

# PROJEKT: FK08 NAVI APP

## Kap. 1 ALLGEMEINES

- Raum F230
- Login auf PCs mit Adminrechten: XXXX

### 1.1 ALLGEMEINES

Ziel der Lehrveranstaltung ist die Entwicklung von Lokalisations- und Navigationsalgorithmen zur Lösung komplexer Probleme aus dem Bereich der mobilen Robotik und Navigation sowie deren Implementierung und Integration in die zur Verfügung gestellten Sensor- und Systemumgebungen. Die Lösungswege werden in Form von Teilprojekten selbst und im Team entwickelt und sukzessive zu einer Applikation zusammengeführt. Hierbei stehen mehrere Sensoren zur Verfügung.

### 1.2 TERMINE UND ANFORDERUNGEN

Das Projekt soll in Gruppenarbeit (**Gruppen zwischen 2-3 Personen**) bearbeitet werden.

- *Praktische Arbeit (50% Notengewichtung):*
  - ⇒ Ein Großteil der programmierrelevanten Arbeiten sollte zeitlich in der Lehrveranstaltung durchgeführt werden können
- *Präsentation (25% Notengewichtung)*
  - ⇒ *1. Präsentationstermin (Zwischenpräsentation zur Semestermitte):*
    - Individuell oder als Gruppenpräsentation / Eigenanteil muss ersichtlich sein
    - Hier sollen
      - der Projektstand vorgestellt werden
      - die Ergebnisse einer Literaturrecherche vorgestellt werden
    - Dauer pro Teilnehmer ca. 10 Minuten
    - Umfang pro Teilnehmer ca. 5-7 Folien
  - ⇒ *2. Präsentationstermin (Abschlusspräsentation zu Semesterende):*
    - Individuell oder als Gruppenpräsentation / Eigenanteil muss ersichtlich sein
    - Als Poster (Demo Movie erwünscht)
    - Hier soll das Projektergebnis präsentiert werden
    - Termin **voraussichtlich 2. Januarwoche**
- *Dokumentation (25%):*
  - ⇒ Vervollständigen Sie Ihr Projekt, indem Sie alle wichtigen Schritte dokumentieren! Verwenden Sie hierzu die publish Methode von Matlab
  - ⇒ Die Dokumentation ist zweitens um einen **Bericht im Sinne einer wissenschaftlichen Veröffentlichung** zu vervollständigen (Template siehe Moodle). Der Umfang soll pro Teilnehmer bei ca. 2 Seiten liegen.
  - ⇒ Gruppenarbeit erlaubt / Eigenanteil muss ersichtlich sein
  - ⇒ Abgabe bis **13.01.2019**

## Kap. 2 PROJEKTBESCHREIBUNG

### 2.1 PROJEKTZIEL UND KURZBESCHREIBUNG DES PROJEKTES

Zu entwickeln ist der Prototyp einer **Indoor Navigations App** unter Verwendung von Matlab wie folgt:

- Ihre Aufgabe ist es, den Prototyp einer Navigations App zu implementieren, mit der Sie sich in einem bekannten Gebäude lokalisieren und zu einem Ziel navigieren können. Über einen Server sollen auch die Positionsdaten anderer Teilnehmer abgefragt und dargestellt werden können
- Lösen Sie die Aufgabe mit Hilfe der bereitgestellten kinects, dem Laserscanner Z+F Imager sowie dem bereitgestellten Grundrissplan der Hochschule München.
- Zur Verfügung steht Ihnen hierzu **Matlab unter Windows**.
- Die Kommunikation zwischen einem zu entwickelnden **client** und **server** soll über
  - tcp/ip oder udp erfolgen (falls Kommunikation möglich, ggf. über vpn)
  - Matlab client / online / cloud. Hinweis: Diese Variante ist wohl nur bedingt Echtzeitfähig
- *Hinweis:* Das Projekt ist open end; die Aufgabe muss also nicht vollständig gelöst werden!

## 2.2 ZU ERFÜLLENDE TASKS

Zu erfüllende Tasks (Open End, d.h. es müssen nicht alle Tasks erfüllt werden! Mit der Bitte um Rücksprache, falls die Tasks nicht der Reihenfolge nach angegangen und gelöst werden). Im ersten Teil der Lehrveranstaltung soll ein Partikelfilter basierter Lokalisationsalgorithmus in der zur Verfügung stehenden Simulationsumgebung MobileSim implementiert und entwickelt werden, um diesen dann im zweiten Teil der Lehrveranstaltung in realen einer realen Umgebung (Hochschule München mit einer kinect) zu verifizieren.

1. **Arbeitspaket** („Statusbericht“)
  - ⇒ Fortlaufend (knappe) Dokumentation über aktuellen Projektstand
  - ⇒ ist mit dem Bericht mit abzugeben!
2. **Arbeitspaket** („Literaturrecherche und Zeitplan“)
  - ⇒ Führen Sie eine Literaturrecherche zu dem Thema durch
  - ⇒ geben Sie eine Zusammenfassung der gefundenen Literatur (vgl. „related work“ in Ihrem Bericht)
  - ⇒ erstellen Sie einen Zeitplan, was und welche APs Sie bearbeiten (wollen)
3. **Arbeitspaket** („Data Acquisition (i)“)
  - ⇒ Kinect über Matlab ansteuern können
  - ⇒ Profillinie aus Daten extrahieren können (als Eingabe für den Partikelfilter, vgl. später)
  - ⇒ Daten in einem geeigneten Format abspeichern können (vgl. LidarScan in Robotic Operating System Toolbox) und mit einem Zeitstempel versehen!
  - ⇒ Daten in geeigneter Form archivieren können (z.B. als datastore objekt (?) oder mit timetables oder (...))
  - ⇒ Bestandteil eines zu entwickelnden clients

*Hinweis:* Auf Rückfrage könnte das ganze Projekt und bei entsprechender Adaption der Arbeitspakete auch mit einer imu oder einem Hokuyo Laser durchgeführt werden
4. **Arbeitspaket** („Client / Server Entwurf“)
  - ⇒ Überlegen Sie sich einen geeigneten Entwurf und geeignete Schnittstellen, um die folgenden Arbeitspakete in einer Client/Server Applikation abbilden zu können
  - ⇒ Stellen Sie sicher, dass Ihr Server auch mit den Clients der anderen Gruppen kommunizieren kann!
5. **Arbeitspaket** („Map Creation (i)“)
  - ⇒ Aus ascii 3D Punktwolke (siehe Moodle) einen Grundrissplan erzeugen
  - ⇒ Verwenden Sie ein Occupancy Grid (vgl. Matlab Hilfe) um den Grundriss darzustellen. Überlegen Sie sich hierzu eine geeignete Skalierung.

⇒ Bestandteil eines zu entwickelnden servers

6. **Arbeitspaket** („Partikelfilter“)

- ⇒ Entwickeln eines Partikelfilter basierten Lokalisationsalgorithmus unter Verwendung Ihrer vom client übermittelten kinect Messwerte und Ihres (bzw. des zur Verfügung stehenden) Gebäudeplans!
- ⇒ Visualisieren Sie Ihre Schritte und Ergebnisse geeignet!
- ⇒ Speichern Sie die Ergebnisse geeignet (z.B. mit dem datastore/imageDataStore Object)
- ⇒ Sprechen Sie die Schnittstelle mit den anderen Gruppen ab, so dass auch deren Sensordaten importiert und also mit deren Clients kommuniziert werden kann
- ⇒ Bestandteil eines zu entwickelnden servers

7. **Arbeitspaket** („Map Creation (ii)“)

- ⇒ Mithilfe eines TLS (Z+F Imager 2010) und der Software des Herstellers sowie Ihren Methoden aus Arbeitspaket („map generation (i)“) selbst einen Grundriss erzeugen
- ⇒ In diesem Arbeitspaket sollen die Scans vorab mit der Software des Herstellers registriert und dann als Ascii Punktwolke exportiert werden
- ⇒ Bestandteil eines zu entwickelnden servers

8. **Arbeitspaket** („Map Creation (iii)“)

- ⇒ Mithilfe eines zur Verfügung gestellten Hokuyo Scanners und den in Matlab verfügbaren Methoden zu LidarSlam selbst einen Grundriss erzeugen
- ⇒ Bestandteil eines zu entwickelnden servers

9. **Arbeitspaket** („Routenplanung (i)“)

- ⇒ Verwenden Sie die Matlab Klasse graph, indem Sie geeignete Knoten und Kanten vergeben und so einen Graph des Gebäudes zu erstellen.
- ⇒ Bestandteil eines zu entwickelnden Servers

10. **Arbeitspaket** („Routenplanung (ii)“)

- ⇒ Verwenden Sie die Matlab Funktion shortestpath, um den kürzesten Weg von Ihrem Standpunkt zu einem Zielpunkt zu finden
- ⇒ Visualisieren Sie das Ergebnis geeignet
- ⇒ Bestandteil eines zu entwickelnden Servers.

11. **Arbeitspaket** („Visualisierung, Movie und GUI“)

- ⇒ Runden Sie Ihr Projekt ab, indem Sie eine Matlab GUI erstellen und Ihre Ergebnisse auf dem Client und Server geeignet visualisieren
- ⇒ Erstellen Sie einen movie mit dem Matlab videoWriter Object

12. **Arbeitspaket** („exe erstellen“)

- ⇒ Verwenden Sie matlab compiler, um eine Applikation in Form einer exe oder einer Web App zu erstellen!

13. **Arbeitspaket** („Test“)

- ⇒ Testen Sie Ihre Ergebnisse

## 2.3 LITERATUR

- <https://de.mathworks.com/help/>
- Corke P. (2011): Robotics, Vision and Control, Springer
- Siegwart, Nourbakhsh, Scaramuzzi (2011), Autonomous Mobile Robots, The MIT Press
- T. Abmayr, Multisensor Navigation, unveröffentlichtes Skript zur Vorlesung, Hochschule München, Fakultät für Geoinformation, 2017

- Trucco E., Verri A. (1998); Introductory Techniques for 3-D Computer Vision. Prentice Hall
- Internet!
- *Weitere Literatur wird in der Lehrveranstaltung bekannt gegeben*

## 2.4 MATERIALIEN

### 2.4.1 SOFTWARE UND MATERIALIEN (DOWNLOAD ÜBER MOODLE)

- PointCloud\_230\_Treppenhaus\_WS2017.txt
  - Scans mit Z+F Imager 5010 aufgenommene und im Z+F LaserControl registrierte
  - und als Punktwolke exportiert
- Grid\_230\_Treppenhaus.png
  - Als Rastergrafik abgespeicherte Daten des Datensatzes PointCloud\_230\_Treppenhaus\_WS2017.txt zur Weiterverarbeitung
- GridBinary.png
  - Aufbereiteter floorplan des Datensatzes PointCloud\_230\_Treppenhaus\_WS2017.txt/ Grid\_230\_Treppenhaus.png zum Import für ein OccupancyGrid
- Grundrissplan 1. Stock HM
  - SS2017\_BA\_Stuber\_SchnitteGang\_Betr\_Czaja.zip
- Mind. ein PC der Fakultät (Labor für Navigation )

## Kap. 3 ANHANG

### 3.1.1 ZU AP5 (MAP CREATION (III))

Ziel: Vermessung eines Gebäudeteils mit terrestrischem Laserscanner, um daraus Ground Truth zu erstellen;  
Gehen Sie in folgenden Schritten vor:

- Targets anbringen
- Datenaufnahme mit Z + F Imager 5010X
- Registrierung der Scans mit Z+F LaserControl ("Targetbasierte Registrierung")
  - Hinweise: Markieren Sie hierzu in den aufgenommenen Scans die Targets. Gehen Sie hierzu wie folgt vor:
    - Starten Sie das Programm Z+F LaserControl
    - Öffnen Sie das Projekt:
      - Im Z+F LaserControl linkes Fenster ("Projektexplorer")
      - Projekt öffnen mit linker Maustaste
    - Für jeden Scan
      - Öffnen Sie einen Scans durch Doppelclick im Projektexplorer
      - Markieren Sie nacheinander die Targets (Targettyp DIN A4) und vergeben Sie die Nummer
- Datenexport aus Z+F LaserControl als txt mit Cut Tool
- Datenimport (mit load) und Visualisierung (mit pcshow) in Matlab
- Transformation in ein OccupancyGrid /vgl. Robotic Operating System Toolbox von Matlab ==> <https://de.mathworks.com/help/robotics/ref/robotics.binaryoccupancygrid-class.html>
- Literatur: Registrierung 3D/3D (vgl. Moodle)