

# Entwicklerhandbuch für das Thema *„3D-Punktwolkenregistrierung mit Festkomma-Arithmetik“*

Betreuer: *Andreas Nüchter*

*Tom Fleischmann, Jonas Wiesner, Yannik Winzer*

*28.08.2025*

## **1 Beschreibung**

Die entwickelte Software ist eine Ergänzung des 3D-Toolkit 3DTK des Lehrstuhls 17 der Julius-Maximilians-Universität Würzburg, das Algorithmen und Methoden zur Verarbeitung von 3D-Punktwolken bereitstellt.

Das vorhandene Softwarepaket wurde um die Möglichkeit, den ICP-Algorithmus zur 3D-Punktwolkenregistrierung mit Festkomma-Arithmetik durch die Verwendung von SystemC auszuführen, erweitert.

Dafür waren umfangreiche Arbeiten im Hintergrund (Anpassungen der CMake-Listen und Neu- oder Re-Implementierungen der nötigen Funktionen) notwendig.

## 2 Backend

### 2.1 Übersicht

Nach dem in 2.5 beschriebenen Build-Vorgang kann der implementierte Algorithmus mit Hilfe des Befehls `bin/icpFixpoint -s <S> -e <E> <V>` mit Nummer des Start-Scans <S> (z.B. <S> = 0), Nummer des End-Scans <E> (z.B. <E> = 1) und Verzeichnis <V> der Scans, z.B. <V> = `dat`, aufgerufen werden.

Alle (weiteren) Parameter (z.B. maximale Anzahl der Iterationen, maximale Distanz zwischen Punkten für das Matching, Epsilon-Wert für die Terminierung des ICP-Algorithmus) sind in der Hilfe, die mit `bin/icpFixpoint -help` aufgerufen werden kann, genauer beschrieben.

### 2.2 Datei-/Ordnerstruktur

*Alle angelegten Code-Dateien werden in den Headern ebenfalls kurz charakterisiert.*

- Ordner `src`: enthält die eigentlichen Code-Dateien
  - Ordner `sc_fixed`
    - \* Datei `sc_fixed_converter.cc`: enthält Methoden zum Konvertieren von `double`-Arrays in `f_float`-Arrays und -Vektoren
    - \* Datei `sc_fixed_math.cc`: enthält mathematische Funktionen in SystemC-Festkomma-Arithmetik (Wurzelziehen durch `sc_fixed_heron_sqrt` mit dem Heron-Verfahren, Cholesky-Zerlegung sowie die Transformation der Scans)
    - \* Datei `sc_ICP.cc`: enthält die `match`-Methode, die den ICP-Algorithmus durchführt
    - \* Datei `sc_ICPapx.cc`: enthält die `Align`-Methode, die den gegebenen Data-Scan am Source-Scan ausrichtet
  - Ordner `slam6D`: Datei `icpFixpoint.cc`: Hauptdatei des Projekts für den ICP-Algorithmus
- Ordner `include/sc_fixed`: enthält die zu den oben genannten Dateien gehörigen Header-Dateien (`sc_fixed_converter.h`, `sc_fixed_math.h`, `sc_ICP.h`, `sc_ICPapx.h`) sowie `sc_ICP.icc` und die Header-Datei `sc_ICPminimizer.h`, die in `sc_ICPapx` eingebunden ist
- Ordner `bin`: enthält alle Executables; nun zusätzlich das Executable `icpFixpoint`
- Ordner `doc`: enthält zu Dokumentationszwecken das Pflichtenheft sowie dieses Entwicklerhandbuch

### 2.3 Genutzte Abhängigkeiten

Es wird lediglich die SystemC-Library verwendet, wie in 2.4 beschrieben.

### 2.4 Einrichten der Entwicklungsumgebung

- Installation der für das 3DTK notwendigen Software, wie sie hier beschrieben wird
- Installation der SystemC-Library, z.B. mittels `sudo apt install libsystemc-dev`
- Der Build-Vorgang zur Entwicklung entspricht dem in 2.5 beschriebenen Vorgang für den Produktiveinsatz

### 2.5 Build

Durch den Aufruf von `make` im Hauptverzeichnis wird das gesamte Projekt gebaut/kompiliert. Alternativ kann das Projekt mit `make config` genauer konfiguriert werden (benötigt `ccmake`). Des Weiteren ist die Konfiguration im Verzeichnis `.build` wie folgt möglich:

```
mkdir .build
cd .build && cmake ..
```