

Projektarbeit

Vorgelegt an der Hochschule für angewandte Wissenschaften Würzburg-Schweinfurt
an der Fakultät Kunststofftechnik und Vermessung

Thema:

Online-Präsentationsform für Punktwolken mit Thermografieinformation

| | |
|-----------------|-------------------------------------|
| Angefertigt in: | Fachhochschule Würzburg-Schweinfurt |
| Betreuer: | Dr. rer. nat. Sebastian Fiedler |
| Prüfer: | Prof. Dr.-Ing. Stefan Knoblach |
| Abgabetermin: | 22. August 2018 |

Eingereicht von:
David Kernstock
aus Deffersdorf, Wieseth
Matrikelnummer: 6015021

Würzburg, den 16. August 2018

ERKLÄRUNG ZUR PROJEKTARBEIT

Hiermit versichere ich, dass die vorgelegte Projektarbeit selbstständig verfasst und noch nicht anderweitig zu Prüfungszwecken vorgelegt wurde.

Alle verwendeten Quellen und Hilfsmittel sind angegeben.

Wörtliche und sinngemäße Zitate wurden als solche gekennzeichnet.

Würzburg, den _____

Unterschrift des/der Studierenden

Inhaltsverzeichnis

| | | |
|----------|--|------------|
| 1 | EINLEITUNG UND MOTIVATION | 1 |
| 2 | POTREE, EINE ONLINEPRÄSENTATIONSFORM FÜR PUNKTWOLKEN | 2 |
| 2.1 | ZWECK VON POTREE | 2 |
| 2.2 | FUNKTIONSWEISE VON POTREE..... | 3 |
| 3 | DARSTELLUNG DER THERMOGRAFIEINFORMATIONEN EINER PUNKTWOLKE..... | 3 |
| 3.1 | BELEGUNG DER KANÄLE EINER LAS-DATEI | 3 |
| 3.2 | UMWANDLUNG DER PUNKTWOLKE VON ASCII IN LAS MIT CLOUDCOMPARE..... | 6 |
| 3.3 | EINLADEN IN POTREE..... | 6 |
| 3.4 | FARBIGE DARSTELLUNG DER THERMOGRAFIE | 8 |
| 4 | ANPASSUNG DER BEDIENUNG | 10 |
| 4.1 | STARTSICHT VOM SCANNERSTANDPUNKT | 10 |
| 4.2 | PROGRAMMIERUNG VON UMSCHALTBUTTONS UND VEREINFACHUNG DER SIDEBAR | 11 |
| 4.3 | ZOOMFUNKTION DURCH VERÄNDERTEN NAVIGATIONSMODUS | 17 |
| 5 | ZUSAMMENFASSUNG UND AUSBLICK..... | 19 |
| | LITERATUR- UND QUELLENVERZEICHNIS | I |
| | ABBILDUNGSVERZEICHNIS..... | II |
| | ANLAGEN..... | III |

1 Einleitung und Motivation

Um dem stetig steigenden Bedarf an Energie in Zeiten des Klimawandels gerecht zu werden, wurde 2014 von den Staats- und Regierungschefs der EU der „Rahmen für die Klima- und Energiepolitik bis 2030“ angenommen. Laut diesem Rahmen verfolgt die europäische Kommission drei Hauptziele, bis 2030 die Treibhausemissionen, im Vergleich zu 1990, um mindestens 40% zu senken, den Anteil erneuerbarer Energien auf mindestens 27% zu erhöhen und die Energieeffizienz um mindestens 27% zu steigern (vgl. ec.europa.eu, 2018).

Um solche Ziele zu erreichen, bietet die Isolierung von Industrieanlagen großes Potential. Die Energieeffizienz wird gesteigert, die CO₂-Emissionen werden gesenkt und außerdem werden Kosten eingespart. So dass sich solche Investitionen meist nach ein bis zwei Jahren bereits auszahlen (vgl. eif.org, 2018). Das Projekt eDlan (effiziente Dämmung von Industrieanlagen) der FHWS versucht mit Wärmebildaufnahmen und terrestrischen Laserscans die Detektion von Schwachstellen zu verbessern, um eine gezieltere und bessere Dämmung zu ermöglichen.

Im Rahmen dieser Projektarbeit soll ein Weg gefunden werden, 3D-Laserscans mit Thermografieinformation in einer geeigneten Form darzustellen.

Die Datensätze sollen hierbei mit der Onlinepräsentationsform Potree verarbeitet werden. Potree bietet die Möglichkeit, Punktwolken mit Klienten, Kunden oder Auftraggebern zu teilen, so dass diese dann die Punktwolken ohne weitere Fachkenntnisse betrachten können. Die Punktwolke wird in einem Webbrowser, wie z.B. Chrome, Firefox etc., dargestellt, so dass keine zusätzliche Software, wie beispielsweise ein Punktwolkenviewer, heruntergeladen und installiert werden muss.

Da Potree für die Darstellung von Punktwolken aus LIDAR-Befliegungen (light detection and ranging) und photogrammetrischen Aufnahmen entwickelt wurde, soll hier eine Anpassung des Programms auf die Darstellung terrestrischer Laserscans mit Thermografieinformation stattfinden. Ziel ist es hierbei, eine Punktwolke mit RGB-Werten und Thermografieinformation in Potree einzuladen und beide Darstellungen, RGB und eine farbige Darstellung der Thermografie zu realisieren.

Des Weiteren soll die Bedienoberfläche den Anforderungen entsprechend und möglichst anwenderfreundlich angepasst und vereinfacht werden.

2 Potree, eine Onlinepräsentationsform für Punktwolken

2.1 Zweck von Potree

Potree ist ein kostenfreier, internetbasierter Open-Source-Viewer zum Darstellen von sehr großen Punktwolken. Es erlaubt dem Nutzer Datensätze mit Milliarden von Punkten aus LIDAR-Messungen oder photogrammetrischen Aufnahmen mit Standardwebbrowsern darzustellen (vgl. Schütz, 2016).

Potree entstand durch eine Masterthesis von Markus Schütz als Teil des Harvest4D-Projekts der TU Wien. Der Octree-Algorithmus, welcher von Potree verwendet wird, wurde ebenfalls an der TU Wien von Michael Wimmer und Claus Schreiblauer im Rahmen des Scanopy Projekts entwickelt (vgl. github, 2018).

Das Rendern großer Punktwolken, das heißt die Darstellung, ohne große Datenmengen vorab laden zu müssen, wird in Potree erreicht, indem die Originaldaten durch eine hierarchische Struktur in verschiedenen Auflösungen gespeichert werden (vgl. Schütz, 2016).

Potree ist ein WebGL(Web Graphics Library)-basierter Renderer, es ist auf einer WebGL 3D rendering library aufgebaut, welche „Three.js“ heißt (vgl. github, 2018).

„WebGL ist eine JavaScript-Programmierschnittstelle, mit deren Hilfe 3D-Grafiken hardwarebeschleunigt im Webbrowser ohne zusätzliche Erweiterungen dargestellt werden können.“ (Wikipedia, 2018)

Potree wurde dazu entwickelt, große Datensätze, welche bei Befliegungen anfallen, darzustellen. Der Vorteil die Darstellung durch Webbrowser zu realisieren liegt daran, dass Punktwolken auf einfache Weise mit Anwendern und der Öffentlichkeit geteilt werden können. So dass der Kunde die Punktwolke anschauen und einige einfache Messungen in der Punktwolke tätigen kann (vgl. Schütz, 2016). Potree bietet hierzu kostenlos die Möglichkeit, ohne dass der Anwender zusätzliche spezielle Software installieren muss. Potree ist außerdem sehr anwenderfreundlich, weil es die Möglichkeit bietet, auf Anwendungen zugeschnitten und angepasst zu werden.

2.2 Funktionsweise von Potree

Mit Potree können Punktwolken, welche im LAS-Format abgespeichert wurden, in einem Webbrowser dargestellt werden.

Damit eine Punktwolke im Potree-Viewer in einem Browser betrachtet werden kann, muss zuerst eine Anwendung ausgeführt werden, um ein Potreeprojekt zu erstellen. Ein solches Projekt besteht aus einer eigenständigen HTML-Seite und zwei Ordnern. Um die Punktwolke im Viewer darzustellen, kann nun die HTML-Seite in einem Browser, z.B. Firefox, geöffnet werden. In dem ersten Ordner „pointcloud“ liegen die eigentlichen Daten der Punktwolke, aufgeteilt in mehrere Unterordner, in verschiedenen Auflösungen vor. Dies ist notwendig, damit Potree die großen Datenmengen der Punktwolke möglichst schnell verarbeiten und jeweils passend anzeigen kann. Im zweiten Ordner „libs“ wird eine Bibliothek aus mehreren HTML-, CSS- und JavaScript-Dateien, zum Teil aus anderen OpenSource-Programmen, angelegt. Beim Aufrufen der HTML-Seite werden die Dateien des Ordners „libs“ im Hintergrund aufgerufen und ausgeführt. Unter anderem befinden sich hier im Unterordner „potree“ auch die Dateien „sidebar.html“ und „potree.js“, welche im Rahmen dieser Projektarbeit, neben der Haupt-HTML-Seite, bearbeitet wurden.

3 Darstellung der Thermografieinformationen einer Punktwolke

3.1 Belegung der Kanäle einer LAS-Datei

Um eine Punktwolke in Potree darzustellen, muss die Punktwolke in einem LAS-Format, also mit der Dateierweiterung „.las“ oder als komprimierte Variante „.laz“ vorliegen. Die Punktwolke muss also zuerst in das LAS-Format konvertiert und abgespeichert werden.

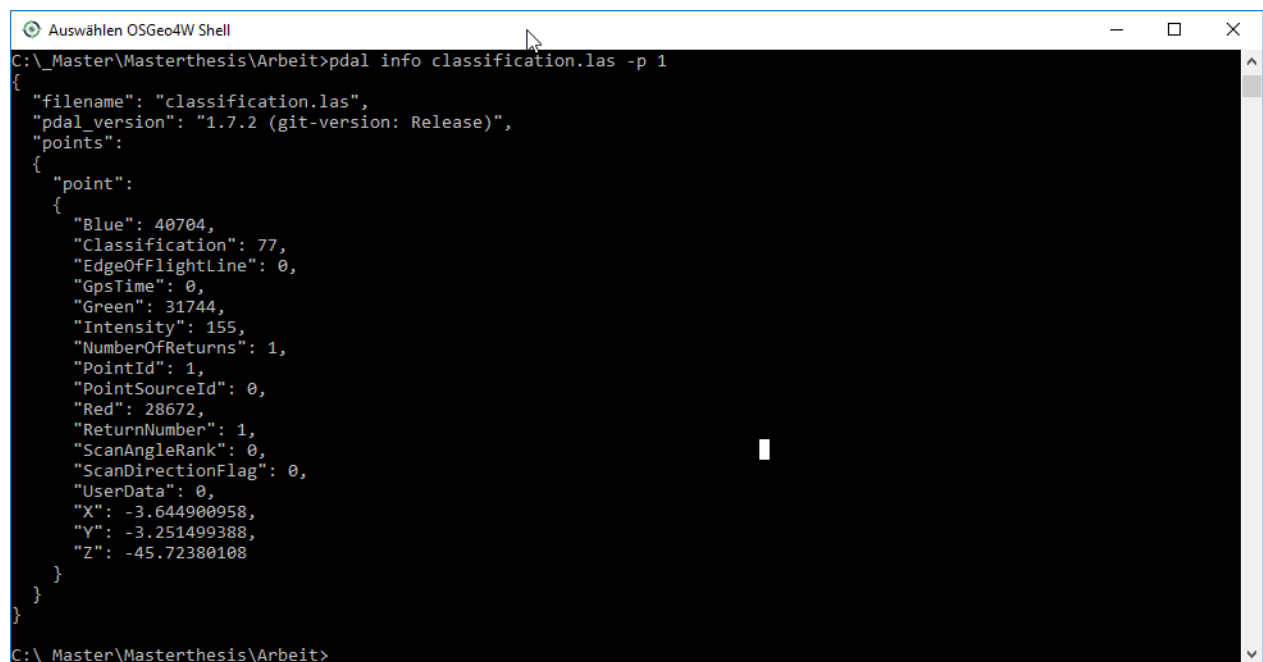
Um zu gewährleisten, dass später sowohl die Thermografie als auch die RGB-Werte der Punkte dargestellt werden können, ist darauf zu achten, dass die Kanäle, welche mit Werten belegt werden, später auch in Potree einlesbar sind.

Das LAS-Format wurde als Austauschdatenformat für LIDAR-Befliegungsdaten entwickelt. Im Gegensatz dazu liegen Punktwolken aus terrestrischem Laserscanning nicht im LAS-Format vor, sondern oft in einem herstellereigenen Format oder als ASCII-

Datei. Deshalb müssen diese Formate zuerst in eine Datei im LAS-Format umgewandelt werden. Hierzu eignet sich CloudCompare, eine Open-Source-Software zum Bearbeiten von Punktwolken, sehr gut. In CloudCompare gibt es zwei verschiedene Möglichkeiten, Punktwolken in LAS-Formaten zu exportieren. Dateien im Format „LAS 1.3 or 1.4“ sind nicht in Potree einlesbar. Deshalb wird das andere Format „LAS cloud“, welches für das Einlesen in Potree geeignet ist, verwendet.

Das LAS-Format wurde von der ASPRS (American Society for Photogrammetry and Remote Sensing) so entwickelt, dass es mehrere verschiedene Punktdatenformate gibt, das heißt verschiedene Formattypen mit verschiedenen Kanälen. Um einen geeigneten Kanal für die Belegung mit der Thermografieinformation zu finden, muss nun das Punktdatenformat bestimmt werden.

Mit der Software „OSGeo4W“ des PDAL-Projekts (Point Data Abstraction Library), kann man LAS-Dateien lesbar machen und nachschauen, welche Kanäle belegt wurden (siehe Abb. 1), (vgl. pdal.io, 2018). Hierdurch wurde bestimmt, dass das „Punktdatenformat 3“ (siehe Abb. 2) verwendet wird.



```
Auswählen OSGeo4W Shell
C:\_Master\Masterthesis\Arbeit>pdal info classification.las -p 1
{
  "filename": "classification.las",
  "pdal_version": "1.7.2 (git-version: Release)",
  "points": {
    "point": {
      "Blue": 40704,
      "Classification": 77,
      "EdgeOfFlightLine": 0,
      "GpsTime": 0,
      "Green": 31744,
      "Intensity": 155,
      "NumberOfReturns": 1,
      "PointId": 1,
      "PointSourceId": 0,
      "Red": 28672,
      "ReturnNumber": 1,
      "ScanAngleRank": 0,
      "ScanDirectionFlag": 0,
      "UserData": 0,
      "X": -3.644900958,
      "Y": -3.251499388,
      "Z": -45.72380108
    }
  }
}
```

Abbildung 1: Bestimmung des Punktdatenformats der LAS-Datei, eigene Darstellung, 2018

Table 12: Point Data Record Format 3

| Item | Format | Size | Required |
|--|-----------------------|---------|----------|
| X | long | 4 bytes | * |
| Y | long | 4 bytes | * |
| Z | long | 4 bytes | * |
| Intensity | unsigned short | 2 bytes | |
| Return Number | 3 bits (bits 0, 1, 2) | 3 bits | * |
| Number of Returns (given pulse) | 3 bits (bits 3, 4, 5) | 3 bits | * |
| Scan Direction Flag | 1 bit (bit 6) | 1 bit | * |
| Edge of Flight Line | 1 bit (bit 7) | 1 bit | * |
| Classification | unsigned char | 1 byte | * |
| Scan Angle Rank (-90 to +90) – Left side | char | 1 byte | * |
| User Data | unsigned char | 1 byte | |
| Point Source ID | unsigned short | 2 bytes | * |
| GPS Time | double | 8 bytes | * |
| Red | unsigned short | 2 bytes | * |
| Green | unsigned short | 2 bytes | * |
| Blue | unsigned short | 2 bytes | * |

Abbildung 2: Kanäle des LAS-Formats, LAS Specification, 2013

Ziel ist es, die Thermografie zusätzlich zu den Koordinaten, den RGB-Werten und der Intensität in Potree einzulesen, um eine Darstellung der Punktwolke in drei Varianten realisieren zu können. Die erste Überlegung war, die Thermografiewerte genauso wie die RGB-Werte auf drei Kanäle verteilt einzulesen. Aufgrund des begrenzten LAS-Formats (keine drei gleichen Kanäle, siehe Abb. 2), wurde sich jedoch dafür entschieden, es auf einen einzelnen Kanal zu beschränken.

Die Thermografie soll, ähnlich wie die Intensität, also als einzelner Kanal eingelesen werden. Als freier Kanal, mit gleicher Speicherkapazität wie die Intensität, steht hier nur noch die „Point Source ID“ (ebenfalls 2 Byte Speicherkapazität) zur Verfügung (siehe Abb.2). Es gibt außerdem noch weitere Kanäle mit genügend Speicherkapazität für Werte von 0 bis 255 (1 Byte): User Data, Classification, Scan Angle Rank, GPS Time (8 Byte).

Aktuell ist es in Potree möglich, neben den Koordinaten die RGB-Werte, die Intensität und Klassifikationswerte einlesen. Deshalb wurde der Kanal „Classification“ mit der Thermografie belegt (siehe Abb. 1). Da in Potree bereits verschiedene Klassifikationstypen vordefiniert sind, funktionierte es jedoch nicht, die Thermografie als Classification einzulesen.

Deshalb wurde entschieden, im Rahmen dieser Projektarbeit auf die Darstellung der Intensitätswerte zu verzichten und die Thermografie stattdessen als „Intensity“ einzulesen.

Um die Thermografiewerte zusätzlich zur Intensität einlesen zu können, kann versucht werden, Potree umzuprogrammieren, sodass beispielsweise zusätzlich der Kanal „Point Source ID“ oder die „Classification“ in anderer Form eingelesen werden können.

3.2 Umwandlung der Punktwolke von ASCII in LAS mit CloudCompare

Da in den meisten Softwarepaketen eine Punktwolke als ASCII-Datei exportiert werden kann, wird eine solche Datei als Ausgangsformat vorausgesetzt. Die Umwandlung von einer ASCII-Datei in das LAS-Format erfolgte in CloudCompare. Die ASCII-Datei sollte folgende Spalten enthalten:

- X-,Y-,Z-Koordinaten der Punktwolke als float (während dieser Projektarbeit mit je vier Nachkommastellen)
- drei Spalten Farbinformation als RGB (Red, Green, Blue) als integer von 0 bis 255
- eine Spalte Thermografieinformation ebenfalls als integer von 0 bis 255

Beim Einlesen in CloudCompare ist zu beachten, dass die ersten sechs Spalten beim Einlesen die Bezeichnung X, Y, Z und R, G, B haben. Die Spalte Thermografie ist als ScalarField einzulesen. Da die Thermografie in Potree später als Intensität eingelesen werden soll, muss das ScalarField jetzt in CloudCompare noch in „Intensity“ umbenannt werden.

Anschließend wird die Punktwolke als LAS- oder LAZ-Datei gespeichert.

Hier ist zu beachten, dass beim Exportieren das Format „LAS cloud“ gewählt wird (siehe 3.1). Die gespeicherte Datei kann dann von Potree weiterverarbeitet werden.

3.3 Einladen in Potree

Um nun die LAS- oder LAZ-Datei in Potree einladen zu können, um somit eine HTML-Datei zu erstellen, welche man später im Webbrowser öffnen kann, muss der Potree-Converter vorhanden sein.

Von der Website „potree.org“ muss im Downloadbereich ein Ordner „Potree Converter“ heruntergeladen und abgespeichert werden. Es ist darauf zu achten, dass eine .exe-

Datei mit dem Namen „PotreeConverter.exe“ vorhanden ist. Diese Anwendung wird später ausgeführt, um die LAS-Punktwolke umzuwandeln.

Die Anwendung wird nun durch eine Eingabe in die MS-DOS-Kommandozeile aufgerufen. Um die Punktwolke einzulesen, muss der Pfad der exe-Datei und der LAS-Datei angegeben werden. Außerdem müssen die Attribute, welche eingelesen werden sollen und der Name der HTML-Datei, die erstellt werden soll, genannt werden (siehe Abb. 2).

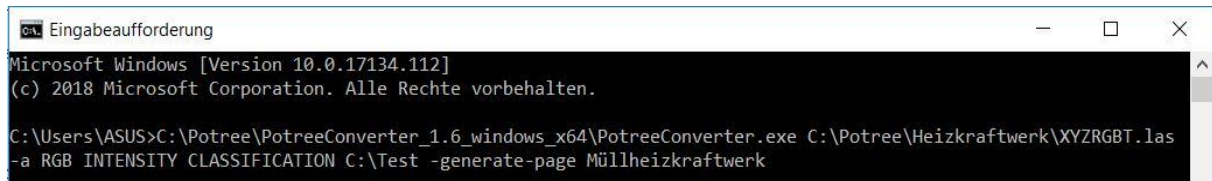


Abbildung 3: Kommandozeilenbefehl zur Erstellung eines Potreeprojekts, eigene Darstellung, 2018

Nun wird die Anwendung aufgerufen und der Prozess des Umschreibens der LAS-Datei in verschiedene Auflösungen in mehrere Unterordner, sowie die Erstellung der HTML-Datei und der Bibliothek gestartet. Um den Speicherplatz des Potree-Projekts zu verringern, gibt es die Möglichkeit, die verschiedenen Auflösungen der Punktwolke, welche später im Unterordner „pointcloud“ zu finden sind, anstatt von BIN-Dateien als LAZ-Dateien abzuspeichern. Hierzu muss bei der Eingabe in die Kommandozeile „--output format LAZ“ angehängt werden (siehe Abb. 3). Durch diese Möglichkeit lässt sich die Größe des Projekts erheblich verringern (z.B. von 740 auf 184 MB). Um jedoch eine ähnliche Performance beider Versionen zu erreichen, ist bei der komprimierten Version eine gewisse Rechnerleistung erforderlich.

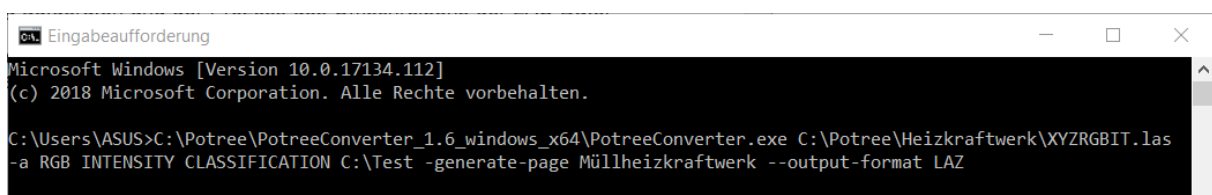


Abbildung 4: Kommandozeilenbefehl zur Erstellung eines Potreeprojekts (LAZ), eigene Darstellung, 2018

Nach Abschluss des Prozesses steht am Speicherort der LAS-Datei ein Ordner zur Verfügung, in welchem neben den beiden Unterordnern „libs“ und „pointcloud“ auch die Haupt-HTML-Datei vorhanden ist. Über diese Datei kann nun der Viewer in einem Standardwebbrowser aufgerufen werden.

3.4 Farbige Darstellung der Thermografie

Da die Thermografieinformation (bedingt durch das LAS-Format) nur auf einem Kanal eingelesen werden kann, wird nun eine andere Möglichkeit gesucht, die Thermografie farbig darzustellen.

Der einzelne Kanal hat für die Punkte 8-bit-Werte (0 bis 255), welche bei einfachem Einlesen als schwarz-weiße Punktwolke angezeigt werden. Um die Thermografie in RGB-Werten (Red, Green, Blue), also drei Kanälen jeweils mit Werten von 0 bis 255, darzustellen, müssen jedem „schwarz-weiß“-Wert drei Werte (Red, Green, Blue) zugeordnet werden.

Potree bietet die Möglichkeit, mittels „materialGradient“ einzelnen Werten RGB-Werte zuzuordnen. Die Werte sind statt 0 bis 255 nun 0 bis 1 (float). Es wurde nun eine Matrix gebildet, die 256 Wertstufen von 0 bis 1 je drei andere Werte (float) zuordnet.

Beim ersten Versuch wurden drei Zeilen (0 = blau, 128 = grün, 255 = rot) festgehalten und der Rest gleichmäßig aufgefüllt (siehe Abb. 4).

| Alter Wert | Red | Green | Blue |
|------------|-----|-------|------|
| 0/255 | 0 | 0 | 1 |
| ... | 0 | ... | ... |
| 128/255 | 0 | 1 | 0 |
| ... | ... | ... | 0 |
| 255/255 | 1 | 0 | 0 |

Abbildung 5: Gradientenmatrix (vorläufig), eigene Darstellung, 2018

Da bei Verwendung dieser Matrix die Farbverteilung stark rot- und grün-lastig war, wurde eine zweite Matrix berechnet. Hierbei wurden fünf Zeilen (blau, türkis, grün, gelb, rot) festgehalten und wieder gleichmäßig aufgefüllt (siehe Abb. 5). Die Farbverteilung hier wirkt mehr differenziert (siehe Abb. 6 & 7).

| Alter Wert | Red | Green | Blue |
|------------|-----|-------|------|
| 0/255 | 0 | 0 | 1 |
| ... | 0 | ... | 1 |
| 64/255 | 0 | 1 | 1 |
| ... | 0 | 1 | ... |
| 128/255 | 0 | 1 | 0 |
| ... | ... | 1 | 0 |
| 192/255 | 1 | 1 | 0 |
| ... | 1 | ... | 0 |
| 255/255 | 1 | 0 | 0 |

Abbildung 6: Gradientenmatrix (endgültig), eigene Darstellung, 2018

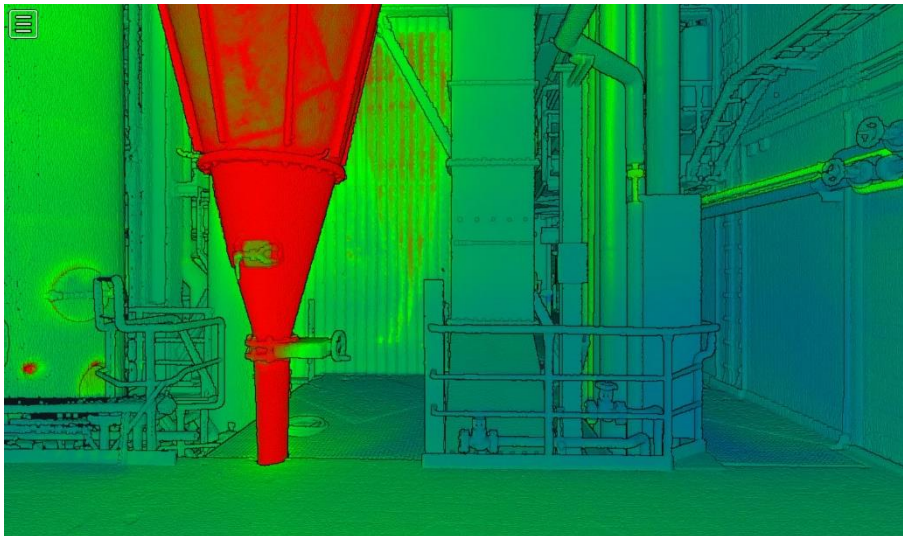


Abbildung 7: Thermografieansicht mit vorläufiger Gradientenmatrix, eigene Darstellung, 2018

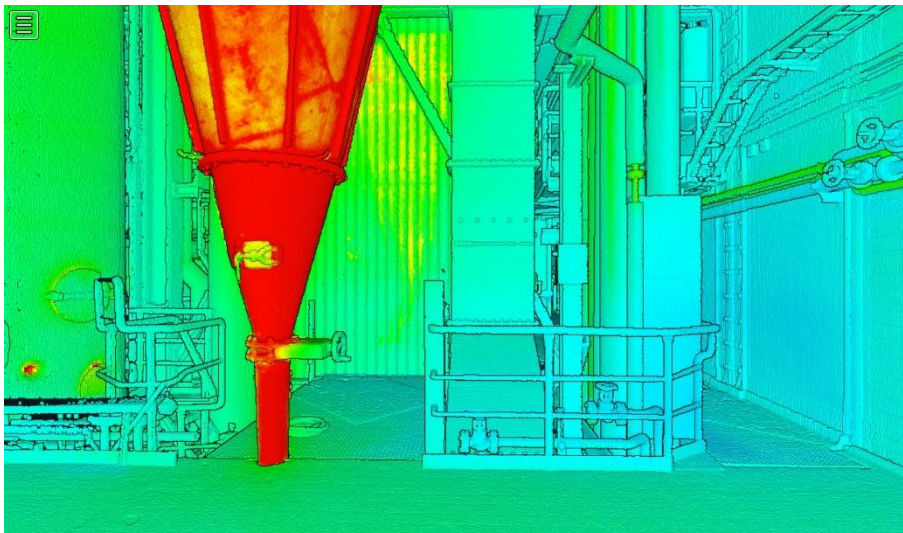


Abbildung 8: Thermografieansicht mit endgültiger Gradientenmatrix, eigene Darstellung, 2018

Die Matrix wurde nun als neue Vorlagen für den „materialGradient“ unter dem Namen „THERMO“ in der JavaScript-Datei „potree.js“ im Ordner „./libs/potree“ angelegt und als Standardwert aktiviert (siehe Abb. 9). So kann nun zwischen RGB-Ansicht und farbig-er Thermografieansicht gewechselt werden.

```
Potree.Gradients = {  
  
  THERMO: [  
    [0/255, new THREE.Color(0.000,0.000,1.000)],  
    [1/255, new THREE.Color(0.000,0.016,1.000)],  
    [2/255, new THREE.Color(0.000,0.031,1.000)],  
    [3/255, new THREE.Color(0.000,0.047,1.000)],  
    [4/255, new THREE.Color(0.000,0.063,1.000)],  
    ...USW.  
  ]  
}
```

Abbildung 9: Gradientenmatrix im Quellcode (potree.js), eigene Darstellung, 2018

4 Anpassung der Bedienung

4.1 Startansicht vom Scannerstandpunkt

Da Potree vor allem für Punktwolken aus Befliegungsdaten gemacht wurde, startet die Ansicht der Punktwolke als Übersicht von außen, so wie es bei Befliegungsdaten auch sinnvoll ist. Für die Ansicht einer Punktwolke aus terrestrischem Laserscanning ist es jedoch besser, die aufgenommenen Objekte aus Sicht des Laserscanners anzuschauen. Ziel ist es also, eine Sicht vom Scannerstandpunkt aus zu realisieren.

Hierzu gibt es in Potree zwei Funktionen, die man in den HTML-Quelltext der Hauptseite einfügen muss:

„viewer.scene.view.position.set()“ & „viewer.scene.view.lookAt()“;

Außerdem muss die Funktion „viewer.fitToScreen()“, welche die Ansicht standardmäßig auf eine Übersichtsperspektive stellt, gelöscht oder auskommentiert werden (siehe Abb. 9).

```
//viewer.fitToScreen();  
viewer.scene.view.position.set(0.000001,-0.000001 -50.8517);  
viewer.scene.view.lookAt(new THREE.Vector3(0, 0, -50.8517));
```

Abbildung 10: Startansicht im Quellcode, eigene Darstellung, 2018

Bei der „lookAt“-Funktion wird die Position des Zentrums festgelegt, um das beim Drehen der Punktwolke rotiert wird. Die „position“-Funktion gibt den eigentlichen Positionspunkt des Sichtzentrums an. Wählt man also bei der „position“-Funktion eine Koordinate aus, die beispielsweise 10m entfernt von der Koordinate der „lookAt“-Funktion liegt, so wird beim Drehen der Ansicht in einem Radius von 10m um die Koordinate der „lookAt“-Funktion gedreht.

Da das Ziel ist, direkt auf der Koordinate des Scannerstandpunktes ohne Radius zu drehen, soll für beide Funktionen dieselbe Koordinate verwendet werden.

Wenn dieselbe Koordinate verwendet wird, ist jedoch keine Punktwolke mehr zusehen. Als praktische Lösung bietet sich an, die Koordinate des Scannerstandpunktes in die „lookAt“-Funktion zu schreiben und die Koordinate der „position“-Funktion minimal anders zu wählen. Hier kann gleichzeitig festgelegt werden, in welche Richtung man bei der Startansicht schaut. Die z-Koordinate, gibt die Höhe an, und kann bei beiden Funktionen gleich gewählt werden.

Im Rahmen der Projektarbeit wurde die Standpunktcoordinate mit CloudCompare ermittelt, indem die GBL-Datei des Laserscans eingeladen und angewandt wurde. Danach konnten die Zentrumskoordinaten angezeigt und kopiert werden. Diese Koordinaten konnten später in Potree eingefügt werden.

Alternativ zur Eingabe im Quelltext der HTML-Hauptseite können beide Funktionen mit den Standpunktkoordinaten auch im Abschnitt, in welchem die Standardwerte definiert werden, in der JavaScript-Datei „potree.js“ hinzugefügt werden.

Wenn man nun Datensätze mit denselben Standpunktkoordinaten hat, muss nur die JavaScript-Datei ersetzt werden, ohne dass ein manuelles Editieren des Quelltextes nötig ist. Beim terrestrischen Laserscanner Faro Focus 3D erhält man gleiche Standpunktkoordinaten, wenn man die Erfassung des barometrischen Luftdrucks deaktiviert. Der Scannerstandpunkt ist dann im Ursprung (Koordinaten: 0, 0, 0).

4.2 Programmierung von Umschaltbuttons und Vereinfachung der Sidebar

Da der Punktwolkenvviewer vor allem von Anwendern, welche keine große Erfahrung im Umgang mit Punktwolken haben, bedient werden soll, gilt es nun, die Bedienung des Viewers durch die Sidebar zu vereinfachen bzw. dem Zweck des Detektierens von Wärmeverlusten entsprechend anzupassen.

Zuerst soll ein Knopf (Button) eingebaut werden, mit welchem man die Ansicht so verändern kann, dass man einerseits eine Übersicht der Punktwolke von außen sieht, aber andererseits wieder zur Sicht vom Scannerstandpunkt aus (siehe 4.1) zurückschalten kann. Die Programmierung des Buttons findet in der „sidebar.html“ im Ordner „./libs/potree“ statt.

Die grobe Funktionsweise wird hierbei von dem bereits vorhandenen Button, mit welchem zwischen Standard und High Quality gewechselt werden kann, kopiert. Genauso wie beim vorhandenen Knopf wird nach einer Überschrift eine neue „selectgroup“ definiert. In dieser Selectgroup wird angegeben, wie viele Teilelemente der Knopf hat, welche Aufschrift auf den verschiedenen Teilelementen steht und welcher Wert für das Betätigen des jeweiligen Elements an die Funktion übergeben werden soll. Der Knopf hat zwei Teilelemente: Scanner und Overview, wobei die Werte „scan“ bzw. „over“ übergeben werden (siehe Abb. 11, 1).

Die Funktion des Buttons war beim vorhandenen Knopf in der JavaScript-Datei „potree.js“ programmiert. Aus Gründen der Übersichtlichkeit wurde die Funktion des neuen Buttons in ein Inline-JavaScript unter der „selctgroup“ in die „sidebar.html“ geschrieben (siehe Abb. 11, 2).

```
<div class="divider"><span>Point of View</span></div>

<li>
  <selectgroup id="view_options">
    <option id="scannerview" value="scan">Scannerview</option>
    <option id="overview" value="over">Overview</option>
  </selectgroup>
</li>
<script>
{
  let elScanView = $("#view_options");
  elScanView.selectgroup({title: ""});

  elScanView.find("input").click( (e) => {
    if(e.target.value === "scan"){
      viewer.scene.view.position.set(0.00000001,-0.00000001, -50.8517);
      viewer.scene.view.lookAt(new THREE.Vector3(0, 0, -50.8517));
    }else if(e.target.value === "over"){
      viewer.fitToScreen();
    }
  });

  let currentView = viewer.scene.view ? "scan" : "over";
  elScanView.find(`input[value=${currentView}]`).trigger("click");
}
</script>
```

Abbildung 11: Ansichtsbutton im Quellcode (sidebar.html), eigene Darstellung, 2018

Die Funktion startet damit, dass aufgrund des Namens „view_options“ die Funktion der „selectgroup“, also dem Button, zugeordnet wird. Als Nächstes wird eine If-Funktion aufgerufen, sobald der Kopf angeklickt worden ist. Die If-Funktion prüft, welches Teilelement des Knopfes angeklickt wurde, also welcher Wert übergeben wurde. Für den Wert „scan“ soll die Sicht zu der des Scannerstandpunktes gewechselt werden, also werden die beiden Funktionen „position“ und „lookAt“ (siehe 4.1) aufgerufen. Für den übergebenen Wert „over“ wird die Funktion „viewer.fitToScreen()“ aufgerufen, welche eine Übersicht der Punktwolke von außen zur Folge hat (siehe Abb. 11, (2.)).

Zuletzt wird mit Hilfe der Variable „currentView“ festgelegt, dass der Knopf mit der aktuellen Sicht als angeklickt angezeigt wird (siehe Abb. 11, 3).

Wie in 4.1 hat man hier dasselbe Problem mit den Standpunktkoordinaten. Man müsste sie für jeden Datensatz neu in der Funktion editieren. Bei jeweils demselben Scannerstandpunkt von (0, 0, 0) würde dieses Problem entfallen.

Als Nächstes soll das Umschalten zum Einfärben der Punktwolke vereinfacht werden. Anstatt die Punktwolke zuerst auswählen zu müssen, um danach im sich öffnenden Drop-Down-Menu einen Kanal auszusuchen, soll wie bei der Ansicht ein Button zum Umschalten zwischen RGB und Thermografie erstellt werden.

Der Knopf ist von der Funktionsweise genauso aufgebaut, wie der, der die Ansicht ändert. Es gibt wieder Teilelemente, welche beim Anklicken je einen Wert an die If-Funktion im Inline-JavaScript, übergeben. Es gibt dieses Mal drei Teilelemente RGB, Thermography und Thermo grey mit den Übergabewerten „rgb“, „ther“, „int“, zum Umschalten zwischen RGB, farbiger Thermografie und Thermografie in grau. Wenn es möglich ist die Intensität miteinzuladen, kann man diese noch hinzufügen oder die Thermografie schwarz-weiß ersetzen (siehe Abb. 12, 1).

Um nun, nach Prüfen des Übergabewerts durch die If-Funktion, den gewünschten Kanal anzuzeigen, muss der Variable „pointcloud.material.pointColorType“ der jeweils richtige „ColorType“ zugeordnet werden. Dies ist bei der farbigen Thermografie der mit dem Gradienten (siehe 3.4) gerenderte Kanal „Intensity“, also „.INTENSITY GRADIENT“, bei der schwarz-weißen Thermografie nur der Kanal „.INTENSITY“ (siehe Abb. 12, 2).

Der Entwickler von Potree Markus Schütz gab den Hinweis, dass vor der Belegung der Variable ein Auswählen der Punktwolke mittels einer for-Schleife nötig sei (siehe Abb. 12, 3).


```
<div class="divider"><span>Channel</span></div>

<li>
  <selectgroup id="display_options">
    <option id="display_options_rgb" value="rgb"> RGB </option> (1.)
    <option id="display_options_thermography" value="ther"> Thermography </option>
    <option id="display_options_intensity" value="int"> Thermo grey </option>
  </selectgroup>
</li>
<script>
{
  let elScanDisplay = $("#display_options");
  elScanDisplay.selectgroup({title: ""});

  elScanDisplay.find("input").click( (e) => {
    if(e.target.value === "rgb"){
      for(let pointcloud of viewer.scene.pointclouds){
        pointcloud.material.pointColorType = Potree.PointColorType.RGB;
        viewer.setEDLEnabled(false);
        $('#chkEDLEnabled')[0].checked = viewer.getEDLEnabled();
      }
    }else if(e.target.value === "ther"){
      for(let pointcloud of viewer.scene.pointclouds){ (3.)
        pointcloud.material.pointColorType = Potree.PointColorType.INTENSITY_GRADIENT; (2.)
        viewer.setEDLEnabled(true);
        $('#chkEDLEnabled')[0].checked = viewer.getEDLEnabled(); (4.)
      }
    }else if(e.target.value === "int"){
      for(let pointcloud of viewer.scene.pointclouds){
        pointcloud.material.pointColorType = Potree.PointColorType.INTENSITY;
        viewer.setEDLEnabled(true);
        $('#chkEDLEnabled')[0].checked = viewer.getEDLEnabled();
      }
    }
  });
  let currentDisplay = viewer ? "rgb" : "int";
  elScanDisplay.find(`input[value=${currentDisplay}]`).trigger("click"); (5.)
}
</script>
```

Abbildung 12: Button zur Kanalauswahl im Quellcode (sidebar.html), eigene Darstellung, 2018

Potree hat die Funktion „Eye-Dome-Lighting“ (EDL); ist diese aktiviert, wird die Anzeige von Schatten realisiert und dadurch Konturen und Kanten angezeigt. Für die Kanäle Thermografie (farbig und grau) und für die optionale Intensität ist diese Anzeige von großem Vorteil, weil man hier nun auch im Thermografiekanal die einzelnen Objekte voneinander unterscheiden kann.

Um das „Eye-Dome-Lighting“ nicht nach jedem Umschalten von RGB auf Thermografie mittels anklicken manuell aktivieren zu müssen, soll die Aktivierung mit in den Button integriert werden.

Das EDL wird mittels „true“ und „false“ beim RGB deaktiviert und bei der Thermografie aktiviert, danach wird jeweils die dazugehörige Checkbox geprüft, damit Anzeige und Aktivierung der Funktion übereinstimmen (siehe Abb. 12, 4).

Wie bei dem Knopf, der die Ansicht steuert, wird zuletzt festgelegt, dass der Knopf zur aktuellen Anzeige als angeklickt angezeigt wird (siehe Abb. 11, 5).

Da es das Ziel ist, später bei unterschiedlichen Datensätzen möglichst wenige Dateien austauschen oder editieren zu müssen, werden die Standardwerte für die EDL (Radius und Strength), die Anzahl der angezeigten Punkte und der Field-of-View im JavaScript „potree.js“ eingetragen. Außerdem wird auch der Scannerstandpunkt hier eingetragen (siehe Abb. 13). Bei Datensätzen mit einheitlichen Standpunktkoordinaten, muss dann die Haupt-HTML-Seite nicht mehr ausgetauscht bzw. editiert werden.

```
{ // set defaults
  this.setFOV(60);
  this.setEDLEnabled(false);
  this.setEDLRadius(1.3);
  this.setEDLStrength(0.5);
  this.setClipTask(Potree.ClipTask.HIGHLIGHT);
  this.setClipMethod(Potree.ClipMethod.INSIDE_ANY);
  this.setPointBudget(5*1000*1000);
  this.setShowBoundingBox(false);
  this.setFreeze(false);
  this.setNavigationMode(Potree.EarthControls);
  this.setBackground('gradient');
  this.scene.view.position.set(0.000001,-0.000001,-50.8517);
  this.scene.view.lookAt(new THREE.Vector3(0, 0, -50.8517));
```

Abbildung 13: Standardwerte (potree.js), eigene Darstellung, 2018

Zum Schluss wird die Bedienleiste durch Editieren in der „sidebar.html“ etwas ausgedünnt, um nicht relevante Funktionen wegzulassen und die Sidebar somit übersichtlicher zu gestalten. So werden die beiden Reiter „Classification filter“ und „Scene“ weggelassen. Klassifikationen sind für das Projekt, thermische Informationen darzustellen, nicht relevant. Durch den Umschaltbutton zwischen RGB-Thermografie ist die wichtigste Funktion des Reiters „Scene“ erhalten geblieben. Die anderen Funktionen in „Scene“ sind für das Projekt ebenso nicht relevant (siehe Abb. 14 und 15).

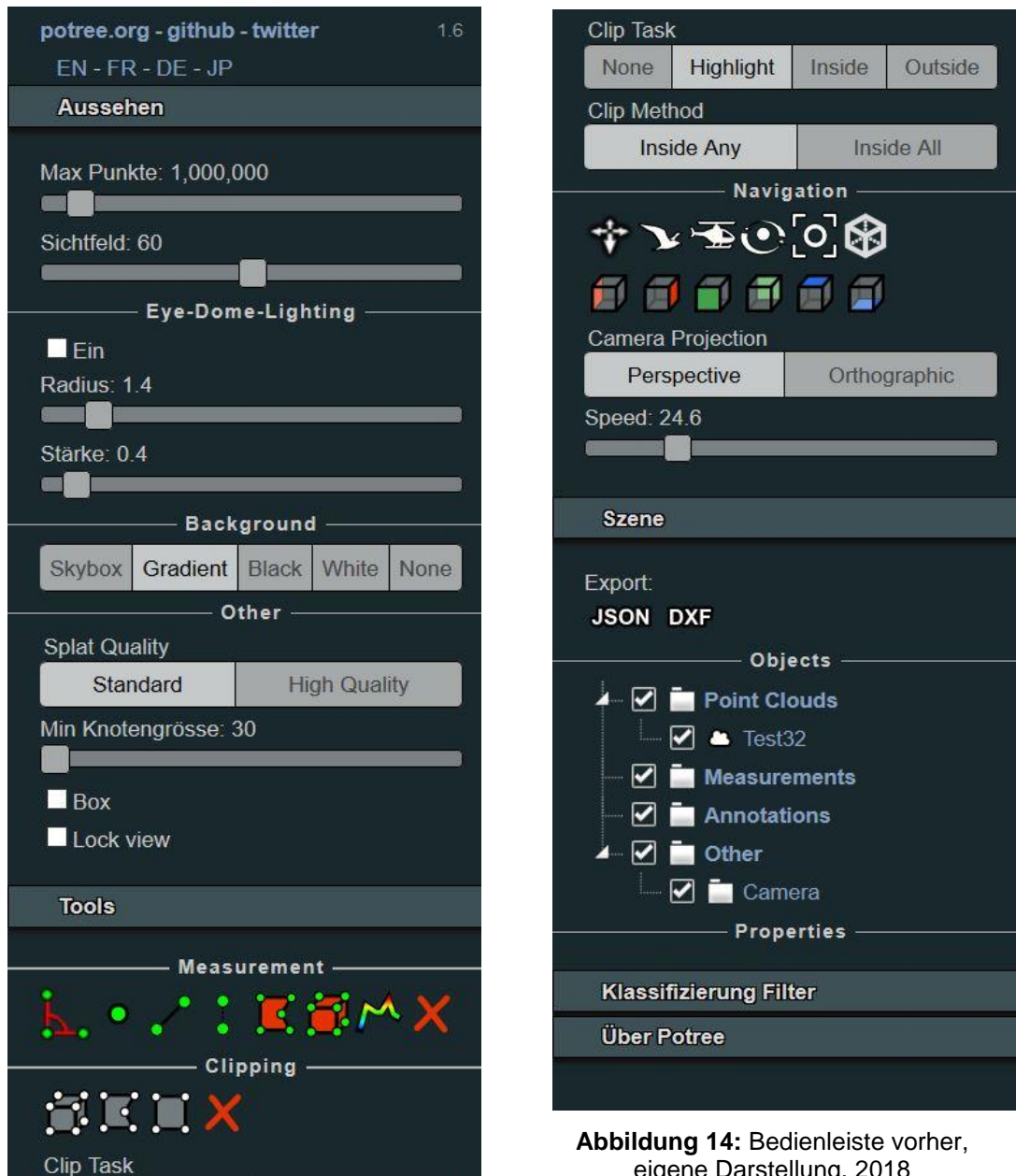


Abbildung 14: Bedienleiste vorher,
eigene Darstellung, 2018

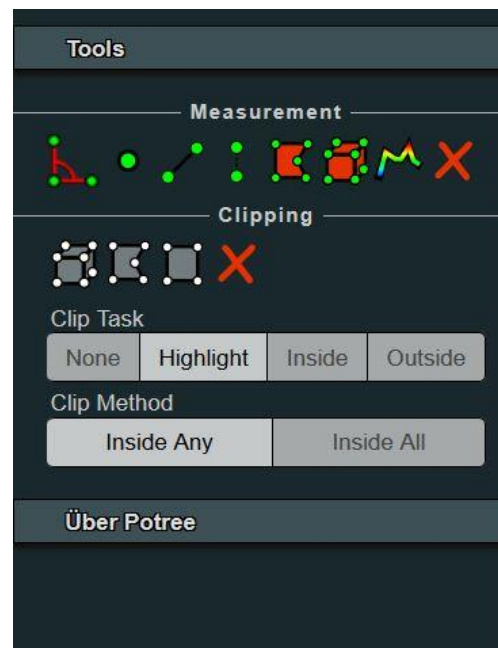
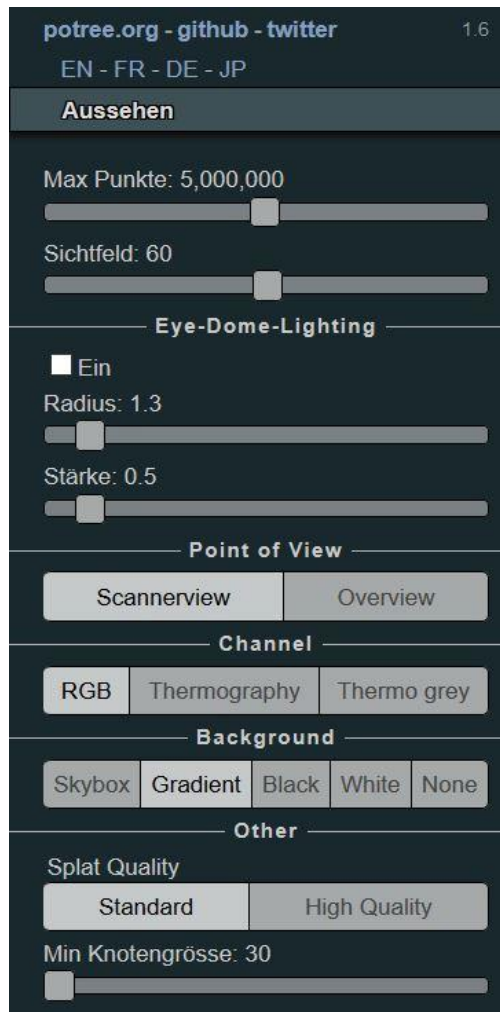


Abbildung 15: Bedienleiste nachher, eigene Darstellung, 2018

4.3 Zoomfunktion durch veränderten Navigationsmodus

Durch die Programmierung der Sicht vom Scannerstandpunkt aus (siehe 4.1) ist es nur noch schwer möglich gewesen, zu zoomen. Die Navigationssteuerung reagierte beim Bedienen des Mausekkrads kaum. Außerdem konnte man nur mittels Doppelklick zu einer Position zoomen, das Sichtzentrum befand sich dann aber nicht mehr am Scannerstandpunkt.

Es wurde nun versucht, das Mausekkrad mit einer Funktion zu belegen, die das Zoomen innerhalb der Punktwolke ermöglicht.

In Potree gibt es die Möglichkeit, zwischen mehreren Navigationsmodi, welche im JavaScript „potree.js“ umgesetzt werden, zu wählen. So funktioniert beim standardmäßig eingestellten Modus „Orbit Controls“ (wie bereits oben erwähnt) das Zoomen bei Sicht vom Scannerstandpunkt fast gar nicht. Im Modus „Earth Controls“ ist das Zoomen viel

besser möglich. Jedoch kann man in diesem Modus durch Anklicken, Festhalten und Schieben der Punktwolke mit der Maus die Punktwolke nicht um den Standpunkt drehen.

Diese sogenannte „Drag & Drop“-Funktion ist nur im „Orbit Controls“ vorhanden und wichtiger als die Möglichkeit, zu zoomen, da sonst nur ein Teil der Punktwolke betrachtet werden kann. Nun wird versucht zusätzlich zur „Drag & Drop“-Funktion, welche auf jeden Fall erhalten bleiben soll, die „scroll“-Funktion, welche für das Zoomen zuständig ist, aus dem „Earth Controls“ hinzuzufügen. Der „Orbit Controls“ Modus wird dazu bearbeitet. Die „Drag & Drop“-Funktion bleibt erhalten und die „scroll“-Funktion wird durch die „scroll“-Funktion aus dem „Earth Controls“ Modus ersetzt.

Dies erscheint zunächst recht leicht möglich zu sein, da beide Modi ähnlich aufgebaut sind und auch die „scroll“-Funktionen gleich benannt sind. Jedoch greifen die beiden „scroll“-Funktionen auf unterschiedliche Variablen zu, das heißt diese Variablen müssen ebenfalls vom „Earth Controls“ in den „Orbit Controls“ Modus übertragen werden. Das funktionierte, auch nach mehreren Versuchen, jedoch nicht.

Deswegen wird nun versucht, die „Drag & Drop“-Funktion, welche nur im „Orbit Controls“ vorhanden ist, in den „Earth Controls“ Modus einzubinden und die „scroll“-Funktion nicht zu ändern. Auch hier müssen, wegen unterschiedlicher Variablen, einige kopiert bzw. neu definiert werden. Dies funktioniert, so dass nun sowohl Zoomen als auch Drehen der Punktwolke möglich ist. Es wurde nun noch der Modus „Earth Controls“ als Standard im „potree.js“ gesetzt.

Die Steuerung ist durch die Methode des Zoomens nun schon erheblich besser als ohne Zoom. Jedoch wird wie beim Doppelklicken auch hier beim Zoomen der Standpunkt verändert, so dass man, um zum exakten Scannerstandpunkt zurückzukommen, den Button aus 4.2 anklicken muss. Als negative Folge der Standpunktänderung werden dem Nutzer die vom Scanner aus nicht sichtbaren Bereiche gezeigt. Da hier keine Punktwolke messbar war, sieht der Nutzer hier unschöne Lücken.

Dies kann verhindert werden, wenn vom Scannerstandpunkt aus gezoomt wird, das heißt wenn der Standpunkt der Sicht beim Zoomen unverändert bleibt. Diese Methode des Zoomens gibt es in Potree bereits als Funktion „Field-of-View“, steuerbar durch einen Schieberegler. Eine weitere Verbesserung für den Anwender wäre es also, durch das Mausrad das „Field-of-View“ zu ändern.

5 Zusammenfassung und Ausblick

Im Rahmen dieser Projektarbeit ist nun eine Onlinepräsentationsform für Punktwolken mit Thermografieinformation gefunden worden. Punktwolken, welche mit einem Laserscanner und einer Wärmebildpanoramaaufnahme erstellt worden sind, können jetzt anwenderfreundlich dargestellt werden. Mit Potree wurde eine Möglichkeit gefunden, Benutzern ohne Erfahrung mit Punktwolken diese durch einen frei zugänglichen Standardwebbrowser zu zeigen. Da Potree eigentlich für Punktwolken aus Befliegungsdaten entwickelt worden ist, sind einige Veränderungen bzw. Anpassungen am Programm vollzogen worden. Dazu zählen in erster Linie die Realisierung einer farbigen Darstellung der Thermografieinformation und die der Sicht vom Scannerstandpunkt. Außerdem ist eine Zoomfunktion erarbeitet und die Bedienleiste vereinfacht bzw. dem Zweck entsprechend angepasst worden. Durch die Präsentationsform können jetzt beispielsweise Kooperationspartner des Projekts eDlan der FHWS Schwachstellen in der Dämmung einfach detektieren und Maßnahmen ergreifen. Die entsprechenden Dateien können auf einen Server hochgeladen werden und es muss somit lediglich ein Link weitergegeben werden.

Durch die Darstellung der Punktwolke mittels Webbrowser kann diese nun auch auf Smartphones und Tablets betrachtet werden, wobei es sich nicht empfiehlt, da die Ansicht nicht für solche Geräte angepasst ist.

Es können jedoch auch noch einige Verbesserungen vorgenommen werden.

Wenn man einen Scanner benutzt, welcher Punktwolken mit Ursprungskoordinaten (d.h. Scannerstandpunkt = 0, 0, 0) ausgibt, muss man den Scannerstandpunkt nicht mehr für jeden Datensatz neu editieren. Ziel soll es sein, den Workflow zur Erstellung von neuen Potreeprojekten mit Thermografieinformation zu vereinfachen, so dass nicht mehr die Haupt-HTML-Seite angepasst werden muss, sondern nur noch die Dateien „sidebar.html“ und „potree.js“ ausgetauscht werden müssen.

Des Weiteren sollte es noch möglich sein, einen Kanal im LAS-Format zu finden, um die Intensität der Punktwolke mit einzuladen und eine zusätzliche Färbung zu RGB und Thermografie zu realisieren. Die Intensitätsinformation stammt direkt aus der Messung des Scanners und wurde im Gegensatz zu Thermografie und RGB nicht nachträglich den Punkten zugeordnet. Somit können geometrische Messungen in einer Intensitätspunktwolke etwas genauer sein.

Außerdem ist eine Verbesserung der Zoomfunktion sinnvoll, um dem Nutzer nicht mehr die nicht sichtbaren Teile des Aufnahmebereichs zeigen zu müssen. Eine Belegung des Mausekurses mit der vorhandenen Field-of-View-Funktion wäre hier denkbar. Ein weiteres Ziel ist es, die vorhandenen Messwerkzeuge Potree zu bearbeiten, um beispielsweise punktuell Abgreifen von Thermografieinformationen zu ermöglichen.

Eine passende Onlinepräsentationsform für Punktwolken mit Thermografieinformation kann helfen, diese besser zu teilen und zugänglich zu machen. Mit solchen Punktwolken können gezieltere Dämmmaßnahmen an Industrieanlagen durchgeführt werden. Denn schon durch kleine Dämmungen lässt sich eine enorme Menge an Energie einsparen. Dies kann ein Beitrag zur Steigerung der Energieeffizienz in Zeiten des Klimawandels sein. Deshalb lohnt es sich, an weiteren Verbesserungen bei der Detektion von Schwachstellen der Dämmung von Industrieanlagen zu arbeiten.

Literatur- und Quellenverzeichnis

- ec.europa.eu, 2018
https://ec.europa.eu/clima/policies/strategies/2030_de
Zuletzt aufgerufen: 13.08.18
- eiif.org, 2018
<https://www.eiif.org/>
Zuletzt aufgerufen: 13.08.18
- github.com, 2018
<https://github.com/potree/potree>
Zuletzt aufgerufen: 13.08.18
- LAS Specification, Version 1.4 – R13, 2013
https://www.asprs.org/wp-content/uploads/2010/12/LAS_1_4_r13.pdf
Zuletzt aufgerufen: 13.08.18
- pdal.io, 2018
<https://pdal.io/workshop/osgeo4w.html>
Zuletzt aufgerufen: 13.08.18
- potree.org, 2018
<http://potree.org/>
Zuletzt aufgerufen: 13.08.18
- Schütz, Markus, 2016
Potree: „Rendering Large Point Clouds in Web Browsers“, Seite 5
<https://www.cg.tuwien.ac.at/research/publications/2016/SCHUETZ-2016-POT/SCHUETZ-2016-POT-thesis.pdf>
Zuletzt aufgerufen: 13.08.18
- wikipedia.org, 2018
<https://de.wikipedia.org/wiki/WebGL>
Zuletzt aufgerufen: 13.08.18

Abbildungsverzeichnis

| | |
|---|----|
| Abbildung 1: Bestimmung des Punktdatenformats der LAS-Datei, eigene Darstellung, 2018 | 4 |
| Abbildung 2: Kanäle des LAS-Formats, LAS Specification, 2013..... | 5 |
| Abbildung 3: Kommandozeilenbefehl zur Erstellung eines Potreeprojekts, eigene Darstellung, 2018 | 7 |
| Abbildung 4: Kommandozeilenbefehl zur Erstellung eines Potreeprojekts (LAZ), eigene Darstellung, 2018..... | 7 |
| Abbildung 5: Gradientenmatrix (vorläufig), eigene Darstellung, 2018..... | 8 |
| Abbildung 6: Gradientenmatrix (endgültig), eigene Darstellung, 2018..... | 8 |
| Abbildung 7: Thermografieansicht mit vorläufiger Gradientenmatrix, eigene Darstellung, 2018 | 9 |
| Abbildung 8: Thermografieansicht mit endgültiger Gradientenmatrix, eigene Darstellung, 2018 | 9 |
| Abbildung 9: Gradientenmatrix im Quellcode (potree.js), eigene Darstellung, 2018. | 9 |
| Abbildung 10: Startansicht im Quellcode, eigene Darstellung, 2018 | 10 |
| Abbildung 11: Ansichtsbutter im Quellcode (sidebar.html), eigene Darstellung, 2018 | 12 |
| Abbildung 12: Button zur Kanalauswahl im Quellcode (sidebar.html), eigene Darstellung, 2018 | 14 |
| Abbildung 13: Standardwerte (potree.js), eigene Darstellung, 2018..... | 15 |
| Abbildung 14: Bedienleiste vorher, eigene Darstellung, 2018 | 16 |
| Abbildung 15: Bedienleiste nachher, eigene Darstellung, 2018..... | 17 |

Anlagen

Einwilligung zur Überprüfung einer Arbeit mit der Plagiatserkennungssoftware PlagScan

Name: Kernstock Vorname: David Matr.-Nr. 6015021
Adresse: Gneisenaustraße 24, 97074 Würzburg
E-Mail: david.kernstock@student.fhws.de Studiengang: Vermessung und Geoinformatik
Titel der Arbeit: Präsentation von Punktwolken mit Thermografieinformation in Potree
Betreuer: Dr. rer. nat. Sebastian Fiedler

Auf Grund der Zielvereinbarung der Fhws mit dem Bayrischen Staatsministerium für Bildung und Kultus, Wissenschaft und Kunst vom 19. März 2014, Ziffer 2.3, hat die Hochschule entschieden, Studien- und Abschlussarbeiten künftig durch die Plagiatserkennungssoftware PlagScan elektronisch auf Plagiate hin zu überprüfen.

Die zu überprüfenden Arbeiten werden an den Dienst Plagscan übermittelt, dort auf Übereinstimmung mit externen Quellen untersucht und zum Zweck des Abgleichs mit zukünftig zu überprüfenden Studien- und Prüfungsarbeiten gespeichert. Die befristete Speicherung Ihrer Arbeit in der Datenbank sowie die Weitergabe Ihrer persönlichen Daten im Rahmen der Plagiatsprüfung ist nur mit Ihrer Einwilligung zulässig.

Mit meiner Unterschrift erkläre ich meine Einwilligung, dass

- die von mir vorgelegte und verfasste Arbeit zum Zweck der Überprüfung auf Plagiate hin an PlagScan übermittelt und vorübergehend (5 Jahre) in der von PlagScan geführten Datenbank gespeichert wird;
- meine persönlichen Daten (Vorname, Name, studentische E-Mail-Adresse) zusammen mit dem Text digital gespeichert und verwendet werden. Diese Daten sind nur meiner Prüferin oder meinem Prüfer/meinen Prüferinnen oder Prüfern zugänglich.

Hinweis:

Diese Einwilligungserklärung ist freiwillig. Sie haben die Möglichkeit die Erklärung abzulehnen. Durch die Verweigerung der Einwilligung kann bei Entfernung der persönlichen Angaben und Wahrung der urheberrechtlichen Vorgaben die Plagiatsprüfung nicht verhindert werden. Die Einwilligung zur Speicherung und Verwendung der persönlichen Daten kann jederzeit durch Erklärung gegenüber der Fakultät mit Wirkung für die Zukunft widerrufen werden.

Ort: Würzburg

Datum: 16.08.2018

Unterschrift: _____