.php3>
- Angel, A. (2010, august 1). *sobel-edge-detection | IMAGE PROCESSING*. Retrieved july 26, 2012, from Blogspot.com: <http://angeljohnsy.blogspot.nl/2011/12/sobel-edge-detection.html>
- Christian Doppler Laboratory. ((n.d.)). *History of Mobile Augmented Reality*. Retrieved juli 18, 2012, from Institute for Computergraphics and Vision:  
<https://www.icg.tugraz.at/~daniel/HistoryOfMobileAR/>
- Din, A. I. (1994). *Introduction to SQL, Version 4.76*. Cambridge, Massachusetts, Verenigde Staten: Blackwell Publishers, Inc.
- Frontier Vision. (n.d.). *Frontier Vision technology - Frontier Vision*. Retrieved juni 25, 2012, from Frontier Vision: <https://sites.google.com/site/fvlibrary/frontier-vision>
- Halle, M., Billingham, M., & Thomas, B. (2006). *Emerging Technologies of Augmented Reality: Interface and Design* (1e ed.). New York: Idea Group Inc.
- Hartley, R., & Zisserman, A. (2000-07). *Multiple View Geometry in Computer Vision* (1e ed.). Cambridge, Verenigd Koninkrijk: Cambridge University Press.
- Hirzer, M. (2008, 10 27). *Thesis*. Retrieved juni 25, 2012, from Studierstube.icg.tugraz.at: [http://studierstube.icg.tugraz.at/thesis/marker\\_detection.pdf](http://studierstube.icg.tugraz.at/thesis/marker_detection.pdf)
- Layar. (2011, augustus 23). *Layar Vision - Layar Developer Documentation*. (Layar)  
Retrieved juni 1, 2012, from Layar Vision:  
<http://www.layar.com/documentation/browser/howtos/layar-vision-doc/>
- Layar. (2012, april). *Developer Lists*. Opgeroepen op april 27, 2012, van Layar.com: <http://www.layar.com/publishing/developers/list/>
- Layar Reality Browser. (2009 - 2012). *Tutorials on creating a layer*. Opgeroepen op april - augustus 2012, van Layar.com: [layar.pbworks.com/w/page/25427941/Tutorials on creating a layer](http://www.layar.com/pbworks.com/w/page/25427941/Tutorials%20on%20creating%20a%20layer)
- Meridianworlddata.com. (1998-2007). *Distance Calculation latitude longitude global database lists*. Retrieved juni 8, 2012, from Meridian World Data:  
<http://www.meridianworlddata.com/Distance-Calculatation.asp>
- Murray, J. D., & Van Ryper, W. (1996-05). *Encyclopedia of Graphics File Formats* (2e ed.). Montreal, Quebec, Canada: O'Reilly Media.
- the PHP Documentation Group. (1997 - 2012). *PHP: PHP Manual*. Opgeroepen op mei 25, 2012, van PHP.net: [www.php.net/manual/en/index.php](http://www.php.net/manual/en/index.php)
- Wikipedia.org. (2012, Juli 2). *Global Positioning System*. Opgeroepen op Augustus 21, 2012, van Wikipedia:  
[http://nl.wikipedia.org/wiki/Selective\\_Availability#Opzettelijke\\_degradatie\\_van\\_de\\_nauwkeurigheid](http://nl.wikipedia.org/wiki/Selective_Availability#Opzettelijke_degradatie_van_de_nauwkeurigheid)

## Bronvermelding formules

Vergelijking	Berekening	Bron / auteur
5.1	Verwerkingssnelheid hardware	Wikipedia.org
7.1	Conversie graden naar kilometers	Meridianworlddata.com
7.2	Afstand uit WGS84 coördinaten	Meridianworlddata.com
8.1	Aantal pixels op scherm	Layar
8.2	Rekenhoogte t.b.v. pixelberekening	Layar
9.1	Berekening gemiddelde helderheidsdimensie	Wikipedia.org
9.2	Projectiematrix	Richard Hartley, Andrew Zisserman
9.3	Pixelgrootte	Richard Hartley, Andrew Zisserman
9.4	Rotatiematrix	Wikipedia.org

## Brondvermelding tabellen

Tabel	Omschrijving	Bron / auteur / datum
4.1	Inventaris augmented realitysoftware	Zoekresultaten op 18 april 2012
6.1	Inventaris support en diensten	Zoekresultaten Layar-website
7.1	Foutcodes JSON antwoord	Layar-website
8.1	OBJ, definitie vertexpunten	Wavefront
8.2	OBJ, definitie 3D-face vorming	Wavefront
8.3	MTL, definities materiaal en kleur	Wavefront
B1.1	Inventaris mobiele apparatuur	Zoekresultaten GSMArena.com

## Bronvermelding afbeeldingen

Figuur	Omschrijving	Auteur (gebruikte software)
in scriptie		
3.1	Reflectorzicht principe	Mark Vrielink (AutoCAD)
4.1	Logo: Layar	Layar
4.2	Logo: Adobe Flash logo	Adobe
7.1	Schema vraagstelling Layarapplicatie	Mark Vrielink (AutoCAD)
7.2	Databasestructuur Layar –server	Mark Vrielink (SQLyog)
7.3	Interactiestroom, bevraging Layarserver	Mark Vrielink (AutoCAD)
8.1	OBJ: vertexpunten	Mark Vrielink (AutoCAD)
8.2	OBJ: facevorming	Mark Vrielink (AutoCAD)
8.3	MTL: opmaakdefinities	Mark Vrielink (Layar 3D model converter)
8.4	Kubusmodelprincipe	Mark Vrielink (AutoCAD)
9.1	Herkenningmarker	Mark Vrielink (AutoCAD)
9.2	Randdetectie Sobe gradiënt i.c.m. thresholding	Mark Vrielink (Matlab)
9.3	HSL kleurenmodel	Wikimedia.org
9.4	Thresholding-principe	Mark Vrielink (Excel)
9.5	Thresholding-analyse	Mark Vrielink (Excel)
9.6	Perspectief principe	Mark Vrielink (AutoCAD)
9.7	Camera-coördinatenstelsel	Mark Vrielink (AutoCAD)
9.8	Geometrische transformaties	Mark Vrielink (AutoCAD)
9.9	Projectie van 2D afbeelding in een 3D ruimte	Mark Vrielink (AutoCAD)
9.1	Beeldverschuiving	Mark Vrielink (AutoCAD)
10.1	Screenshot positie landhoofd spoorbrug	Mark Vrielink (Layar 3D model converter)
10.2	Screenshot landhoofd-augment in Layar	Mark Vrielink (Layar applicatie)
10.3	Overzichtskaart plaatsing augment Station Vijzelgracht	Mark Vrielink (Layar 3D model converter)
10.4	Screenshot augment Station Vijzelgracht (open terrein)	Mark Vrielink (Layar applicatie)
10.5	Screenshot augment Station Vijzelgracht (tussen bebouwing)	Mark Vrielink (Layar applicatie)
11.1	Vervorming relatieve hoogte	Mark Vrielink (AutoCAD)
11.2	Screenshot riolerings-augment in Layar	Mark Vrielink (Layar applicatie)
11.3	Screenshot augment riolering (vervormingseffect)	Mark Vrielink (Layar applicatie)
in bijlagen		
B1.1	Visualisatie inventaris mobiele apparatuur	Mark Vrielink (AutoCAD)
B2.1	JSON bericht (schematisch)	Mark Vrielink (AutoCAD)
B3.1	Censuur   Kantoorgebouw	Witteveen+Bos (AutoCAD)
B3.2	Terreinmodel	Mark Vrielink (Bryce7)
B3.3	Censuur   Origineel draadmodel in AutoCAD	Mark Vrielink (AutoCAD)
B3.4	Censuur   OBJ/MTL Trimble SketchUp	Mark Vrielink (Layar 3D model converter)
B3.5	Censuur   OBJ/MTL Bryce7	Mark Vrielink (Layar 3D model converter)
B3.6	Conversie volgorde naar L3D	Mark Vrielink (AutoCAD)
B3.7	Layar 3D Model converter   Overview	Mark Vrielink (Layar 3D model converter)
B3.8	Layar 3D Model converter   Materials	Mark Vrielink (Layar 3D model converter)
B3.9	Censuur   Layar 3D Model converter   Preview	Mark Vrielink (Layar 3D model converter)
B3.10	Layar 3D Model converter   Placement	Mark Vrielink (Layar 3D model converter)
B3.11	Vorming SQL-database in phpMyAdmin	Mark Vrielink (Screenshot phpMyAdmin)
B3.12	Registratie augmentgroep in Layar	Mark Vrielink (Layar, augmentregistratie)
B3.13	Screenshot Layar testconsole	Mark Vrielink (Layar.com)

# Bijlagen

Ter completering van de scriptie zijn de volgende documenten toegevoegd.

- B1(a) Inventaris bruikbare mobiele apparatuur
- B1(b) Grafische kwaliteit mobiele apparatuur
- B2 Optimaliseren van datasets (*beschrijving van alle tabellen met bijbehorende attributen*)
- B3 Genereren & Publiceren (*workflow voor het maken en publiceren augments*)

B1(a) Inventaris bruikbare mobiele apparatuur <sup>(1/2)</sup>

De onderstaande tabel (vervolg zie volgende pagina) betreft de inventaris van voor Layar-bruikbare-apparatuur. De lijst is gesorteerd op de verwerkingssnelheid van de apparaten. Er wordt géén onderscheid gemaakt in enkel-, dual- of quad-core processor aangezien slechts één core gebruikt wordt bij het uitvoeren van één applicatie.

Rang	Merk	Uitvoering	OS	Scherm- resolutie	Camera		Processor	Verwerking				RAM	
					Foto	F/s		bus Hz	bit	cyclus	MB/S		
1	Apple	The New iPad 4G	iOS	2048x1536	5 MP	30	1 GHz dual-core	250	MHz	64	9	18000	1024 MB
	Apple	iPad 2 3G	iOS	1024x768	0,7 MP	30	1 GHz dual-core	250	MHz	64	9	18000	512 MB
2	HTC	One S	Android	540x960	8 MP	30	1,50 GHz dual-core	100	MHz	64	9	7200	1024 MB
	HTC	Evo 3D	Android	540x960	5 MP	30	1,20 GHz dual-core	100	MHz	64	9	7200	1024 MB
	HTC	Rezound	Android	720x1280	8 MP	60	1,50 GHz dual-core	100	MHz	64	9	7200	1024 MB
	HTC	One X	Android	720x1280	8 MP	30	1,50 GHz quad-core	100	MHz	64	9	7200	1024 MB
	Huawei	Ascend D	Android	720x1280	8 MP	30	1,50 GHz dual-core	100	MHz	64	9	7200	1024 MB
	LG	Prada	Android	480x800	8 MP	30	1 GHz dual-core	100	MHz	64	9	7200	1024 MB
	LG	Nitro HD	Android	720x1280	8 MP	30	1,50 GHz dual-core	100	MHz	64	9	7200	1024 MB
	Samsung	Galaxy Nexus	Android	720x1280	5 MP	30	1,20 GHz dual-core	100	MHz	64	9	7200	1024 MB
	Samsung	I9100 Galaxy S II	Android	480x800	8 MP	30	1,20 GHz dual-core	100	MHz	64	9	7200	1024 MB
	Samsung	Infus 4G	Android	480x800	8 MP	30	1,20 GHz dual-core	100	MHz	64	9	7200	1024 MB
	Sony	Ericsson Xperia S	Android	720x1280	12 MP	30	1,50 GHz dual-core	100	MHz	64	9	7200	1024 MB
	HTC	Sensation	Android	540x960	8 MP	30	1,20 GHz dual-core	100	MHz	64	9	7200	768 MB
	BlackBerry	Bold 9930	BlackBerry 7 OS	640x480	5 MP	24	1,20 GHz	100	MHz	64	9	7200	768 MB
	BlackBerry	Bold 9900	BlackBerry 7 OS	480x800	5 MP	24	1,20 GHz	100	MHz	64	9	7200	768 MB
	BlackBerry	Torch 9850	BlackBerry 7 OS	480x800	5 MP	24	1,20 GHz	100	MHz	64	9	7200	768 MB
	BlackBerry	Torch 9860	BlackBerry 7 OS	480x800	5 MP	24	1,20 GHz	100	MHz	64	9	7200	768 MB
	Apple	iPhone 4	iOS	960x640	5 MP	30	1 GHz	100	MHz	64	9	7200	512 MB
	Apple	iPhone 4S	iOS	960x640	8 MP	30	1 GHz dual-core	100	MHz	64	9	7200	512 MB
	Huawei	U8860 Honor	Android	480x854	8 MP	30	1,40 GHz	100	MHz	64	9	7200	512 MB
	LG	Optimus 2x	Android	480x800	8 MP	24	1 GHz dual-core	100	MHz	64	9	7200	512 MB
	LG	Optimus 3D P920	Android	480x800	5 MP	30	1 GHz dual-core	100	MHz	64	9	7200	512 MB
	LG	Optimus Black P970	Android	480x800	5 MP	30	1,40 GHz	100	MHz	64	9	7200	512 MB

Bron: (Phone Finder - search for a phone by feature - GSMArena.com, (n.d.))

## B1(a) Bruikbare mobiele apparatuur <sup>(2/2)</sup>

Rang	Merk	Uitvoering	OS	Resolutie	Camera		Processor	Verwerking				RAM	
					Foto	F/s		bus Hz	bit	cyclus	MB/S		
2	LG	Optimus Dual	Android	320x480	3,15 MP	24	800 MHz	100	MHz	64	9	7200	512 MB
	Samsung	Google Nexus S	Android	480x800	5 MP	30	1 GHz	100	MHz	64	9	7200	512 MB
	Samsung	Captivate	Android	480x800	5 MP	30	1 GHz	100	MHz	64	9	7200	512 MB
	Samsung	Galaxy S	Android	480x800	5 MP	30	1 GHz	100	MHz	64	9	7200	512 MB
	Samsung	Galaxy S Vibrant	Android	480x800	5 MP	30	1 GHz	100	MHz	64	9	7200	512 MB
	Samsung	Galaxy W	Android	480x800	5 MP	30	1,40 GHz	100	MHz	64	9	7200	512 MB
	Sony	Ericsson Xperia ray	Android	480x854	8 MP	30	1 GHz	100	MHz	64	9	7200	512 MB
3	BlackBerry	Torch 9810	BlackBerry 7 OS	480x640	5 MP	24	1,20 GHz	100	MHz	32	6	2400	768 MB
	BlackBerry	Torch 9800	BlackBerry 7 OS	360x640	5 MP	24	624 MHz	100	MHz	32	6	2400	512 MB
	Nokia	N8	Symbian	360x640	12 MP	25	680 MHz	100	MHz	32	6	2400	512 MB
	Apple	iPhone 3GS	iOS	640x480	3 MP	30	600 MHz	100	MHz	32	6	2400	256 MB
	Nokia	C6-01	Symbian	360x640	8 MP	25	680 MHz	100	MHz	32	6	2400	256 MB
	Nokia	C7	Symbian	360x640	8 MP	25	680 MHz	100	MHz	32	6	2400	256 MB
	Nokia	E7	Symbian	360x640	8 MP	25	680 MHz	100	MHz	32	6	2400	256 MB
	Nokia	N97	Symbian	360x640	5 MP	30	434 MHz	100	MHz	32	6	2400	128 MB

Bron: (Phone Finder - search for a phone by feature - GSMArena.com, (n.d.))

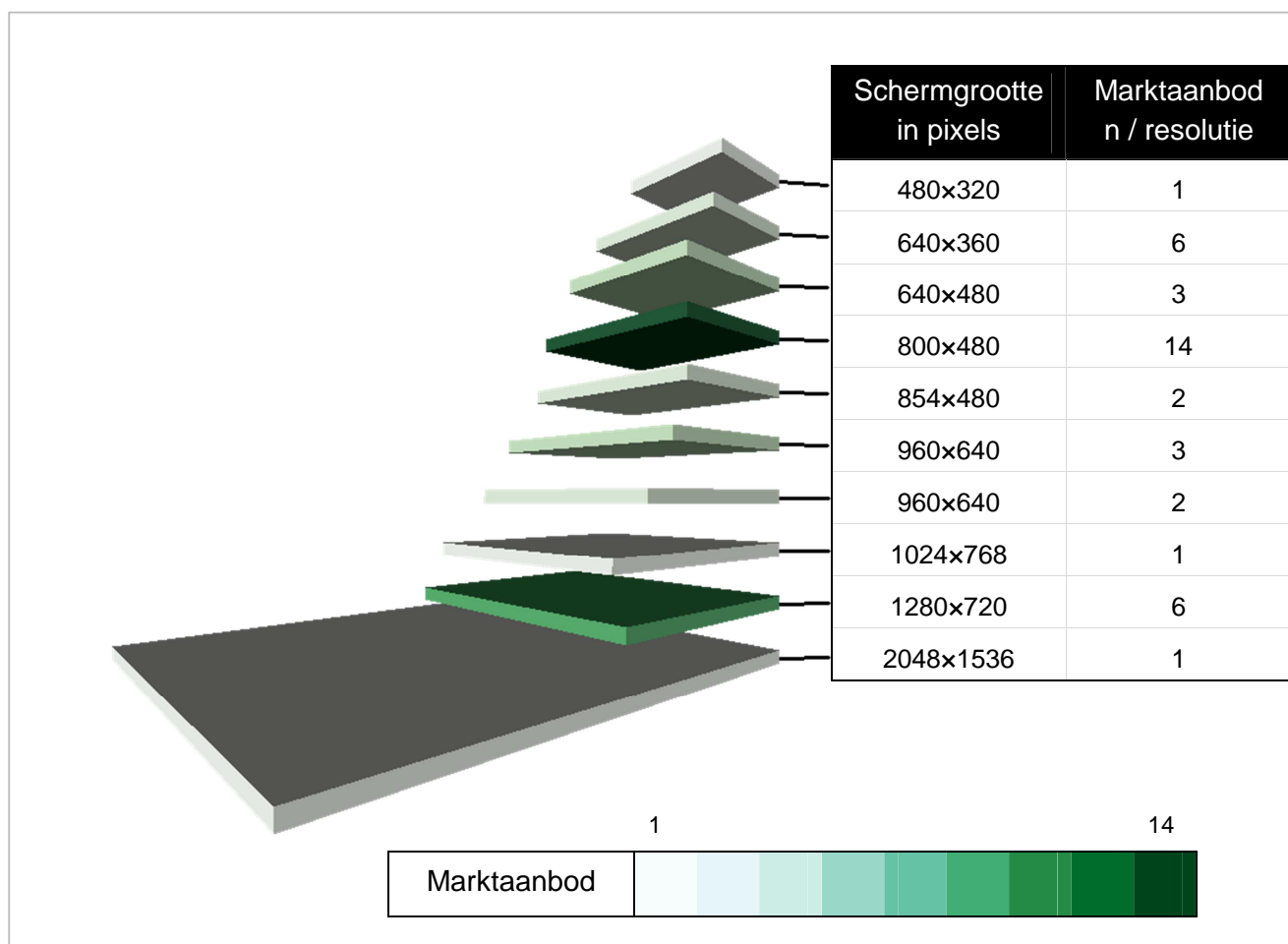
Tabel B1.1

Uit de inventaris “mobiele apparatuur” is geconcludeerd dat de apparaten “iPad 2 3G” en “The New iPad 4G” van Apple qua hardware de hoogste prestaties kunnen leveren, waardoor deze automatisch de meest geschikte apparaten zijn voor het uitvoeren van de Layaapplicatie.

## B1(b) Grafische kwaliteit mobiele apparatuur

Uit de inventaris mobiele apparatuur in bijlage B1(a) is in figuur B1.1 het aanbod van mobiele apparatuur, waarop Layar fungeert, inzichtelijk gemaakt.

Om een goede grafische weergave te verkrijgen is het van belang, dat de resolutie van deze apparaten bekend is. Een apparaat met een hoge schermresolutie kan een hoger detailniveau afbeelden.



Figuur B1.1



## B2 OPTIMALISEREN VAN DATASETS

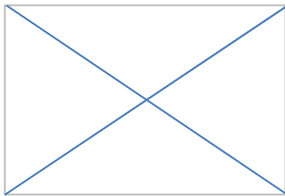
Deze bijlage geeft een beschrijving van alle in de SQL-database voorkomende tabellen inclusief attributen en de daarbij behorende eigenschappen en functies.

Waar in Hoofdstuk D de gehele anatomie van de SQL-database beschreven wordt, komt in dit hoofdstuk de inhoud van die betreffende database aan bod.

### B2.1 Excel invulblad

Om efficiënter een nieuw augmented realityproject aan te maken en te vullen is er een spreadsheet ontwikkeld die *(voor zover dat kan)* automatisch attribuutinformatie toekent aan de records. Veel attributen zijn van elkaar afhankelijk doordat bijvoorbeeld een audio-bestand een vast contenttype betreft.

Van alle ingevulde en automatisch ingevulde attributen wordt een SQL-script gevormd dat online door de phpMyAdmin service ingelezen kan worden. Tevens kunnen het serverscript en het verificatiescript met behulp van dit bestand gemaakt worden.



Met dit bestand kan alle benodigde code (*SQL en PHP*) gevormd worden om een eigen augmentserver te realiseren.

Alle invulvelden die in dit bestand voorkomen, worden beschreven in de bijlage B2.2

## B2.2 Tabel en attribuutdefinities

In deze bijlage een verklaring van alle tabellen en bijhorende attributen.

Koppelingstype

1 op meer<sup>(∞)</sup> of meer<sup>(∞)</sup> op 1

V= verplicht

O = optioneel

### B 2.2.1 POI-tabel

POI	Omschrijving	Invoerwaarde	Type	
id	Unieke naam	tekst	sleutelwaarde	V
footnote	Voetregel	tekst	tekst	O
title	Titel	tekst	tekst / zoekwaarde	V
lat	breedtegraad	decimaal getal	WGS84-coördinaat	V
lon	lengtegraad	decimaal getal	WGS84-coördinaat	V
imageURL	Augment voorvertoning	http://	URL	O
description	Augment omschrijving	tekst	tekst / zoekwaarde	V
biwStyle	Menustijl	collapsed / classic	Menu-instelling	V
doNotIndex	(beter) vindbaar worden	0 of 1	Indexeerattribuut	V
showSmallBiw	Menustijl (klein of groot)	0 of 1	Menu-instelling	V
showBiwOnClick	Menuweergave bij actie	0 of 1	Menu-instelling	V
poiType	Type poi	geo	geogerefereerd	V
iconID	Koppeling met icon-tabel	getal	sleutelwaarde	V
objectID	Koppeling met object-tabel	getal	sleutelwaarde	V
transformID	Koppeling met transform-tabel	getal	sleutelwaarde	V
referenceImage	Afbeelding LayarVision	http://	URL	O
layerID	Koppeling met layer-tabel	getal	sleutelwaarde	V

De POI-tabel (*Places of Interest*) betreft de hoofdtabel waaraan alle volgende tabellen gekoppeld zijn. Deze tabel bevat alle attributen waarop gezocht kan worden, zowel geografisch als tekstueel.

De zoekattributen in deze tabel betreffen: “footnote”, “title”, “lat”, “lon”, “description” en optioneel “referenceImage”. Het attribuutveld “doNotIndex” geeft aan of Layar dit augment mag indexeren, zodat deze beter vindbaar wordt. 0 betreft wél indexeren.

De attributen “biwStyle”, “showSmallBiw” en “showBiwOnClick” betreffen parameters voor het informatiekader. Hoe moet het kader zich in de betreffende situatie gedragen? Biw staat dan ook voor Brief Information Widget. De informatie, die hierin weergegeven wordt, betreffen de attributen: “footnote”, “title”, “description” en de afbeelding waarnaar verwezen wordt in het attribuut: “imageURL”.

Het attribuut “poiType” is bij Witteveen+Bos altijd gevuld met de waarde “geo” wat staat voor een geogerefereerde augment. Als deze waarde in “vision” wordt veranderd, kan een augment op elke willekeurige plek op aarde (*mits voorzien van een internetverbinding*) bekeken worden doordat deze relatief ten opzichte van de gebruiker geprojecteerd zal worden.

De overige (*gekleurde*) attributen betreffen sleutelattributen die een koppeling vormen tussen de overige tabellen.

### B2.2.2 POIAction-tabel

POIAction	Omschrijving	Invoerwaarde	Type	
poiID	1			
label	Actietitel	tekst	tekst	O
uri	Bestandslocatie actie	afhankelijk van type actie: http://, mailto: of tel:	URL	V
autoTriggerRange	automatische actie-uitvoer binnen afstand	getal / n Meters	parameter	O
autoTriggerOnly	Altijd automatische actie of afhankelijk van autoTriggerRange	0 of 1	parameter	V
id	Recordnummer	Getal	id nummer	V
contentType	application/vnd.layar.internal = website, e-mail, telefoonnummer audio/mpeg = audio (mp3/mpeg) video/mp4 = video (mp4)	application/vnd.layar.internal, audio/mpeg of video/mp4	parameter	V
method	GET = website, e-mail, audio video POST = telefoonnummer	GET of POST	parameter	V
activityType	1 = website (http://) 2 = audio (mp3/mpeg), (http://) 3 = video (mp4) (http://) 4 = telefoonnummer (tel:) 5 = e-mail (mailto:)	1,2,3,4 of 5	parameter	V
params	locatieafhankelijke actie	lat,lon	parameter	V
closeBiw	Gesloten actie-infovenster	0 of 1	parameter	V
showActivity	verborgen of zichtbare activiteit	0 of 1	parameter	V
activityMessage	Pop-up bericht bij actie	tekst	tekst	O
autoTrigger	Automatische actie	0 of 1	parameter	V

In de POIAction-tabel kunnen acties toegekend worden aan een augment. Deze acties betreffen het openen van een website, het afspelen van een audio- of videobestand of een met een hyperlink voorzien telefoonnummer of e-mailadres.

Er kunnen meerdere acties gekoppeld worden aan één augment / POI. Doordat elk type actie unieke eigenschappen bevat, moet het betreffende content ("*contentType*") en de bijbehorende actie ("*activityType*" en "*method*") gedefinieerd worden. Het type content is ook bepalend voor de adresopbouw ("*uri*") van de content.

Bestanden als audio- en videomateriaal moeten, voordat het wordt afgespeeld, eerst gedownload worden. De zijn gehost op een publieke server, daarom is de bestandslocatie opgebouwd uit een http:// (*Hypertext Transfer Protocol*) adres.

Tevens moet gedefinieerd worden wanneer een actie uitgevoerd dient te worden binnen een bepaalde afstand vanaf het plaatsingspunt ("*autoTriggerRange*" en "*params*") of alleen wanneer het apparaat gericht staat op het augment ("*autoTriggerOnly*"). Het attribuut "*autoTrigger*" bepaalt of de actie alleen weergegeven of ook daadwerkelijk uitgevoerd dient te worden. Het attribuut "*activityMessage*" is een optioneel informatielabel dat verschijnt voordat de actie uitgevoerd wordt. Het attribuut "*closeBiw*" betreft de instelling of het menu in- of uitgeklapt in het scherm moet verschijnen.

“showActivity” kan alleen op “0” gezet worden wanneer de actie het afspelen van audio betreft. Alle overige acties gaan samen met een schermweergave en zouden géén toevoeging geven wanneer deze verborgen uitgevoerd zouden worden.

### B2.2.3 Icon-tabel

Icon	Omschrijving	Invoerwaarde	Type	
id	Koppeling met POI-tabel	getal	sleutelwaarde	V
url	Bestandslocatie JPG/PNG	http://	URL	O
type	Type media ( <i>afbeelding</i> =)	1	parameter	O

De Icon-tabel is een weergave-instelling voor augments op grotere afstanden. Wanneer de gebruiker zich op een verder dan een vooraf gedefinieerde afstand (*zie figuur 8.4*) van de augment bevindt, wordt het beeld van het 3D-model vervangen door het pictogram dat gedefinieerd is in de Icon-tabel.

Het attribuut “type” beschrijft het type media dat voor een icoon altijd een afbeelding betreft. Bij Layar is een afbeelding gedefinieerd als typewaarde: 1. Het “url”-attribuut betreft de bestandsverwijzing naar dit pictogram.

### B2.2.4 Layer-tabel

Layer	Omschrijving	Invoerwaarde	Type	
layer	Omschrijving layer	tekst	tekst	O
refreshInterval	Updatetijd	getal / <i>n</i> Seconden	parameter	V
refreshDistance	Updateafstand	getal / <i>n</i> Meters	parameter	V
fullRefresh	Definities worden ingeladen	getal / <i>n</i> Minuten	parameter	V
showMessage	Berichtgeving na verversen	tekst	tekst	O
id	Koppeling met POI-tabel	getal	sleutelwaarde	V
biwStyle	Menustijl	collapsed / classic	Menu-instelling	V

In de Layer-tabel wordt de interval van modelupdates gedefinieerd. De weergave van een augment betreft een gerenderde afbeelding van het 3D / L3D model. Een nieuwe weergave wordt automatisch verzorgd wanneer men zich een *n* seconden niet verplaatst (“*refreshInterval*”), of zich meer dan *n* meters verplaatst (“*refreshDistance*”). Een volledige nieuwe databasebevraging wordt uitgevoerd wanneer de gebruiker hetzelfde augment langer dan *n* minuten (“*fullRefresh*”) op het apparaat open heeft staan.

Een update kan met een ingegeven tekst (“*showMessage*”) gemeld worden in de vorm van een pop-up informatiekader (“*biwStyle*”).

### B2.2.5 Transform-tabel

Transform	Omschrijving	Invoerwaarde	Type	
id	Koppeling met POI-tabel	getal	sleutelwaarde	V
rel	Relatieve transformatie	true / false	parameter	V
angle	rotatiehoek	getal / nGraden	parameter	V
rotate_x	rotatie ( <i>angle</i> ) actief op as	0 of 1	parameter	V
rotate_y	rotatie ( <i>angle</i> ) actief op as	0 of 1	parameter	V
rotate_z	rotatie ( <i>angle</i> ) actief op as	0 of 1	parameter	V
translate_x	Modelverplaatsing in X-as	getal / n Meters	parameter	V
translate_y	Modelverplaatsing in Y-as	getal / n Meters	parameter	V
translate_z	Modelverplaatsing in Z-as	getal / n Meters	parameter	V
scale	Schaal van het model	Getal	parameter	V
Description	Omschrijving transformeer-record	Tekst	tekst	V

Wanneer men het in de database gedefinieerde augment wil bekijken met behulp van de Layaapplicatie, kan het voorkomen dat het augment verkeerd gepositioneerd wordt in het camerabeeld. Wanneer de gebruiker in het bezit is van twee mobiele apparaten (*één voor het bekijken van het augment, en één voor het benaderen / muteren van de database*) kan het augment in realtime getransformeerd worden tot een correcte inpassing.

Wanneer het augment verkeerd geroteerd is, dient bij “angle” de draaihoek opgegeven te worden. Deze draaihoek heeft effect op de rotatie-as(sen) die gedefinieerd zijn als 1 (“rotate\_x”, “rotate\_y” en/of “rotate\_z”). Naast een rotatie kan het model tevens in de X- (*translate\_x*), Y- (*translate\_y*) of Z- (*translate\_z*) richting verplaatst worden. Deze verplaatsing is altijd in de richting van de geografische X-, Y- en / of Z-as, ongeacht een eventueel gedefinieerde rotatie.

Verder kan het augment verschaald worden (“scale”) en kan worden ingesteld dat het augment altijd relatief ten opzichte van de gebruiker getoond wordt (“rel”). De transformatie-instelling dient voorzien te worden van een omschrijving (*de omschrijving is alleen zichtbaar in de database, niet wanneer het model benaderd wordt via de Layaapplicatie*).

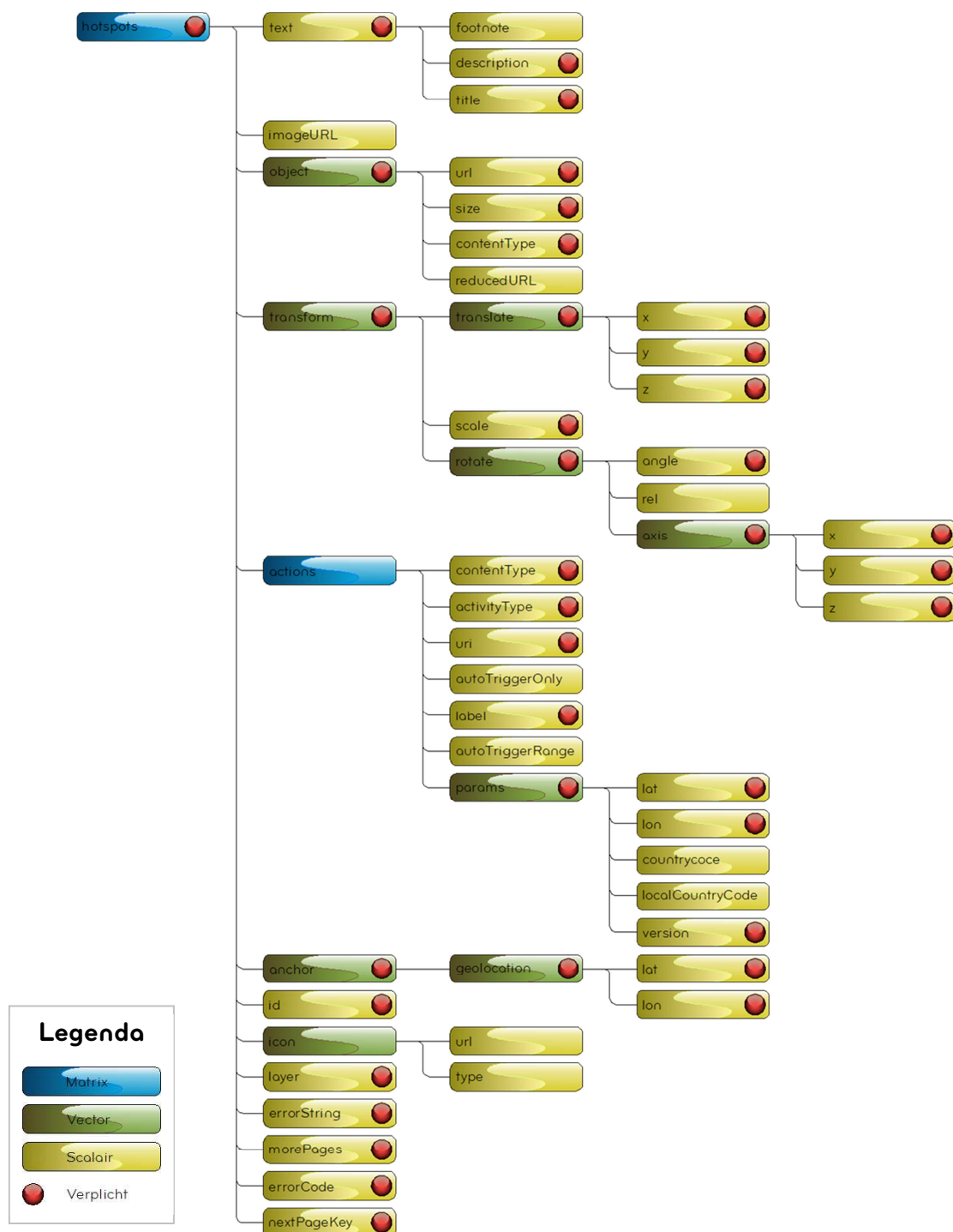
### B2.2.6 Object-tabel

Object	Omschrijving	Invoerwaarde	Type	
id	Koppeling met POI-tabel	Getal	sleutelwaarde	V
contentType	Type inhoud (L3D)	model/vnd.layar.l3d	parameter	V
url	Bestandslocatie L3D-bestand	http://	URL	V
reducedURL	Versimpeld L3D-bestand	http://	URL	O
size	Objectgrootte	getal	parameter	V

De Object-tabel voorziet de database van het fysieke 3D-model. Optioneel is het om twee augments te definiëren onder één record. Het model dat verwezen is in attribuut “url” betreft het meest gedetailleerde L3D-bestand. Optioneel is om onder “reducedURL” een (*versimpelde*) tweede L3D-bestand te definiëren dat zichtbaar wordt wanneer men buiten de vooraf gedefinieerde afstand van het gedetailleerde model treedt. Het attribuut “size” is een bepalende parameter voor de weergave-afstand van de modellen en wordt concreet uitgelegd in hoofdstuk 8 (zie figuur 8.4).

## B2.3(a) JSON Response

Het JSON bericht wordt zoals in B2.2 weergegeven, gevormd en verzonden naar het mobiele apparaat.



Figuur B2.1

## B2.3(b) JSON Response

JSON Response (deel 1 van 2)

### B2.3(b) JSON Response <sup>(vervolg)</sup>

JSON Response (deel 2 van 2)



# B3 GENEREREN & PUBLICEREN

Deze instructie is een bijlage van de afstudeerscriptie “Augmented Reality voor Bouw & Infra” en is opgezet doordat er problemen zijn ervaren, met name voor de modelopbouw van een augment. Deze instructie heeft alleen betrekking op het maken, converteren, georefereren en publiceren van augments voor de Layarapplicatie.

De instructie betreft kort samengevat de volgende onderdelen:

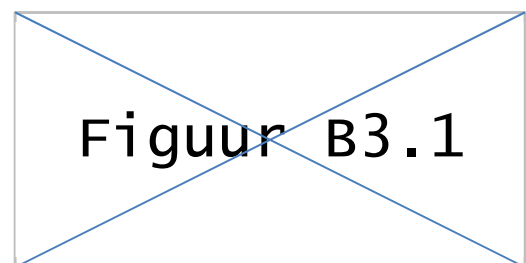
- keuze van softwareprogrammatuur;
- ontwerpeisen 3D model;
- conversieprocessen;
- implementatie in een eigen database;
- geografische inpassing.

Voordat de productie van de augment begint, dient er het besef te zijn dat de Layarapplicatie gemaakt is voor de visualisatie van relatief simpele 3D-modellen (*dat wil zeggen +/- 7000 faces*). De geografische afmeting van het augment speelt hierin géén rol.

## Geschied:

Objectgericht: gebouwen, civieltechnische kunstwerken, utiliteitsbouw, (*ondergrondse*) infrastructuur (*oftewel allemaal praktisch rechte (orthogonale) objecten*).

*Het kantoorgebouw in figuur B3.1 bevat 3825 faces. Dit betreffen alleen de faces die vanaf de buitenkant van het model zichtbaar zijn (orbit-model).*

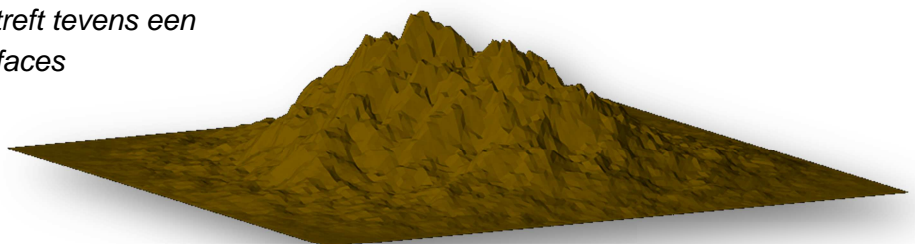


Figuur B3.1

## Ongeschied:

Terreingericht: landschapsarchitectuur, triangulatiemodellen overstijgen nagenoeg altijd 7000 faces.

*Het hoogtemodel in B3.2 betreft tevens een orbit-model en bevat 28800 faces*



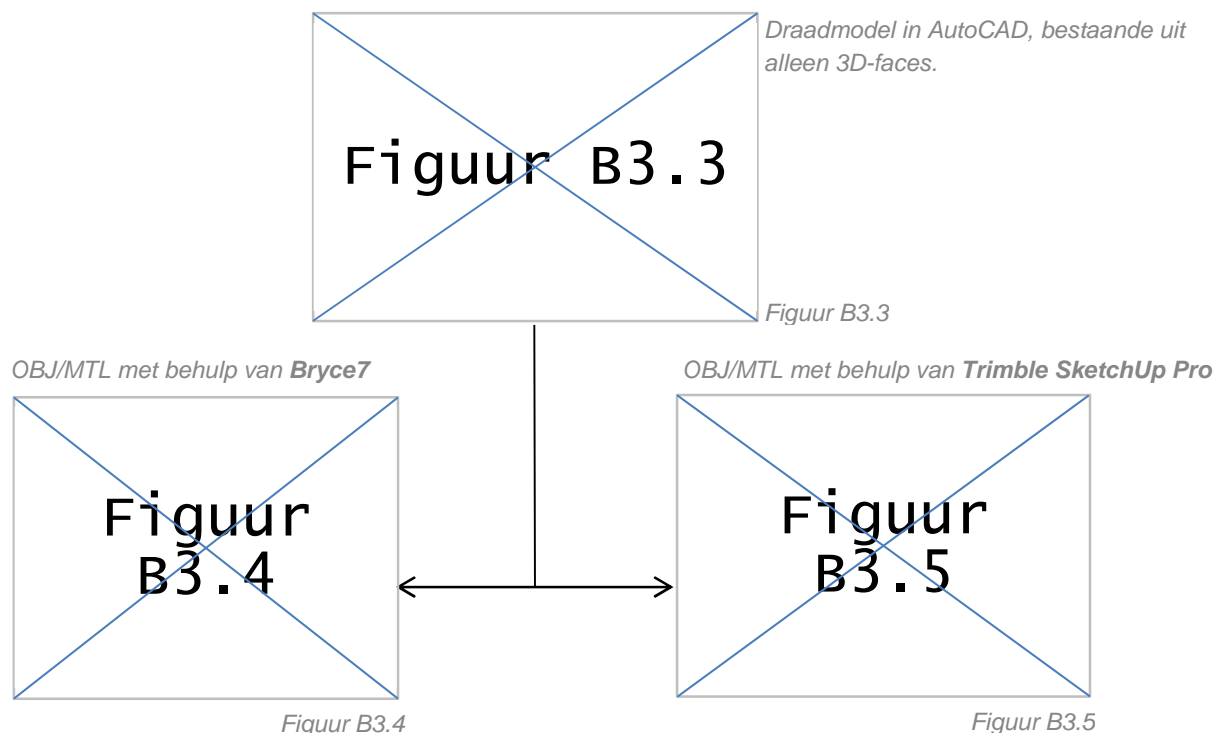
Figuur B3.2

### B3.1 Ontwerpsoftware

Voor de keuze van de ontwerpsoftware is het bepalend dat deze direct dan wel via een omweg een conversie moet kunnen maken naar een \*.OBJ/\*.MTL-formaat (zie § 8.1 *Wavefront's OBJ* en §8.2 *Wavefront's MTL*). Het OBJ/MTL-formaat is het enige dat met de Laya 3D Model Converter geconverteerd kan worden naar het verplichte L3D-formaat. Software waarbij dit onder andere mogelijk is betreffen:

- AutoCAD 3DS Max
- Blender (gratis)
- Bryce7 (gratis)
- Trimble SketchUp Pro (voorheen Google SketchUp Pro)

Ondanks dat Trimble SketchUp Pro van de bovenstaande software als enige een “two-sided-export” naar OBJ/MTL kan maken, is deze software af te raden, doordat de opmaakdefinities van de faces op een negatieve manier sterk worden beïnvloed. Bryce7 geeft daarentegen tenminste van de genoemde software grafisch het beste resultaat in OBJ/MTL (zie de figuren B3.4 en B3.5), doordat de originele kleuren ook na de conversie behouden blijven.

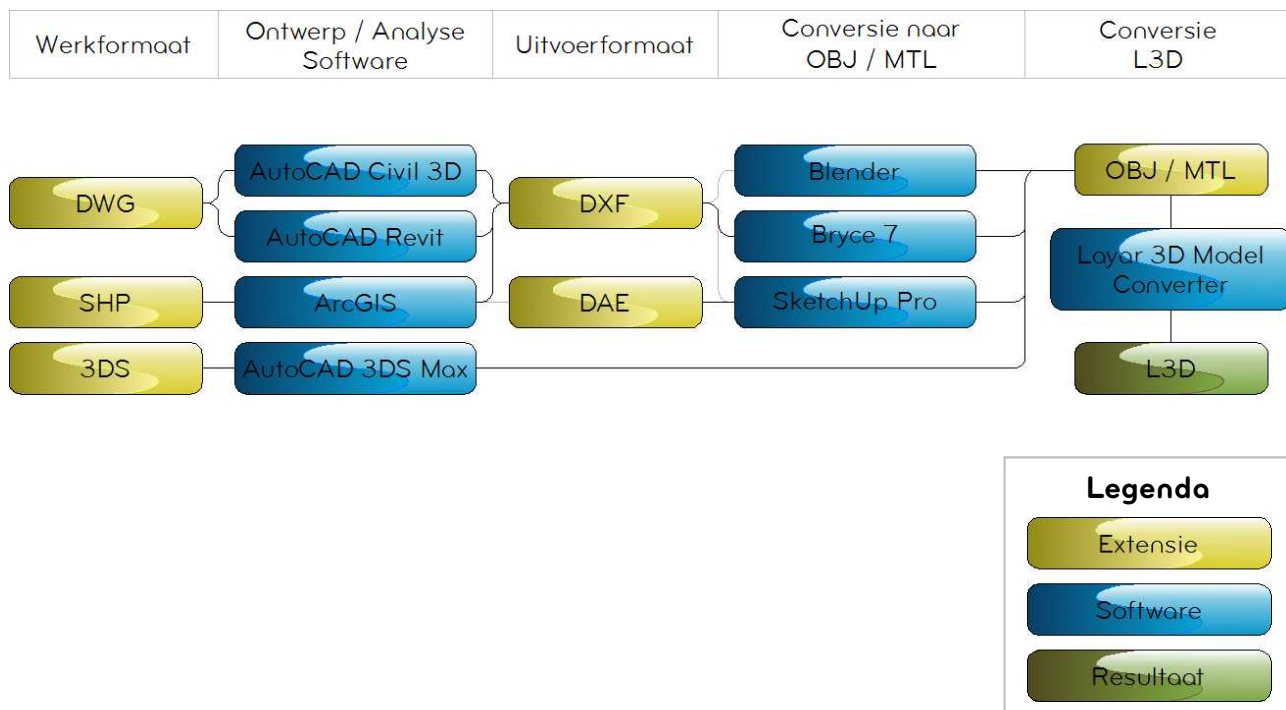


In bovenstaand voorbeeld is als eerste het gehele model in AutoCAD omgezet naar 3D-faces. Om een model zonder onnodige vervuiling van 3D-faces te realiseren, is het nagenoeg altijd noodzakelijk grote gedeelten van het model handmatig over te trekken.

Onnodige vervuiling staat voor 1. gebruik van kleine faces terwijl deze niet noodzakelijk zijn; 2. Faces die elkaar snijden (*niet op elkaar aansluiten*).

### B3.1.1 Conversieproces

In onderstaand schema (*figuur B3.6*) een overzicht van de meest voor de hand liggende softwareprogrammatuur en het bijbehorende proces om tot het uiteindelijke L3D-formaat te komen. Let op dat wanneer men tot de eerste conversie overgaat, het bestand alleen nog uit 3D-faces met drie of vier vertexpunten mag bestaan.



Figuur B3.6

## B3.2 Ontwerpvoorwaarden

### B3.2.1 Algemeen

- Stel géén schaduw / lichtbron in, deze wordt niet ondersteund door Layar.
- Probeer bij grote modellen de kleine details als deuren en ramen niet driedimensionaal uit te tekenen, maar weer te geven in een textuur.
- Test eerst het “kale” basismodel, voordat texturen aan de faces worden toegekend.
- Maak bij grote modellen ook een versimpelde versie (*reducedURL*). Deze versimpelde versie wordt automatisch geladen wanneer het apparaat niet capabel genoeg is om het gedetailleerde model te laden (*ook op grotere afstanden, wanneer het detailniveau niet meer weergegeven kan worden, wordt dit model door het apparaat gebruikt*).

### B3.2.2 Faces

- Alleen de entiteit “3D-face” wordt in het OBJ/MTL-formaat ondersteund.
- Het aantal vertexpunten per face dient beperkt te worden tot maximaal vier.
- Houd het aantal faces beperkt (*advies  $\leq 7000$* ).
- Verwijder faces die niet door de gebruiker gezien kunnen worden (*maak een orbit-model*).
- Voorkom dat 3D-faces elkaar snijden, laat ze in elkaar overlopen / op elkaar aansluiten.
- In het OBJ/MTL-formaat worden faces volgens de “**right-hand-rule**” getekend. Dit betekent dat een face een voor- en achterkant heeft. De achterkant is niet zichtbaar voor de gebruiker (*zie hoofdstuk 8, § 8.1.1*). Wanneer men naar de zichtbare zijde van de face kijkt, zijn de vertexpunten van de face in de tegengestelde richting van de klok gedefinieerd.

### B3.2.3 Texturen

- Alleen vierkante texturen in de grootte van 64×64, 128×128, 256×256 of 512×512 pixels worden in Layar ondersteund. Tenminste de in deze handleiding genoemde software exporteert tijdens de conversie naar OBJ/MTL, automatisch de eventueel gedefinieerde texturen naar een geaccepteerde afmeting.
- Gebruik zo min mogelijk verschillende texturen.
- Er zijn twee afbeeldingsformaten als textuur bruikbaar, JPEG en PNG. Wanneer men transparantie wil gebruiken is dit alleen mogelijk met PNG.
- Wanneer transparantie aan een textuur wordt toegekend, betekent dit dat het vlak transparant wordt en de echte wereld hier doorheen gezien kan worden. Wanneer er zich een 3D-face achter een transparant gedefinieerd vlak bevindt, dan is deze niet zichtbaar.

### B3.2.4 Oriëntering versterken

Probeer “in het gebied herkenbare patronen” op te nemen in de augment, de gebruiker ervaart hierdoor betere oriëntering (*zeker aan te raden bij ondergrondse infrastructuur en augments in het open veld*).

Open veld: projecteer een vlak met transparantie dat het bestaande maaiveld symboliseert.

Ondergrondse infrastructuur: creëer vanaf het maaiveld de diepte in een virtuele bak, waarin de ondergrondse infra is gelegen. Dankzij de virtuele bak ervaart de gebruiker het besef van de liggingdiepte.

### B3.2.5 AutoCAD

In AutoCAD kunnen alle 3D-solids naar 3D-faces omgezet worden door deze eerst naar een mesh te converteren [**>\_MESHSMOOTH**] alvorens deze te converteren naar 3D-faces [**>\_EXPLODE**]. Deze bewerking is alleen toepasbaar op de volgende entiteiten: 3D-solids, 3D-surfaces, polygon meshes, polyface meshes, regions en gesloten (2D/3D) polylines.

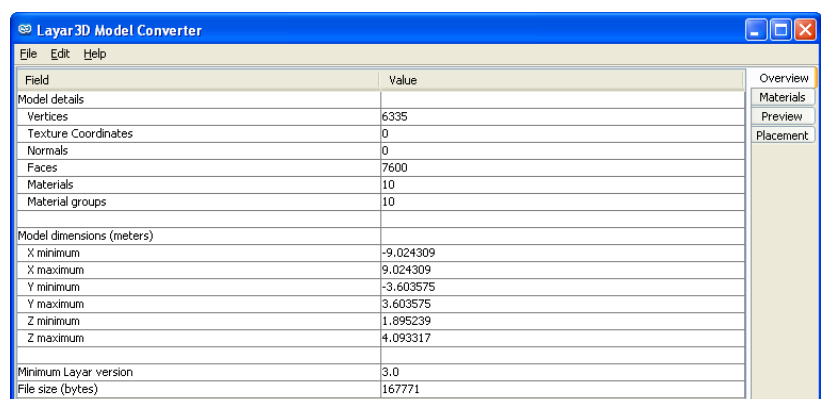
## B3.3 Laya 3D Model Converter

De Layaapplicatie ondersteunt alleen 3D-modellen in \*.L3D-vorm. Om hiertoe te komen heeft Laya een 3D-model-converter ontwikkeld om Wavefront-bestanden (\*.OBJ/\*.MTL) naar \*.L3D te converteren. Deze conversiesoftware is gratis te gebruiken en kan door middel van deze URL (<http://public.layar.com/downloads/Laya3DModelConverter.inlp>) gedownload worden.

Wanneer het OBJ/MTL-model in de Laya 3D Model Converter is geïmporteerd, verschijnt er een venster met vier tabbladen (*Overview, Materials, Preview en Placement*).

### B3.3.1 Overview

Een overview betreft een overzicht met technische de modeleigenschappen: bestandsgrootte, modelafmetingen, aantal vertexpunten, gevormde faces, materialen en materiaalgroepen. Tevens staat vermeld welke minimumversie van de Laya applicatie gebruikt dient te worden om het model te kunnen bekijken.



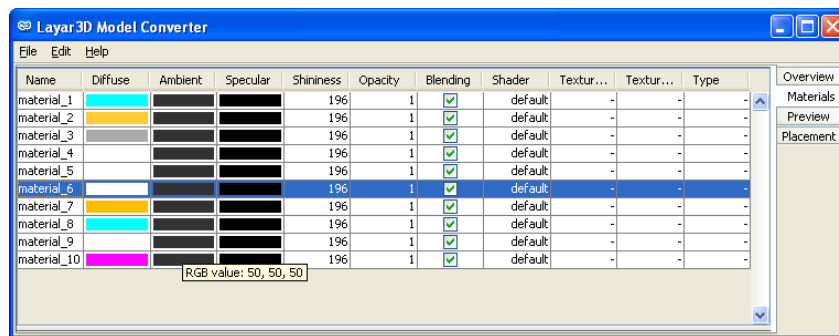
Field	Value
<b>Model details</b>	
Vertices	6335
Texture Coordinates	0
Normals	0
Faces	7600
Materials	10
Material groups	10
<b>Model dimensions (meters)</b>	
X minimum	-9.024309
X maximum	9.024309
Y minimum	-3.603575
Y maximum	3.603575
Z minimum	1.895239
Z maximum	4.093317
Minimum Laya version	3.0
File size (bytes)	167771

Figuur B3.7

### B3.3.2 Materials

Een materiaalgroep is bijvoorbeeld een laag (*layer*) in een CAD-softwarepakket. Een materiaal betreft de kleur of textuuropmaak. Wanneer er geen gebruik wordt gemaakt van texturen én alle componenten volgens de laagdefinities (*ByLayer*) getekend zijn, dan zijn er evenveel materiaalgroepen als materialen. Wordt er wel gebruik gemaakt van texturen dan betreft elk uniek textuur een nieuw materiaal (*géén materiaalgroep*).

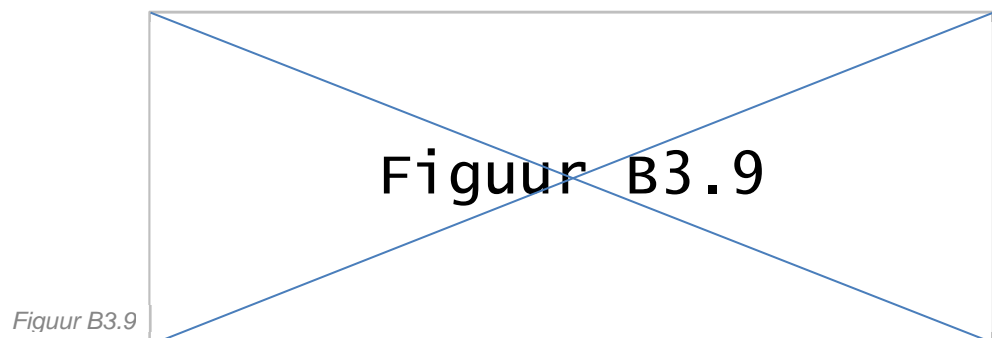
De materiaalweergave kan in dit menu aangepast worden, waarna de mutatie direct in het tabblad "Preview" te controleren is. In hoofdstuk 8, § 8.2 worden de opmaakdefinities nader omschreven.



Figuur B3.8

### B3.3.3 Preview

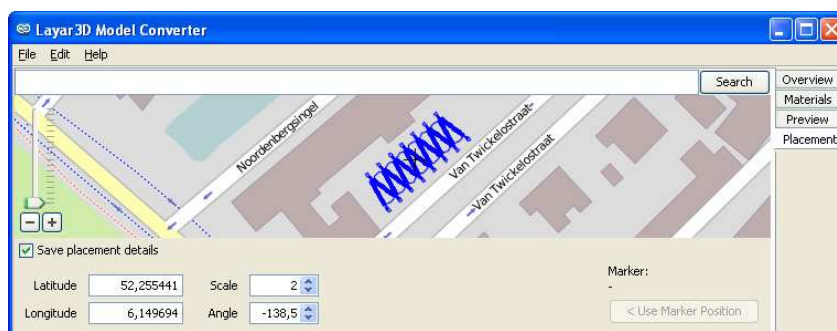
Onder dit tabblad kan het 3D-model bekeken worden. De weergave in dit scherm is gelijk aan hoe het model zich als augment in de Layarapplicatie zal tonen.



Figuur B3.9

### B3.3.4 Placement

Onder het tabblad placement kunnen de plaatsingsparameters (*breedtegraad, lengtegraad, rotatie en schaal*) voor in de SQL database worden bepaald. Schrijf deze waarden op, ze worden niet met het L3D-model opgeslagen.



Figuur B3.10

## B3.4 Registratie en Publicatie

De volgorde van de onderstaande sub-paragrafen in “Hoofdstuk 4 Registratie en Publicatie” is géén verplichte volgorde. Met elke stap kan begonnen of geëindigd worden. Wanneer alle stappen gereed zijn, zal het augment operationeel zijn via de Layarapplicatie op een smartphone/tablet-computer.

### B3.4.1 Maken en vullen van de SQL-database

Om augments te serveren vanaf een eigen server dient een database, server-script (*API endpoint URL*) en een verificatie-script (*config.inc.php*) gemaakt te worden.

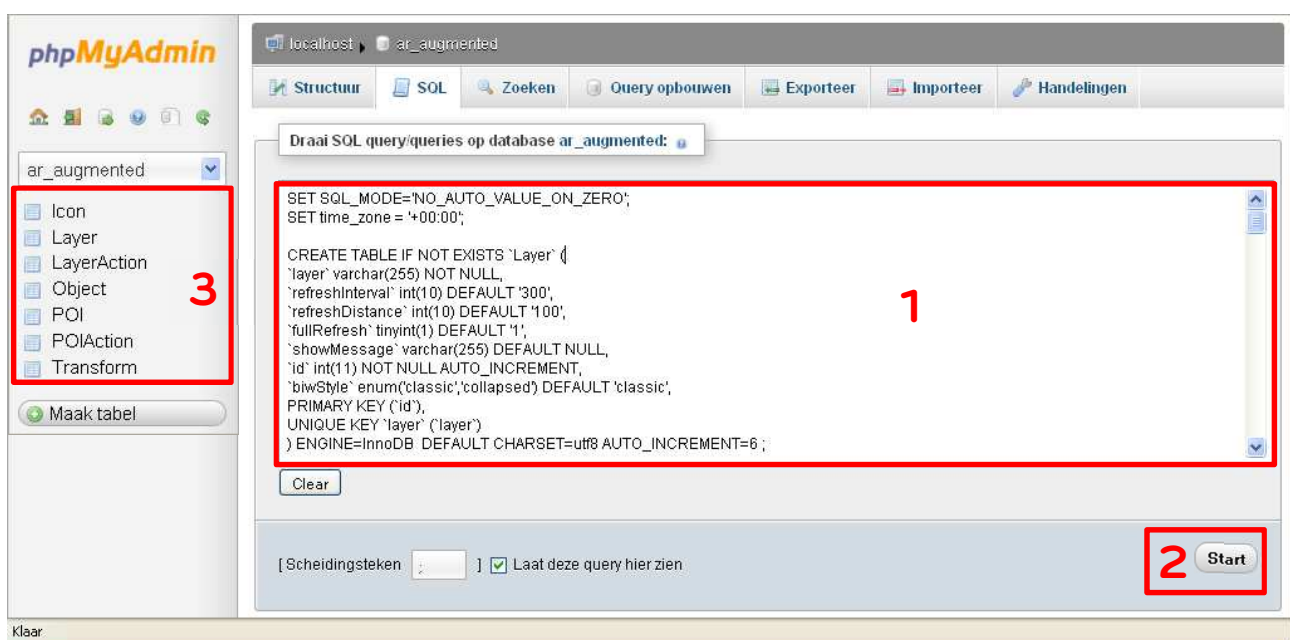
Met behulp van de Excelsheet “AR-Project invulsheet” kan het SQL-script dat de database gaat vormen én vullen, samen met het serverscript en verificatiescript snel gerealiseerd worden.

In dit bestand staat precies beschreven welke attributen noodzakelijk ingevuld moeten worden om een werkende SQL-database te genereren.

Wanneer de tabellen in de “AR-Project invulsheet” zijn ingevuld, dan kan de gevormde SQL-code gekopieerd worden naar de databaseservice (*phpMyAdmin, MySQL*) van de server, zie B3.11 [1]. Na het uitvoeren [2] van deze code wordt de tabellenstructuur inclusief eventuele inhoud aangemaakt [3] binnen de database.

LET OP: Wanneer men een bestaande database gaat uitbreiden met meer regels in de POI-tabel, dienen de verwijzingen (*ID's*) van de POI-tabel naar de onderliggende tabel te kloppen. Wanneer de verwijzing niet mogelijk is doordat het sleutelattribuut in één van de tabellen niet bestaat, wordt het gehele SQL-importeurscript genegeerd.

Het is ten sterkste aan te raden de SQL-database te exporteren naar een SQL-script alvorens nieuwe data geïmporteerd wordt.



Figuur B3.11

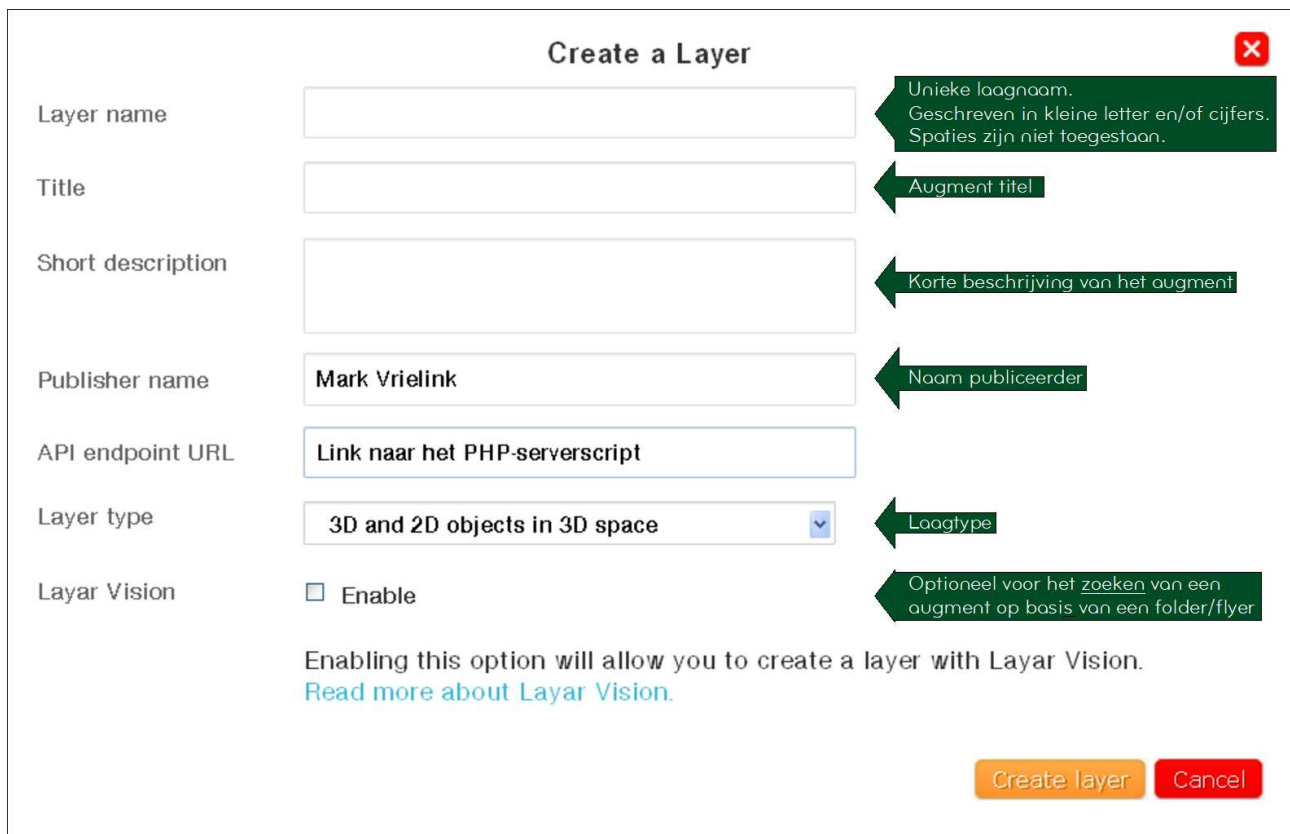


### B3.4.2 Uploaden naar server

Log in op de server met een FTP-programma (*bijvoorbeeld FileZilla*). Bestanden die voor een nieuw project tenminste geüpload dienen te worden, betreffen het serverscript, verificatiescript en de L3D-bestanden. Wanneer men de augment in de Layarapplicatie wil verfraaien door muziek, video of afbeeldingen toe te voegen, dan moeten deze bestanden ook geüpload worden.

### B3.4.3 Registratie bij Layar

Wanneer een account op de website van Layar ([www.layar.com](http://www.layar.com)) is aangemaakt, kan men een augment realitygroep (*layer*) registreren. De registratie bij Layar betreft slechts de koppeling van het eigen PHP-serverscript (*API endpoint URL*) aan Layar en het invullen van de in Layar weer te geven naam (*titel*). Het PHP-script zelf is al gekoppeld aan de SQL-database.



The screenshot shows the 'Create a Layer' form in the Layar application. The form includes the following fields and options:

- Layer name:** A text input field. Annotation: 'Unieke laagnaam. Geschreven in kleine letter en/of cijfers. Spaties zijn niet toegestaan.'
- Title:** A text input field. Annotation: 'Augment titel'
- Short description:** A text input field. Annotation: 'Korte beschrijving van het augment'
- Publisher name:** A text input field containing 'Mark Vrielink'. Annotation: 'Naam publiceerder'
- API endpoint URL:** A text input field containing 'Link naar het PHP-serverscript'.
- Layer type:** A dropdown menu with '3D and 2D objects in 3D space' selected. Annotation: 'Laagtype'
- Layar Vision:** A checkbox labeled 'Enable'. Annotation: 'Optioneel voor het zoeken van een augment op basis van een folder/flyer'

Below the 'Layar Vision' checkbox, there is a note: 'Enabling this option will allow you to create a layer with Layar Vision. [Read more about Layar Vision.](#)'

At the bottom right, there are two buttons: 'Create layer' (orange) and 'Cancel' (red).

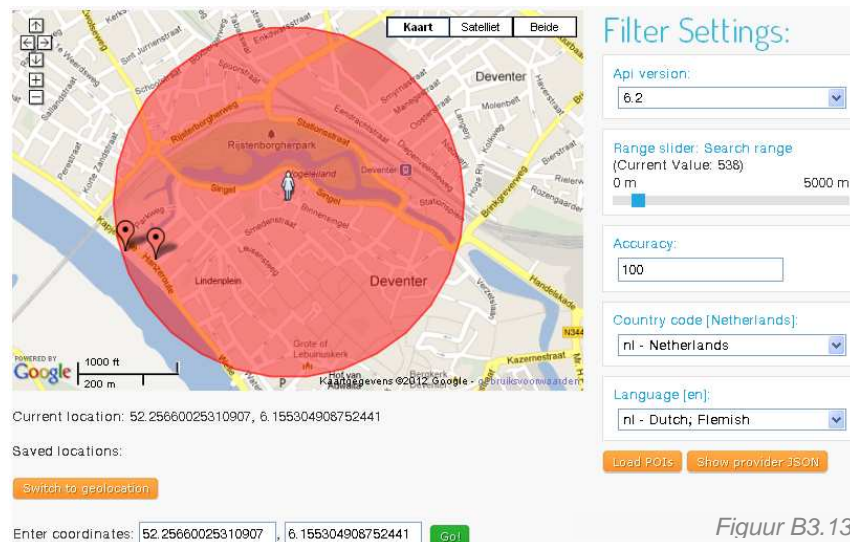
Figuur B3.12

Wanneer deze koppeling gemaakt is, behoudt men de mogelijkheid records in de SQL database aan te maken dan wel te bewerken.



### B3.4.4 Test en publicatie

Wanneer alle voorgenoemde onderdelen doorlopen zijn, zou de augment-server moeten werken en benaderd kunnen worden met een smartphone / tablet-computer in de testmodus van de Layarapplicatie. Mocht het om één of andere reden niet werken, dan kan men na het inloggen op [layar.com](http://layar.com) met de testconsole (zie *figuur B3.13*) achterhalen wat de oorzaak is. De mogelijke foutcodes die gegeven kunnen worden, staan beschreven in hoofdstuk 7 § 7.3.11.



Figuur B3.13

Wanneer men de augment openbaar wil publiceren moet een online-aanvraag gedaan worden bij Layar, die de augments gaat beoordelen op netheid. De overbruggingsperiode vanaf de aanvraag tot publieke publicatie duurt (*mits goedgekeurd*) circa 1 week.

### B3.4.5 Publicatiekosten

- 3D-augments zonder referentie-afbeeldingen *Gratis*
- 3D-augments met referentie-afbeeldingen (*tot 1000 bevestigingen gratis*) daarna € 0,01 / bevestiging. Of zonder limiet voor € 1.000,-. De betaalvariant moet vooraf gekozen worden, het is altijd per direct mogelijk de status van een augment op non-actief te stellen. De augment kan dan niet meer worden bevestigd, waardoor er tevens géén kosten gemaakt worden.

Betalen voor een augment met referentieafbeeldingen wordt afgeraden, aangezien een investering van € 1.000,- zich pas terugverdient nadat er 100.000 bevestigingen zijn gedaan op hetzelfde augment. Dit is een onwaarschijnlijk hoog aantal, zeker doordat het augment locatiegebonden is.