**Terrestrisches Laserscanning**

SS 19

GST.410UF(UE)

**Technischer** **Bericht**

Praktische Arbeiten mit dem terrestrischen Laserscanner RIEGL LMS-Z620

und

Robuste Ausgleichung einer Kugel mithilfe des Gauß-Helmert-Modells und des RANSAC-Algorithmus

Gruppe 4

Paul Arzberger   
Christian Ziesler  
Bendedikt Zörfus

Inhaltsverzeichnis

Inhalt

[Teil 1 Registrierung der Punktwolken und Transformation in ein übergeordnetes Koordinatensystem 2](#_Toc12258918)

[Teil 2 – Robuste Ausgleichung einer Kugel mithilfe des Gauß-Helmert-Modells und des RANSAC – Algorithmus 2](#_Toc12258919)

[Aufgabenstellung 2](#_Toc12258920)

[Grundlagen 2](#_Toc12258921)

[Softwareseitige Implementierung des Gauß-Hemert-Models (GHM) 5](#_Toc12258922)

[Parameterschätzung 5](#_Toc12258923)

[RANSAC 5](#_Toc12258924)

[Distanzberechnung der beiden kugelförmigen Zielmarken 7](#_Toc12258925)

[Ergebnisse 7](#_Toc12258926)

[Kugelmittelpunktskoordinaten ermittelt mit Gauss-Helmert und RANSAC 7](#_Toc12258927)

[Distanz zwischen den beiden ermittelten Kugelmittelpunktskoordinaten 9](#_Toc12258928)

[Genauigkeitsabschätzung 9](#_Toc12258929)

[Selbstständigkeitserklärung 9](#_Toc12258930)

# Teil 1 Registrierung der Punktwolken und Transformation in ein übergeordnetes Koordinatensystem

# Teil 2 – Robuste Ausgleichung einer Kugel mithilfe des Gauß-Helmert-Modells und des RANSAC – Algorithmus

## Aufgabenstellung

Aus den beiden aufgenommen Punktwolken, welche in das Ziellandessystem angehangene Koordinatensystem Gauss-Krüger transformiert wurden, sollen zwei kugelförmige Zielmarken extrahiert werden und deren Position zueinander ermittelt werden. Hierfür soll ein Ausgleich mit einer Nebenbedingung auf die Daten angewandt werden um die Kugelparameter der Position und Dimension zu schätzen. Mittels dem RANSAC Algorithmus soll iterativ die beste Schätzung der Kugelparameter gefunden werden, welche abgebildet auf alle Messungen die meisten Inlier liefert. Hierzu wurde eine Software in Python 3.7 geschrieben und wird im folgenden Kapitel beschrieben

## Grundlagen

Wir stellen die Behauptung auf das mit folgender Formel eine (nicht verschobene) Kugel implizt mit den (wenn fehlerfrei) getätigten Messungen dargestellt werden kann.

Da aber bei der Messung mit dem Laserscanner Messfehler getätigt werden, kann die oben erwähnte Kugelgleichung nicht 100% widerspruchsfrei sein und diese Widersprüche müssen berücksichtigt werden.

Durch Verbesserung ***v*** der Beobachtungen ***l*** und durch die Berücksichtigung der Widersprüche ***w*** wird die Bedingungsgleichung wieder konsistent. Der funktionale Zusammenhang liest sich für eine verschoben Kugel wie folgt beschreiben

Der funktionale Zusammenhang wird mittels Taylorreihenentwicklung bis zum linearen Therm an den Stellen ***v0*** und ***x0*** für die Berechnung der Verbesserungen ***v*** und Zuschläge ***∆x*** iterativ berechnet. Da für den iterativen Vorgang Näherungswerte der Kugelposition für ***xc***, ***yc***, ***zc*** sowie dem Radius ***r*** benötigt werden (die auch auf oder innerhalb der Kugel liegen müssen), wird jene Messung mit dem größten Z-Wert extrahiert. Diese Messung wird in die Z-Richtung noch um den geschätzten initialen Radius der Kugel vermindert, um im Anschluss als heuristischer Mittelpunkt für den Ausgleich verwendet zu werden. Der ganze Algorithmus kann als Minimierungsaufgabe mit Nebenbedingung verstanden werden und erfolgt unter Verwendung eines Lagrange-Multiplikators ***k***, oder Korrelat. Die Kovarianzmatrix der Beobachtungen Σπ wird als Einheitsmatrix angenommen, da die Messungen alle als gleichgewichtet angesehen werden.

Die Designmatrix ***A***, die Bedingungsmatrix ***B***, der Widerspruchsvektor ***w***, die Korrelaten ***k***, die Verbesserungen ***v***, die Normalgleichungsmatrix ***N*** und die ausgeglichen Parameter ***x*** des linearisierten funktionalen Zusammenhangs setzen sich wie folgt zusammen:

Designmatrix ***A***  


Bedingungsmatrix ***B***



Widerspruchsvektor *w*



Hilfsvariable ***ri***



Aus der um die Nebenbedingung erweiterte Minimumsforderung ( Angabe Blatt Formel (7) ) wird folgende zu minimierende Funktion

Diese wird partiell nach ***v***, ***x***, und ***k*** abgeleitet.



Die Korrelaten k lassen sich wie folgt bestimmen:

)

Die Zuschläge ***∆x*** werden wie folgt ermittelt

Und werden zum bestimmen der ausgeglichenen Parameter herangezogen.

Die Standardabweichung der Gewichtseinheit ***σ0***erhält man durch



Und die Genauigkeit der ausgeglichenen Parameter x ergeben sich aus der Kovarianzmatrix ***Σxx***



## Softwareseitige Implementierung des Gauß-Hemert-Models (GHM)

Das oben diskutierte GHM in Kombination mit dem RANSAC Algorithmus wurde mittels Python 3.7.1 und den Bibliotheken numpy 1.15.4 sowie matplotlib 3.0.2 (mpl) unter Win7 64bit, Win 10 64bit sowie Ubuntu 16.04 64bit entwickelt und getestet. Die Software ist in zwei Teile untergliedert. Einerseits dem Hauptprogramm ***main.py*** welches über ein Commandline Interface (CMD) aufgerufen werden muss, da es zwei Inputparameter übergibt

1. Den Dateipfat/Dateinamen der zu processierenden TLS Daten
2. wie viele zufällig ausgewählte Messdaten für die Parameterbestimmung herangezogen werden müssen

***python main.py filename 4***

Optional können noch verbose Argumente mit übergeben werden, um mehr Informationen über die aktuellen Arbeitsschritte der Software zu erhalten. Das optionale Argument -v, -vv und -vvv stellen drei unterschiedliche Detaillevel zusätzlicher Information dar, um bei eventuell auftretenden Schwierigkeiten bei der Fehlersuche zu unterstützen.

***python main.py filename 4 -v***  
  
Andererseits der ***TLSToolbox.py***, einer Funktionssammlung die im Hauptprogramm eingebunden wird. Die Distanzberechnung wird mittels des Codes in ***TLSDistanceCalc.py*** durchgeführt. In ihr befinden sich die gemittelten Werte der Kugelmittelpunkte aus der Tabelle 1 - 4. Die Berechnung der Distanz der Kugelmittelpunkte wird im nachfolgenden genauer beschrieben.

### Parameterschätzung

Die Parameterschätzung für die Kugelparameter ***x***, ***y***, ***z*** und ***r*** werden iterativ bestimmt und zwar bis der maximale Zuschlag ***∆x*** *1e10-6* als Schwellwert unterschreitet. Das GHM wurde als Funktion *gauss\_helmert\_model()* implementiert und verlangt als Input einen Datenblock von der Dimension nx3, die Messung mit dem maximalen Z-Wert in der Dimension 1x3, einen Figure-Instance von *matplotlib*, sowie einen *verbose-Flag* um unterschiedliche Ausgabeinformationslevel an die Konsole auszugeben. Als Rückgabewerte liefert die Funktion die ausgeglichenen Parameter, die Standardabweichung der Gewichtseinheit, die Kovarianzmatrix der Parameter, die mpl figure-Instanz, die mpl Ax-Instanz und einen True/False Flag namens ***det\_ok***. Dieser ist insofern wichtig da eine Abfrage im Hauptprogramm überprüft ob die Determinante im GHM bestimmt werden konnte und falls nicht wird in weiterer Folge die gerade durchlaufene Schleifeniteration nicht als RANSAC-Iteration gezählt.

### RANSAC

Der RANSAC Algorithmus ist wie schon das vorangegangene GHM von iterativem Charakter. Der Algorithmus wählt zufällig eine Mindestanzahl von 4 Messungen (***s=4***) aus dem Datenblock und schätzt ungeachtet der Güte oder Sinnhaftigkeit der Messungen das Kugelmodell.

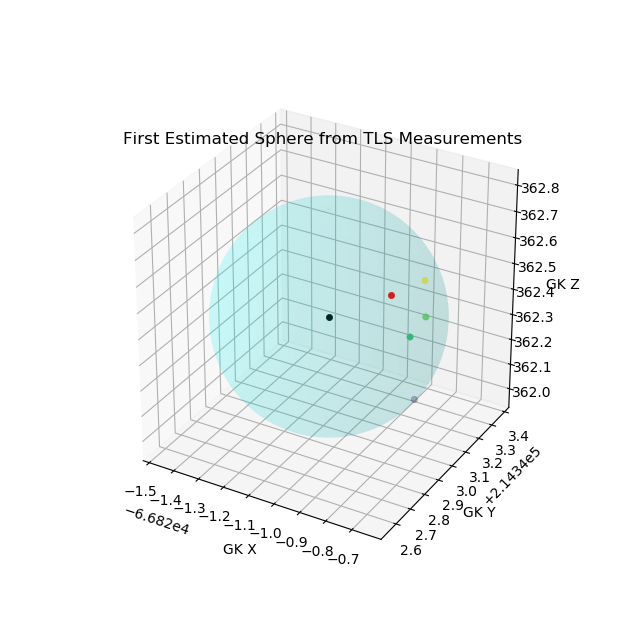


Abbildung 1 aus einer zufälligen Auswahl von 4 Messungen geschätzes Kugelmodel. rot = initialer Kugelmittelpunkt, schwarz = verbesserter Kugelmittelpunkt, farbig = Messungen

Die geschätzten Parameter werden in die Kugelgleichung eingesetzt und **alle** Messdaten einem Distanztest unterzogen.

Jene Messdaten die in die eingesetzte Kugelgleichung eine Differenz von mehr als einem zuvor festgelegtem Schwellwert überschreiten, werden als Ausreißer oder ***Outlier*** klassifiziert, alle die innerhalb des Schwellwertes liegen als ***Inlier***. Der ***Schwellwert*** wurde mit der *Herstellerangabe der* *Messungenauigkeit* angenommen und beträgt *1cm*. Die Anzahl der Inlier und Outlier wird bestimmt und auf die Variablen ***sum****\_****inliers*** und ***sum\_outliers*** geschrieben. Sie sind insofern von Bedeutung da die Anzahl der RANSAC Iterationen von der Anzahl der Outlier direkt abhängt.

Die Natur des RANSAC erlaubt es nicht zu 100% das beste Modell zu schätzen, sondern nur zu einer gewissen Wahrscheinlichkeit, welche mit ***p=99%*** angenommen wird. Diese Wahrscheinlichkeit von 99% und die Anzahl der detektierten Outlier nimmt direkt Auswirkung auf die Anzahl der Iterationen, die der RANSAC durchlaufen wird. Zu aller erst wird die Anzahl der Iterationen in der Software mit Unendlich angesetzt. Die Iterationen werden über eine Whileschleife gesteuert, die so lange läuft, bis die berechnete Anzahl der Iterationen ***N*** kleiner den bisherigen Durchläufen ***counter\_ransac\_loop*** der Whileschleife wird. Mit der ersten Schätzung und der ersten verfügbaren Anzahl an Outliern, wird eine neue Anzahl an nötigen Iterationen berechnet, die gewähren soll das der Algorithmus zu einer 99% Wahrscheinlichkeit ein richtiges Modell liefert. Diese Anzahl der Iterationen N wird berechnet und im Iterationsalgorithmus des RANSAC upgedatet.

Die Anzahl der Inliers, sowie die geschätzten Parameter werden ebenfalls verspeichert und mit jenen der nächsten Iteration verglichen. Ist Die Anzahl der Inlier der neuen Iteration höher als die bisherigen, werden die Schätzparameter in der Variable ***best\_sphere\_estimate*** mit den neuen Werten überschrieben und ***N*** für die Anzahl der Iterationen wird neu berechnet. Dies geschieht nun so lange bis die Anzahl der Iterationen durchlaufen sind oder die Abbruchbedingung in der Whileschleife mit ***False*** evaluiert wird. Wird im GHM jedoch beim Bestimmen der Determinante der Normalgleichungsmatrix ***N*** ein Schwellwert unterschritten, gilt das Gleichungssystem als nicht brauchbar, die zufällige Auswahl der Daten wird ignoriert sowie NICHT in den Iterationen mitgezählt und der Algorithmus beginnt von neuen.

Jene Parameter mit den meisten Inliers werden verspeichert und als Endergebnis ausgegeben. Um die Distanz der Kugeln zueinander zu bestimmen wird für jede Scanposition (SP) das Programm 3 mal ausgeführt, tabellarisch festgehalten und die Positionsdaten der Kugel gemittelt.

### Distanzberechnung der beiden kugelförmigen Zielmarken

Da die beiden Zielmarken in sehr geringem Abstand zueinander stehen müssen keine Krümmungseffekte berücksichtigt werden. Die Distanz lässt sich sehr einfach über den Betrag des Richtungsvektors der beiden Kugelmittelpunkte zueinander bestimmen.

Ergebnisse

### Kugelmittelpunktskoordinaten ermittelt mit Gauss-Helmert und RANSAC

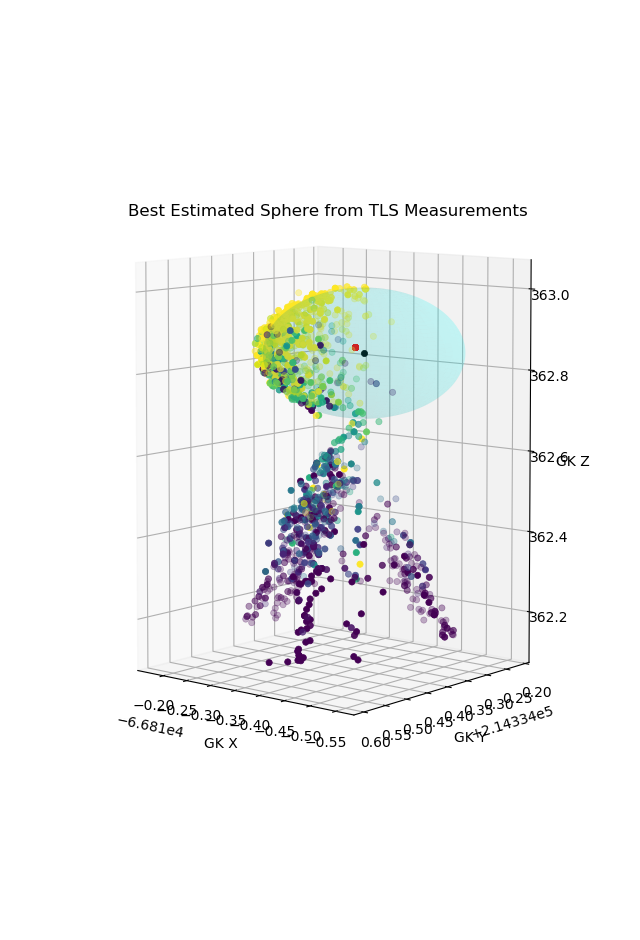
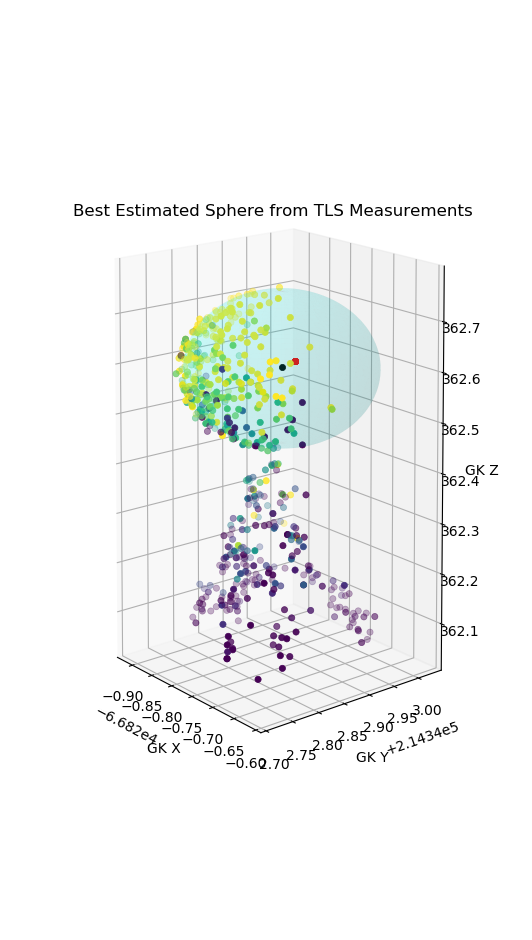


Abbildung 2 – Links Messungen und Kugelinterpolation von Kugelzielmarke 1 von Scanposition 1 mit ausgeglichener Kugeldarstellung   
- Rechts Messungen und Kugelinterpolation von Kugelzielmarke Kugel 1 von Scanposition 2 – mit ausgeglichener Kugeldarstellung  
rot = initialer Kugelmittelpunkt, schwarz = verbesserter Kugelmittelpunkt, farbig = Messungen, cyan=geschätze Kugel

Tabelle 1 Kugelmittelpunkt der ersten Zielmarke von Scanposition 1 gerechnet und gemittelt

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SP1 Kugel 1 | 1 | 2 | 3 | Mittel |
| Inliers/Messungen | 5321/10286 | 5392/10286 | 5326/10286 | 5346,3/10286 |
| X | -66820,767 m | -66820,764 m | -66820,762 m | -66820,7643 m |
| Y | 214342,870 m | 214342,875 m | 214342,878 m | 214342,8743 m |
| Z | 362,601 m | 362,595 m | 362,593m | 362,5963 m |
| R | 0,1466 m | 0,1524 m | 0,15313m | 0,15071 m |

Tabelle 2 Kugelmittepunkzt der ersten Zielmarke von Scanposition 2 gerechnet und gemittelt

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SP2 Kugel 1 | 1 | 2 | 3 | Mittel |
| Inliers/Messungen | 1524/2874 | 1526/2874 | 1519/2874 | 1523/2874 |
| X | -66820,776 m | -66820,786 m | -66820,770 m | -66820,777 m |
| Y | 214342,864 m | 214342,864 m | 214342,865 m | 214342,8643 m |
| Z | 362,596 m | 362,598 m | 362,598m | 362,5973 m |
| R | 0,1535 m | 0,1600 m | 0,1468m | 0,15343 m |

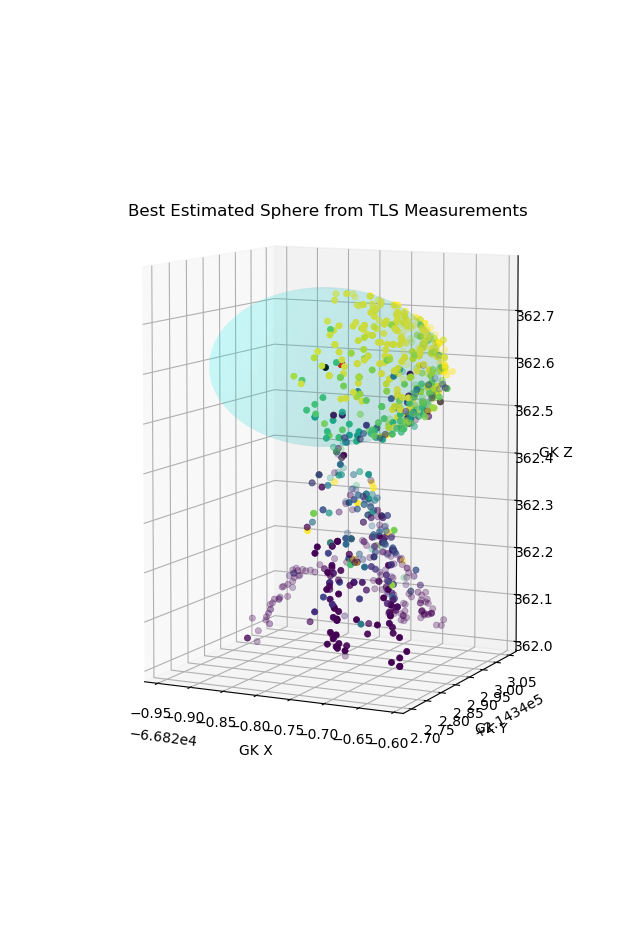
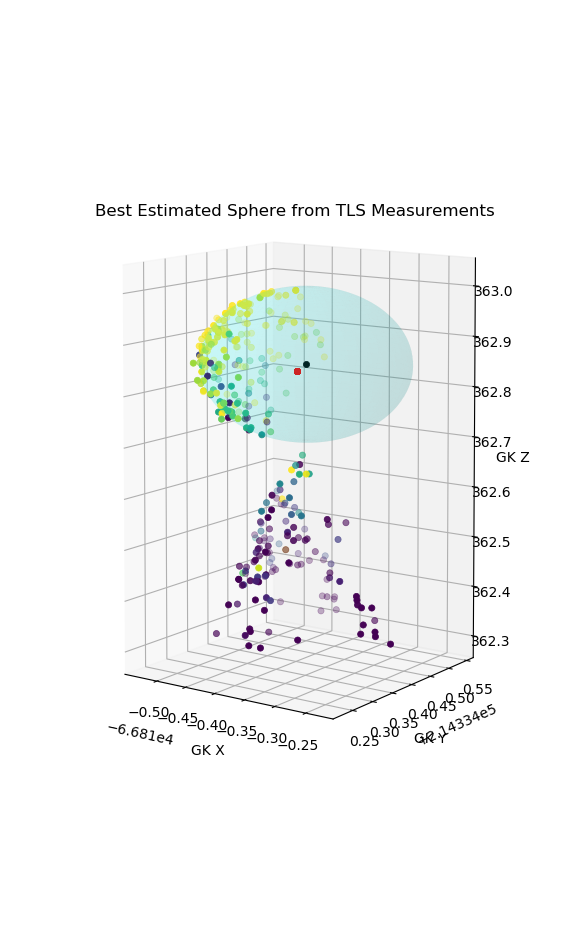


Abbildung 3 – Links Messungen und Kugelinterpolation von Kugelzielmarke Kugel 2 von Scanposition 1 – mit ausgeglichener Kugeldarstellung  
 - Rechts Messungen und Kugelinterpolation von Kugelzielmarke Kugel 2 von Scanposition 2 – mit ausgeglichener Kugeldarstellung  
rot = initialer Kugelmittelpunkt, schwarz = verbesserter Kugelmittelpunkt, farbig = Messungen, cyan=geschätze Kugel

Tabelle 3 Kugelmittepunkt der zweiten Zielmarke von Scanposition 1 gerechnet und gemittelt

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SP1 Kugel 2 | 1 | 2 | 3 | Mittel |
| Inliers/Messungen | 2907/5123 | 2981/5123 | 3059/5123 | 2982,3/5123 |
| X | -66810,385 m | -66810,374 m | -66810,374 m | -66810,3776 m |
| Y | 214342,384 m | 214342,390 m | 214342,391 m | 214342,3883 m |
| Z | 362,851 m | 362,853 m | 362,851m | 362,8516 m |
| R | 0,1465 m | 0,1495 m | 0,1537m | 0,1499 m |

Tabelle 4 Kugelmittepunkzt der zweiten Zielmarke von Scanposition 2 gerechnet und gemittelt

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SP2 Kugel 2 | 1 | 2 | 3 | Mittel |
| Inliers/Messungen | 3560/7617 | 3476/7617 | 3496/7617 | 3510,6/7617 |
| X | -66810,390 m | -66810,390 m | -66810,390 m | -66810,390 m |
| Y | 214342,382 m | 214342,384 m | 214342,381 m | 214342,3823 m |
| Z | 362,852 m | 362,856 m | 362,846 | 362,8513 m |
| R | 0,1482 m | 0,1489 m | 0,1487 m | 0,1486 m |

### Distanz zwischen den beiden ermittelten Kugelmittelpunktskoordinaten

Distanz Kugel 1 – Kugel aus SP1: 10.4011 m  
Distanz Kugel 1 – Kugel aus SP2: 10.4012 m

### Genauigkeitsabschätzung

Alle von der Software gelieferten Ergebnisse bewegen sich innerhalb der vom Hersteller angegebenen Toleranz von 1cm.

Selbstständigkeitserklärung  
Hiermit bestätigen wir, die Ergebnisse nach besten Gewissen selbstständig erstellt und interpretiert zu haben.

Paul Arzberger, Christian Ziesler, Benedikt Zörfus Graz am 30.05.2019