

Projektgruppe FastSense

# **Meilenstein 3**

TODO: Bild

25. Januar 2021

# Inhalt

Recap MS 2

Ziele für MS 3

Ergebnisse MS 3

Drohne, Laserscanner

Aufbau

Base Design

Kommunikation

Algorithmus

Mesh Rekonstruktion

Paper

---

Evaluation

Zeit

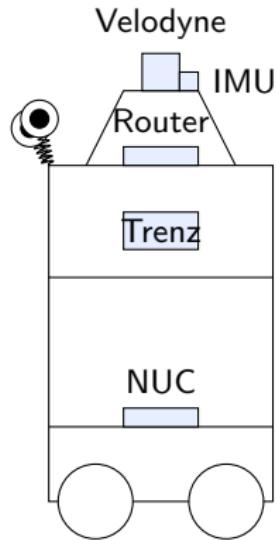
Power Consumption

Qualität

Demonstration

Ausblick

# Recap MS 2



- Grundlegender Hardware Accelerated TSDF SLAM Algorithmus fertig
- Zeit: 0,87 fps
- Power Consumption: 10,32 W
- Verbesserungspotenzial vorhanden

# Ziele für MS 3

- Aufbau einer SLAM-Box
  - Nutzung als Sensor
  - Einfache Portierung zwischen Drohne, Roboter, Rucksack etc.
  - Festes Interface, einfache Bedienung, Kapselung
- Verbesserung und Optimierung des Algorithmus

Variable	Ziel
Genauigkeit	Wiederfinden erneuter Pose
Energie	0,5 J/frame
Frequenz	<b>20 fps</b> (echtzeitfähig)
Geschwindigkeit	10 km/h

- Mesh-Generierung auf Basis der TSDF Werte

# Ergebnisse MS 3

- Aufbau einer SLAM-Box → **Aufbau**
  - Nutzung als Sensor
  - ~~Einfache Portierung zwischen Drohne, Roboter, Rucksack etc.~~
  - Festes Interface, einfache Bedienung, Kapselung
- Verbesserung und Optimierung des Algorithmus
  - **Base Design**
  - **Kommunikation**
  - **Preprocessing**
  - **Registrierung vollständig auf Hardware**
  - **Asynchrones TSDF Update**
- Mesh-Generierung auf Basis der TSDF Werte
  - **Mesh Rekonstruktion**

# Drohne, Laserscanner

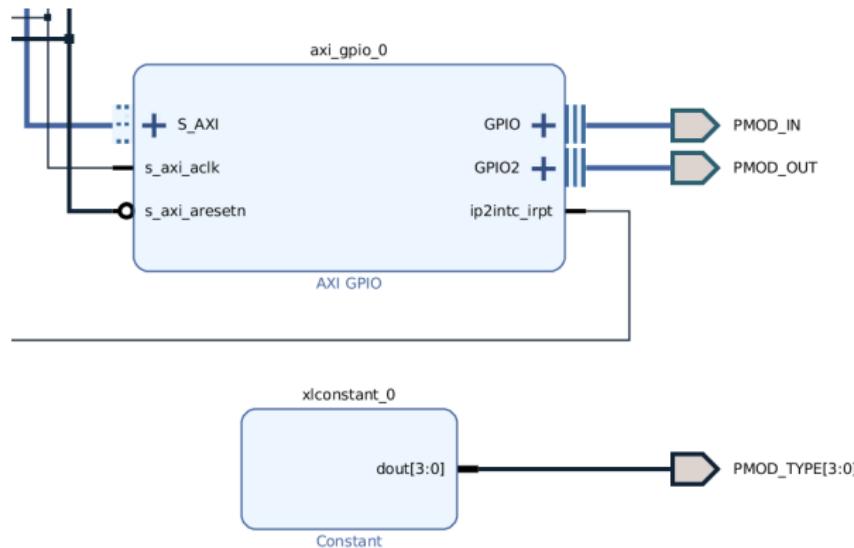
- Drohne erfordert neuen Laserscanner
    - Velodyne nicht geeignet
  - Kontakt mit Firmen:
    - Ouster
    - Blickfeld
- Future Work



# Aufbau

TODO: Steffen mit Bildern von Adrian

# Base Design

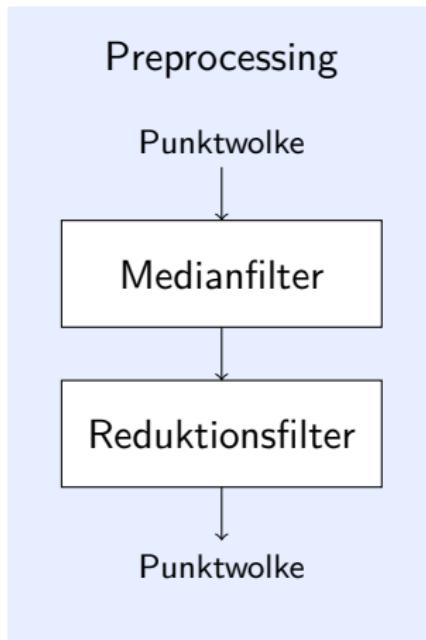


- GPIO
- Autostart
- Beliebige Taktrate für die Kernel

# Kommunikation

- Kommunikation funktioniert in beide Richtungen
  - Trenz → Host (MS02)
  - Host → Trenz
- **Grundlage für Evaluierung und Reproduzierbarkeit**
- Speichern des Poseverlaufs
  - **Visualisierung der Abweichungen und Pfades in RViz**
- Nicht-Blockend
  - **Sauberer Abspeichern der Global Map**

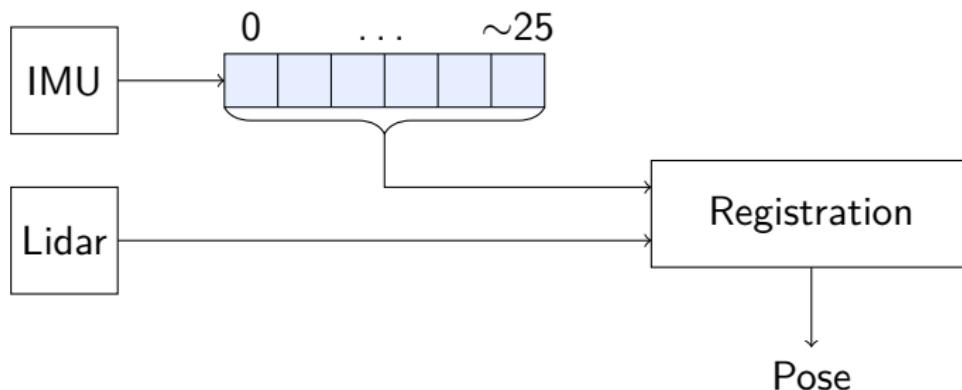
# Algorithmus: Preprocessing Punktwolken



- Parallelisiert
- Verschiedene Varianten für den Reduktionsfilter
  - Average
  - Voxel Center
  - Random Point
  - Closest to Center
- Kaum Unterschiede

# Algorithmus: Preprocessing IMU

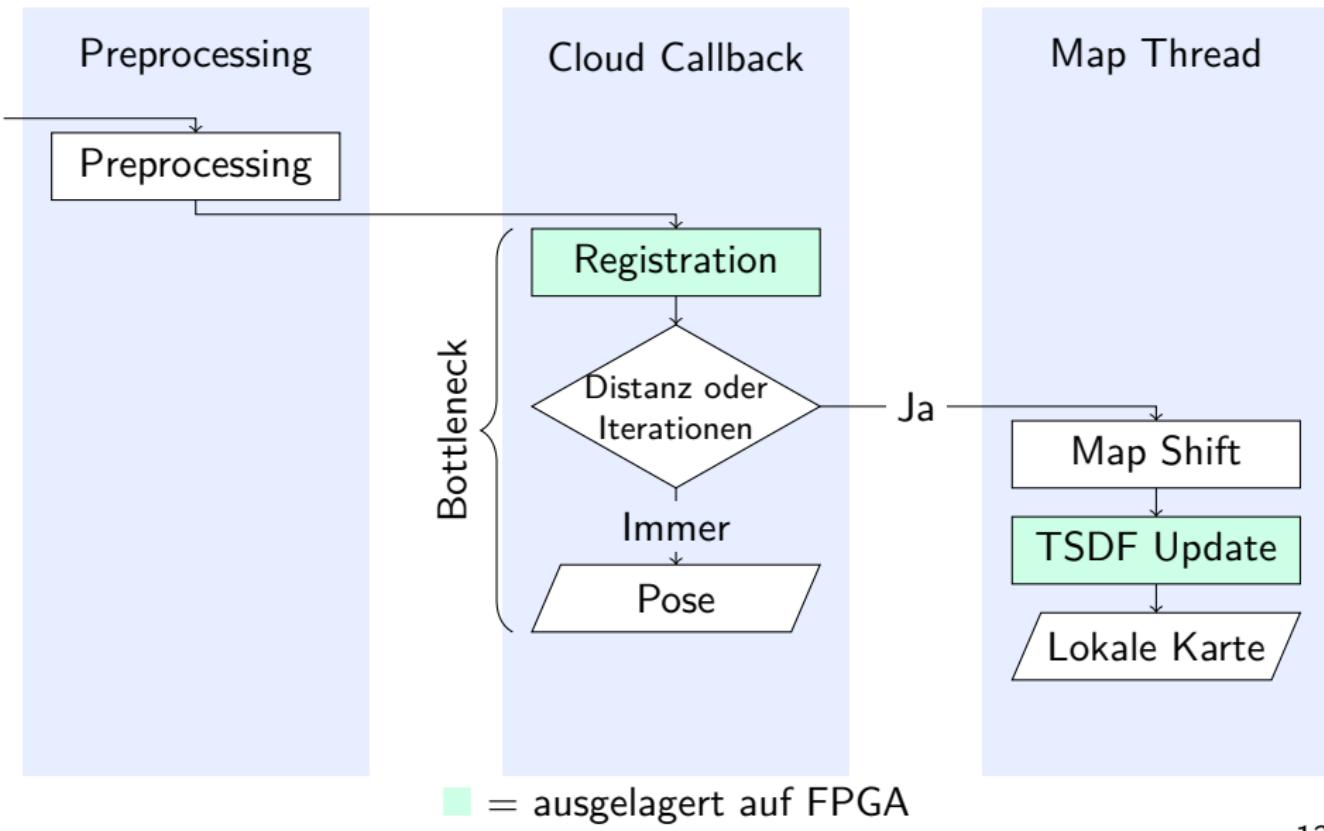
- Vor Iteration des Algorithmus: Konfigurierbarer Rolling Average Filter
- Während des Algorithmus:  
Akkumulation bis zur nächsten Punktwolke



# Algorithmus: Registrierung

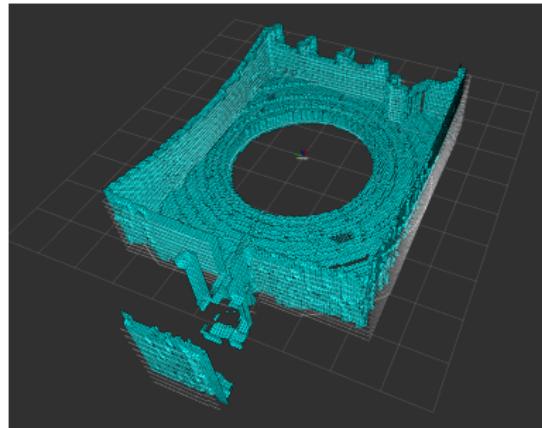
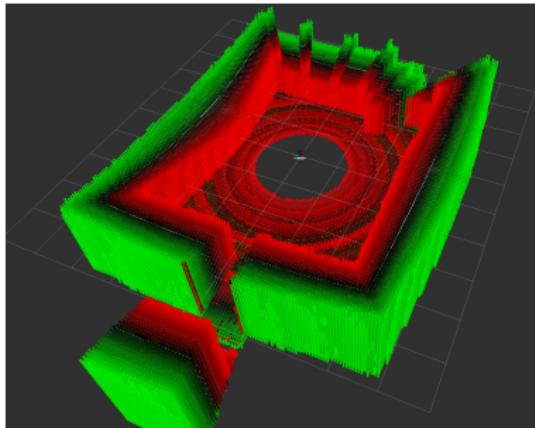
- |   |        |
|---|--------|
| • Vollständig auf HW                        | 676 ms |
| • Solver                                    | 74 ms  |
| • Float Matrizen auf HW                     |        |
| • Split & Memports                          |        |
| • Erst 2×                                   | 57 ms  |
| • Dann 3×                                   | 50 ms  |
| • Abbruchkriterium: $\varepsilon$ verändert |        |
| • $0,0001 \rightarrow 0,01$                 | 44 ms  |
| • $0,01 \rightarrow 0,04$                   | 30 ms  |
| • Rotation gefixt                           |        |
| • Relativ zum Scanner statt zum Ursprung    |        |

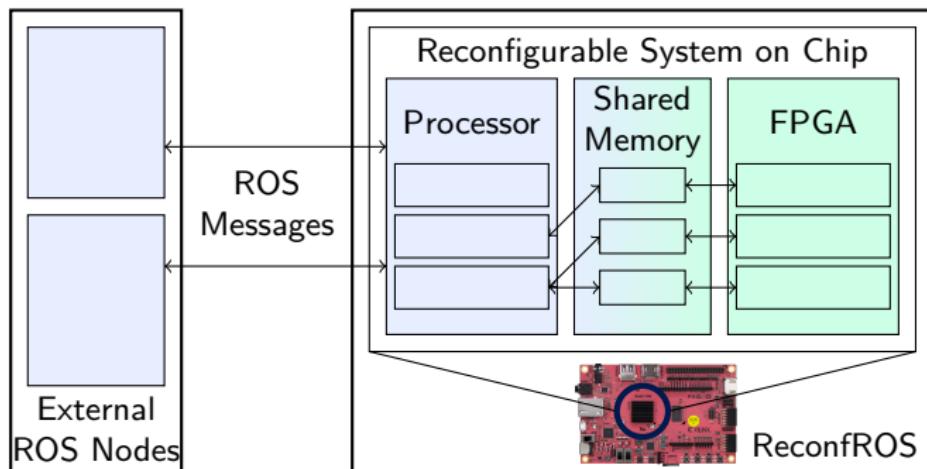
# Algorithmus: Asynchronität



# Mesh Rekonstruktion

- Global Map offline
  - Programm im LVR2 Repository
  - Mesh Verbesserungen
  - HDF5 → PLY
- Local Map online
  - ROS Node
  - Marker Message → Mesh Message





Camera image



Removing noise



Trail pixel extraction



Thresholding

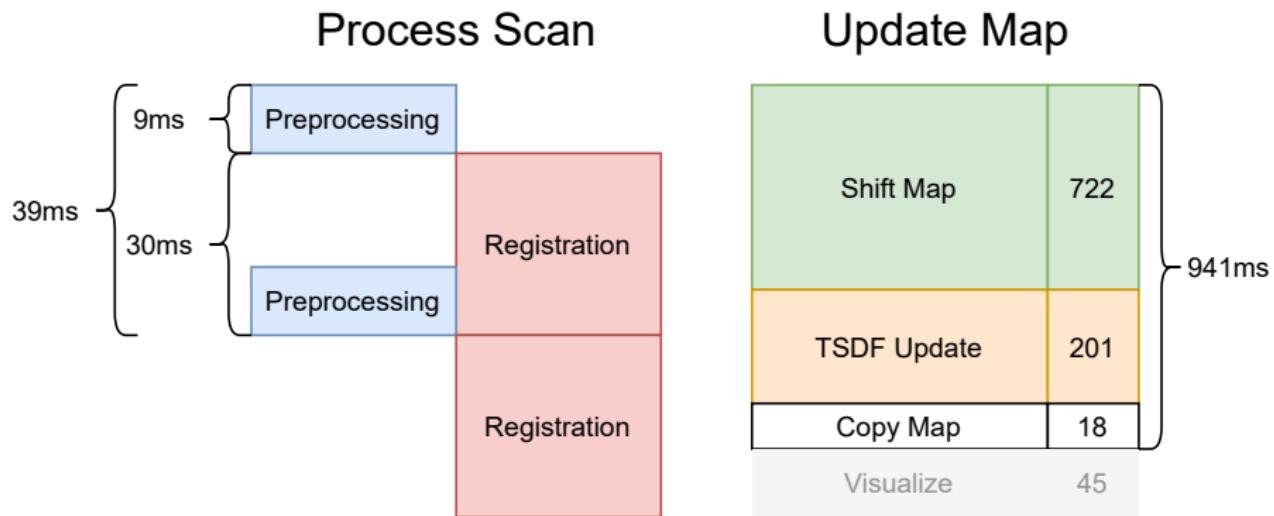


Remove fragments



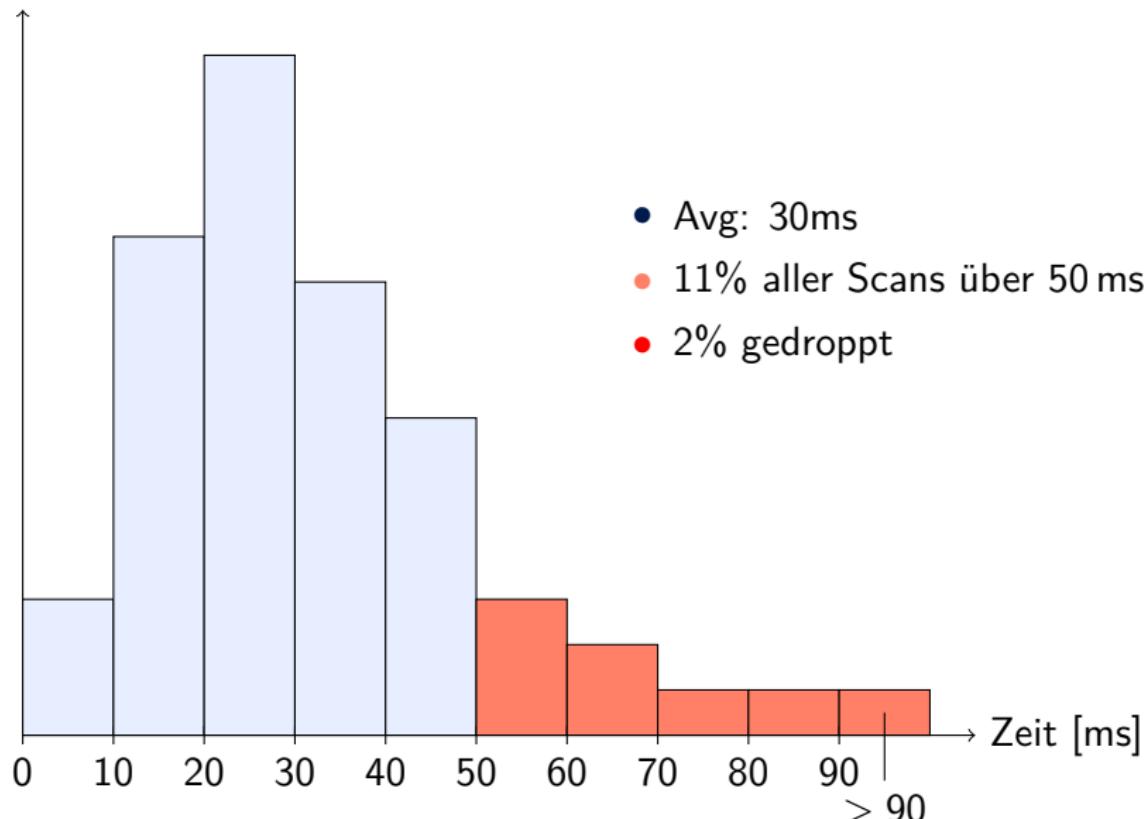
Trail direction

# Evaluation: Zeit



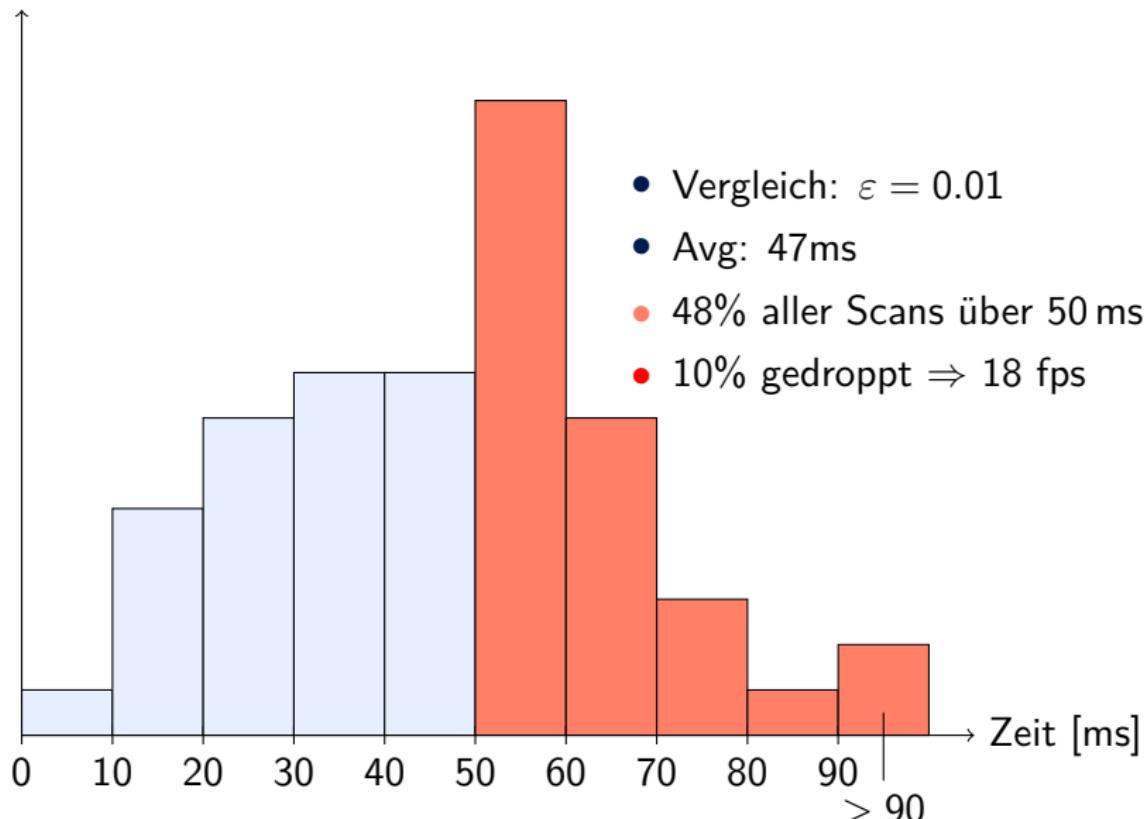
## Evaluation: Zeit

Anzahl Frames



# Evaluation: Zeit

Anzahl Frames



## Evaluation: Power Consumption

TODO: Adrian

# Evaluation: Qualität

- 6. Stockwerk (Distanzabweichung in Meter)

epsilon	0.01	0.04
	0.0615	0.0505

Geschw.	langsam	schnell
	0.0505	0.0437

- gesamt (langsam, epsilon = 0.04): 0.075349
- 8 Meter Labortest (Distanzabweichung in Meter)

	hin	zurück	gesamt
langsam	0.0548	0.0650	0.0861
schnell	0.1676	0.0459	0.1320

- Leerlauf (nach einer Stunde, epsilon = 0.04): 0.156095

# Ausblick bis Hauptvortrag

- Aufnahme eines Teils der Innenstadt auf Kettcar
- Aufarbeitung theoretischer Hintergründe  
(Registrierung, Kartierung, etc. . . )
- Vervollständigung der Dokumentation und  
Static Code Checking
- Erstellung einer Übersicht über den gesamten Algorithmus

# Gesamtausblick

- FastSense Paper
- Loop Closing
- Drohne
- Modulares Design
- Grillen bei Mario

Ende

Vielen Dank für Eure Aufmerksamkeit!

Fragen?