Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

MIN Faculty
Department of Informatics

Trixi the Librarian

Masters Project Intelligent Robotics

Shang-Ching Liu, Björn Sygo, Mykhailo Koshil, Fabian Wieczorek

University of Hamburg
Faculty of Mathematics, Informatics and Natural Sciences
Department of Informatics

Technical Aspects of Multimodal Systems

20. October 2022

Outline

Motivation for this master's project

Main specification that lead for creation this project:

- Develop a robotics system with a real life application
- Based on the equipment available at TAMS group
- Preferably including the PR-2 and the Shadow Hand

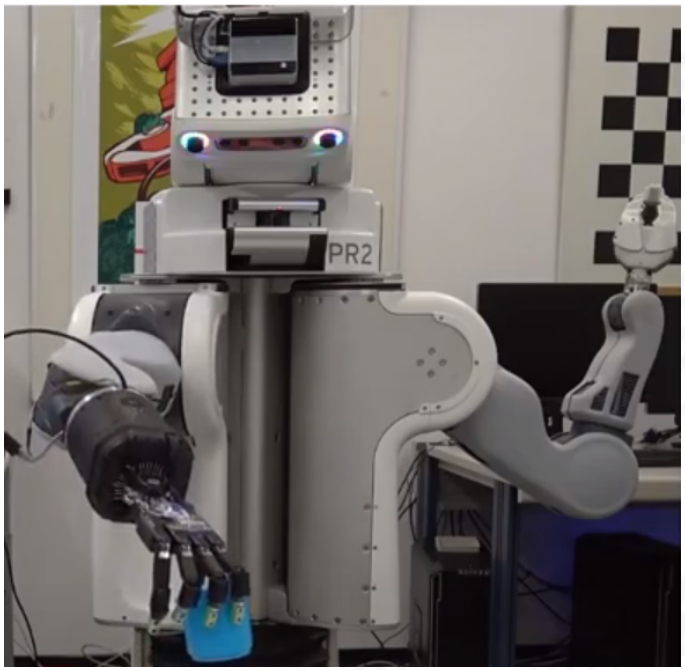


Figure: PR-2 with Shadow Hand

Goal for this master's project

- The ideal scenario: the PR-2 can perform routine tasks of the librarian like fetch/return, sort the books, etc.
- Achieved until now: sort the books on the shelf

Mainly we were inspired by a promo filmed for the TAMS featuring PR-2 in a role of the librarian.

Figure: PR-2 from the RACE project promo
`https://www.youtube.com/watch?v=nqRjo1cPBRY&t=284s`

System overview

Our system enables interaction and perception using three modules:

- Vision
- Task planning
- Manipulation

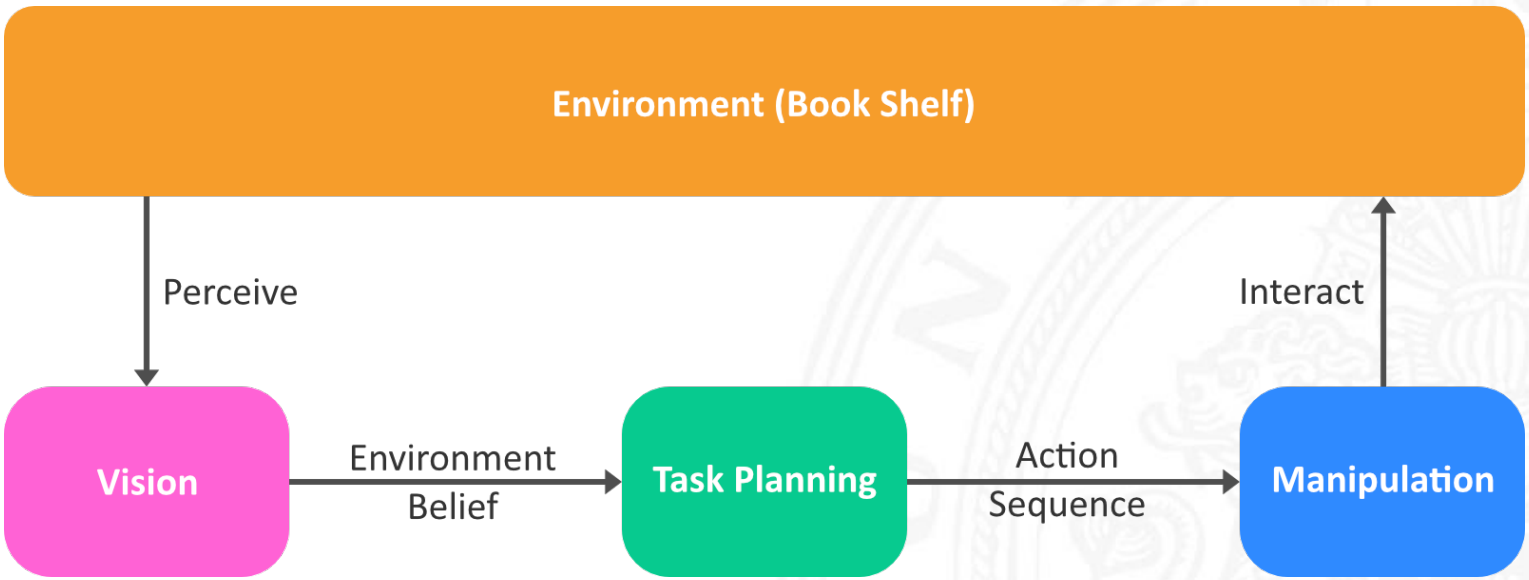


Figure: System Overview

Vision pipeline tasks

Tasks of the vision: process and fuse the data about the environment

- book recognition and localization (complex pipeline developed by us)
- shelf detection and localization (using the fiducial marker)

Use inferred information about the **pose and the title of the books** present in the scene and estimated **pose of the shelf**, vision module **creates environment representation**.

Manipulation pipeline tasks

Tasks of the manipulation: interact with physical books.
Following commands can be executed:

- extract selected book from the shelf
- place the book next to the shelf wall
- inspect the book to increase the recognition confidence

These commands are enough to sort the books on the shelf.

Task planning tasks

Tasks of the task planning: based on the information regarding desired behaviour come up with the sequence of commands for the manipulation module.

- process the environment state
- create sequence of the commands for the manipulation module
- allow for the communication between the manipulation and vision modules (e.g. stop perceiving when the motions are executed, etc.)

Vision pipeline implementation

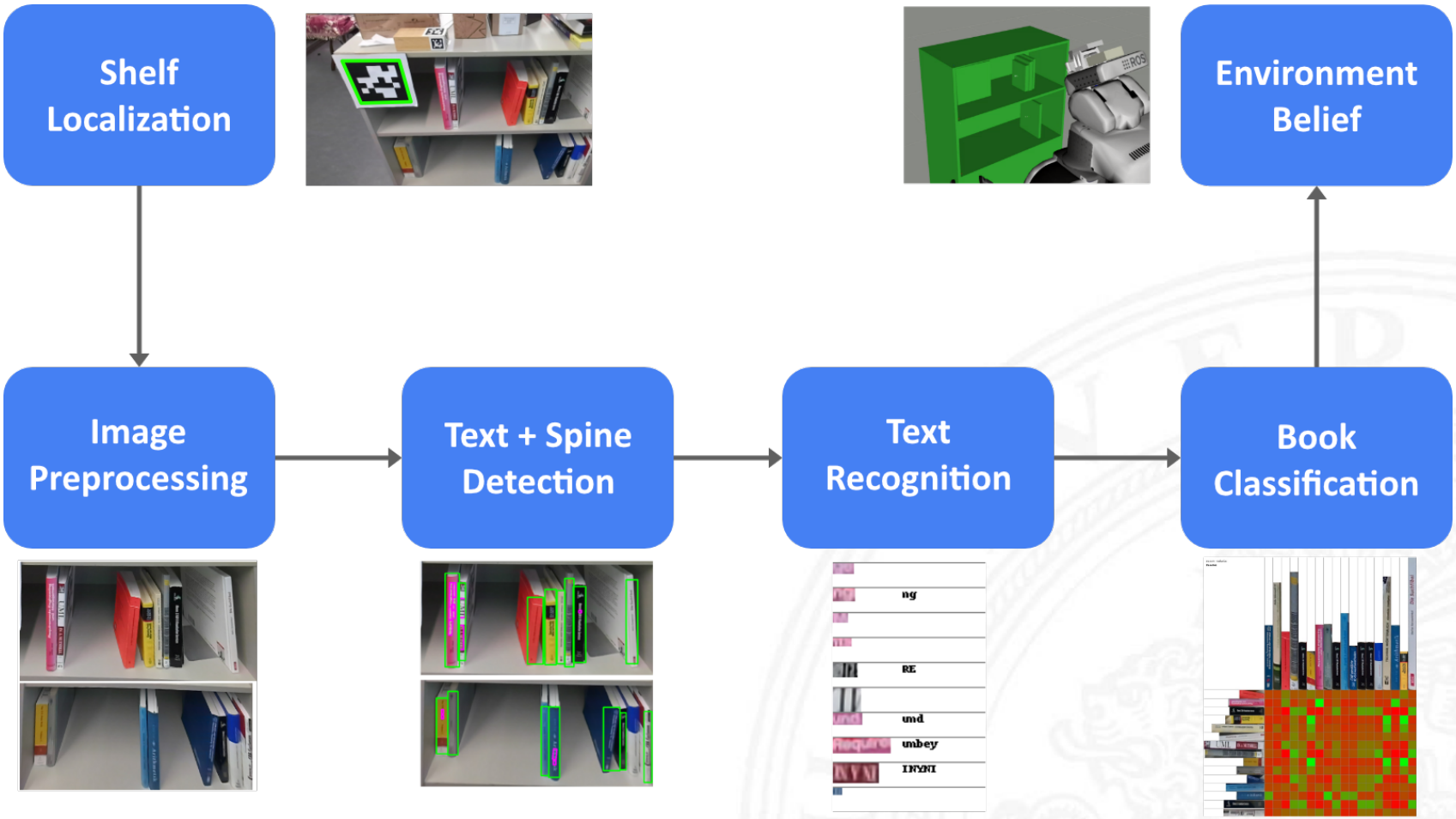


Figure: Overview of the vision pipeline showing the individual processing steps.

1. Shelf Localization

- Shelf Localization using AprilTags
- Problem: Azure Kinect has up to 12MP resolution
  - Slow processing during development
- Solution: Scaled down image + CameraInfo

Figure: AprilTag detection is used to localize the shelf.

- Problem: Book edges not aligned with image axes
- Solution: Perpspective Transformation for each shelf level
- Anchors determined by projecting hard-coded 3D coordinate

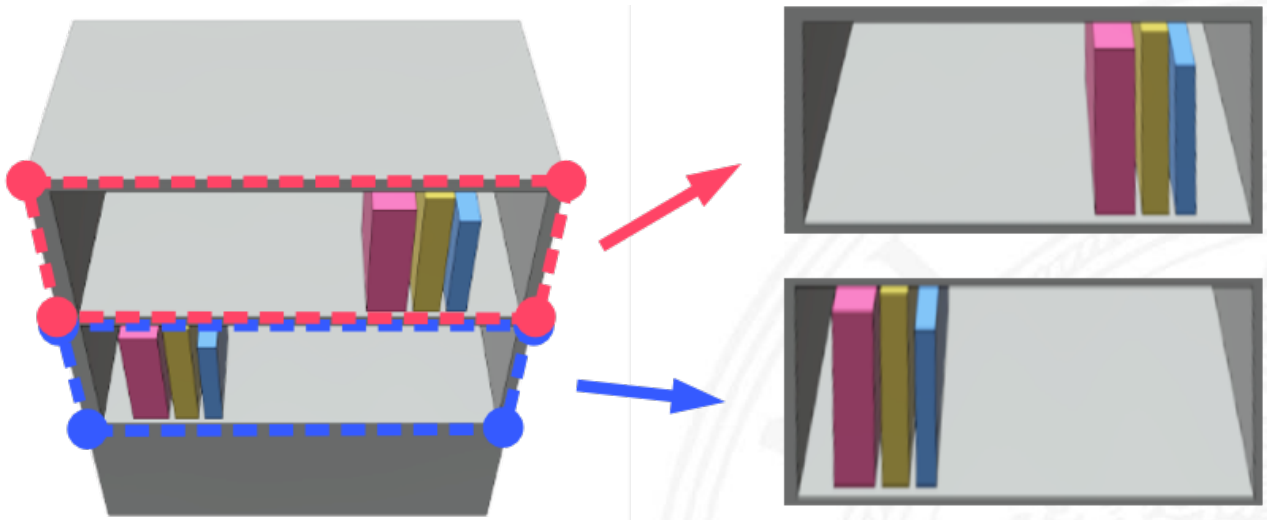


Figure: Perspective transformation is applied twice.

3. Detection - Book Spines

- YOLOv5[1] is used to detect book spines
- Fine-tuned on custom dataset
- 611 labeled images
- Only one class present

Figure: The detected book spines (green) and text (magenta).

[1] https://github.com/ultralytics/yolov5

3. Detection - Text

- CRAFT[2] is used to detect text
- Necessary for later recognition
- Lots of text not detected

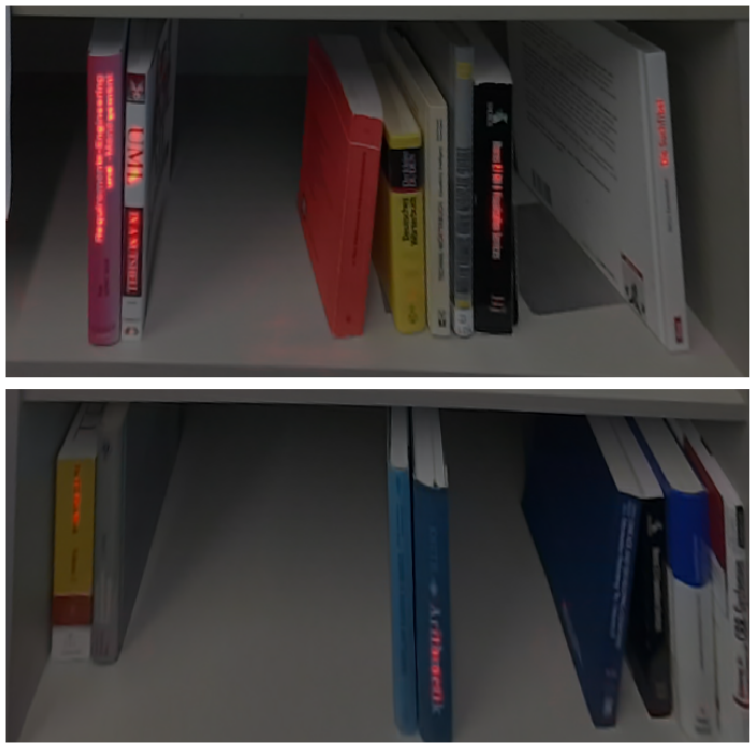


Figure: The detected text.

[2] https://github.com/clovaai/CRAFT-pytorch

Vision pipeline implementation

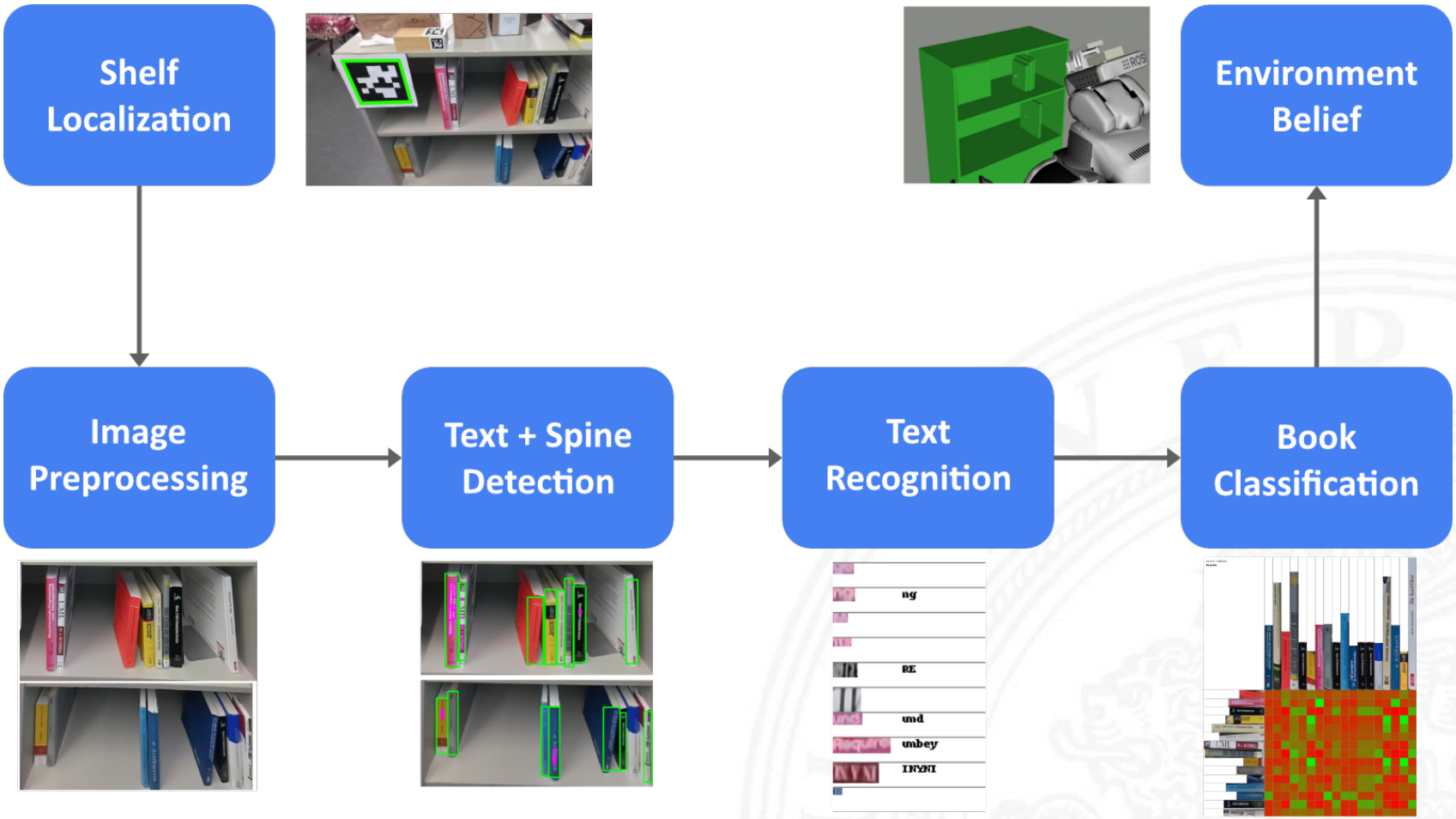


Figure: Overview of the vision pipeline showing the individual processing steps.

- Model from Deep text recognition benchmark[3] is used
- Also tried Google cloud vision[4] text recognition
- No good results from neither because of poor image quality (presumably)

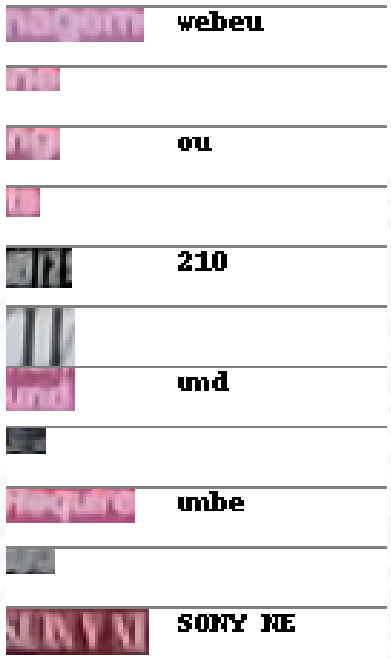


Figure: Results of DTRB text recognition.

[3] https://github.com/clovaai/deep-text-recognition-benchmark
[4] https://cloud.google.com/vision/docs/ocr

5. Book Classification

- SIFT Matching on grey-scale images
- Book dimension similarity
  - Both showed poor performance
- HSV histogram similarity
  - Sufficient for the demo but suffers from different light conditions (presumably)
- Classifier considers choosing each book only once

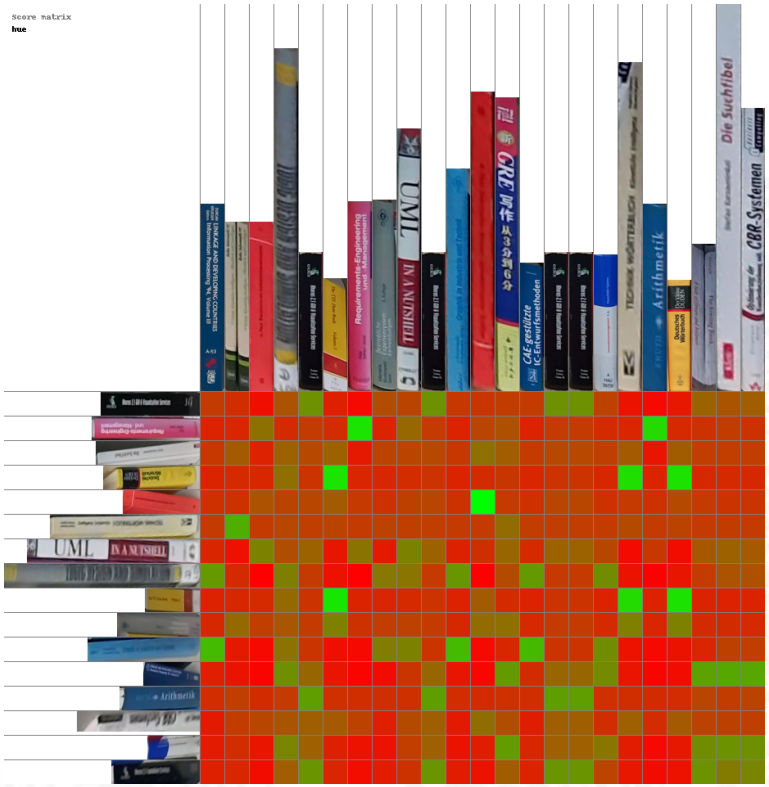


Figure: Similarity between book detections (rows) and books from database (columns). Green means high similarity.

6. Environment Belief Creation

- 10 Observations are aggregated to reduce noise
- Books are clustered by position using k-Means
  - too small cluster are pruned
- Book attributes are averaged for each cluster
- Book position is corrected to make them stand in shelf

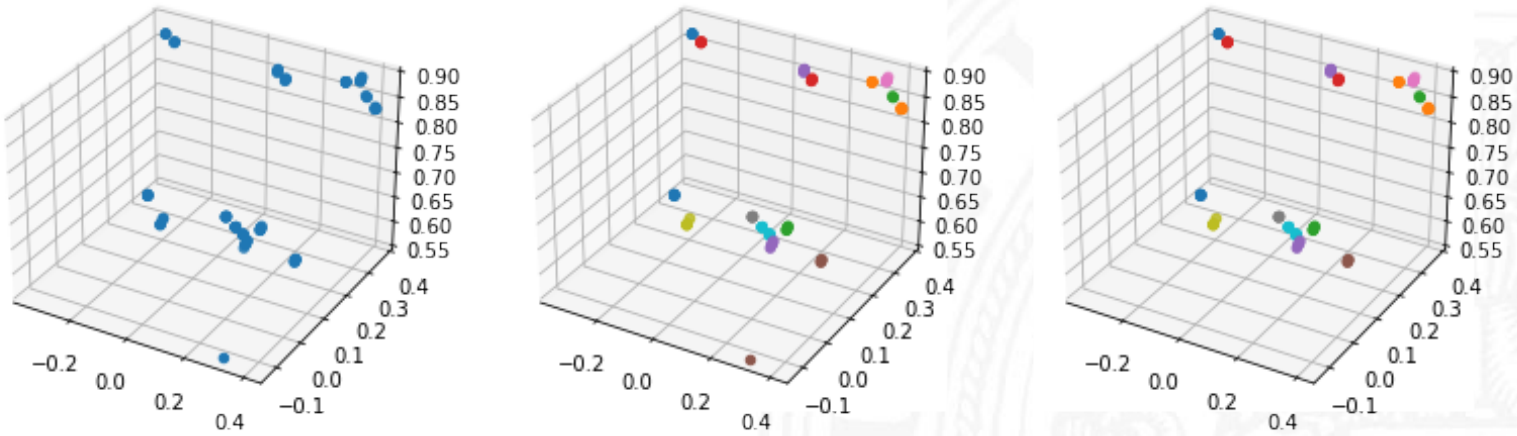


Figure: K-Means clustering over accumulation of observations.

Task planning implementation

The backbone of the communicating between modules is ROS [5]

- the task planning module communicates with vision and manipulation part by custom defined messages with the environment and using a service interface with the manipulation
- descriptive model of the world, therefore outcome of the actions is considered determined
- currently no fallback recovery
- use world state from vision module to procedurally create sequence of commands to move books on the lower shelf in the correct order
- send commands out to the manipulation module

[7] `https://www.ros.org`

Manipulation pipeline implementation

The manipulation is implemented using a MoveIt[6] frame work in conjunction with the BioIK[7] solver in order to facilitate OMPL planner and performed on a bimanual robot equipped with a Shadow Hand.

- Use pose and OMPL for easy path finding
- Get a goal joint configuration from the BioIK given the goal pose, then use the OMPL planner → higher success rate for more complicated paths

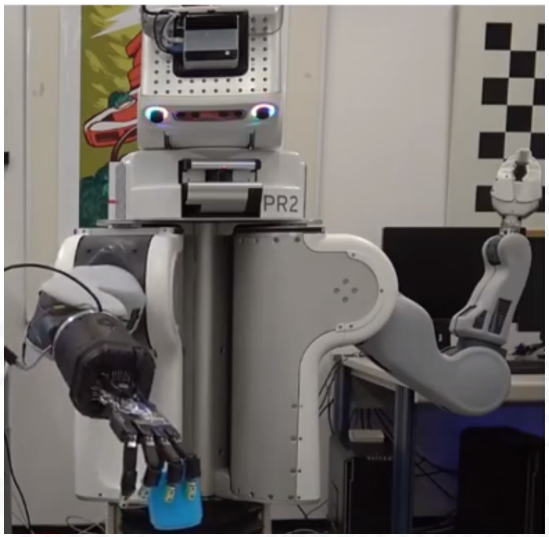
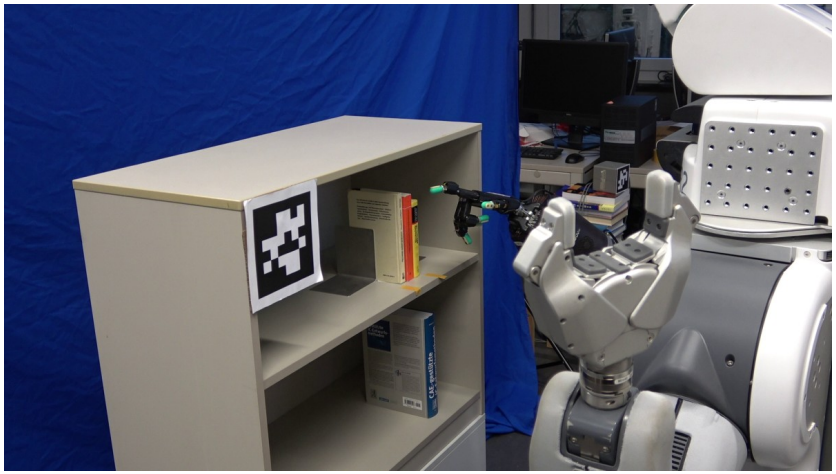
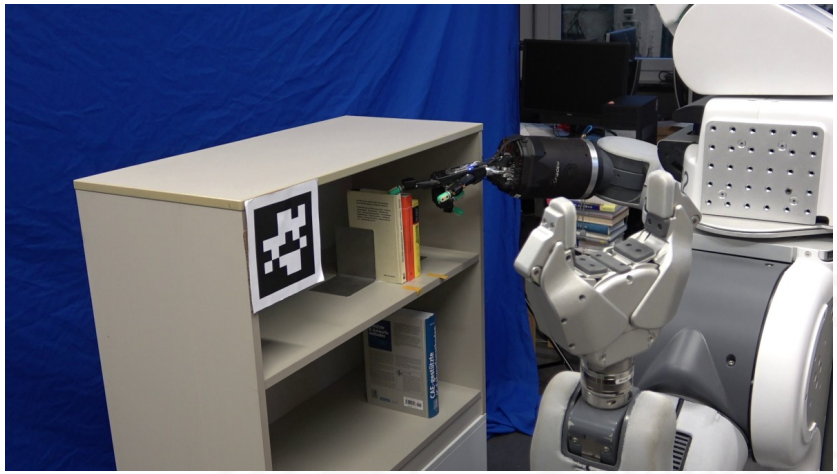


Figure: PR-2 with Shadow Hand

[5] https://github.com/ros-planning/moveit
[6] https://github.com/TAMS-Group/bio_ik

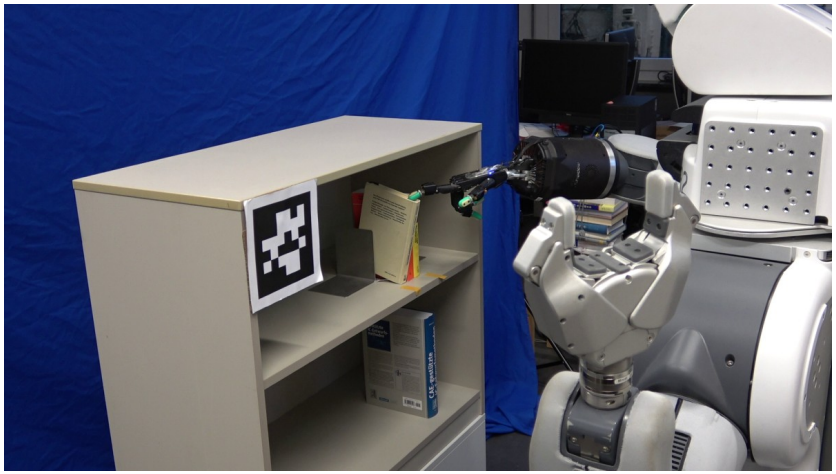


(a) Moving before the book to prepare tilting.

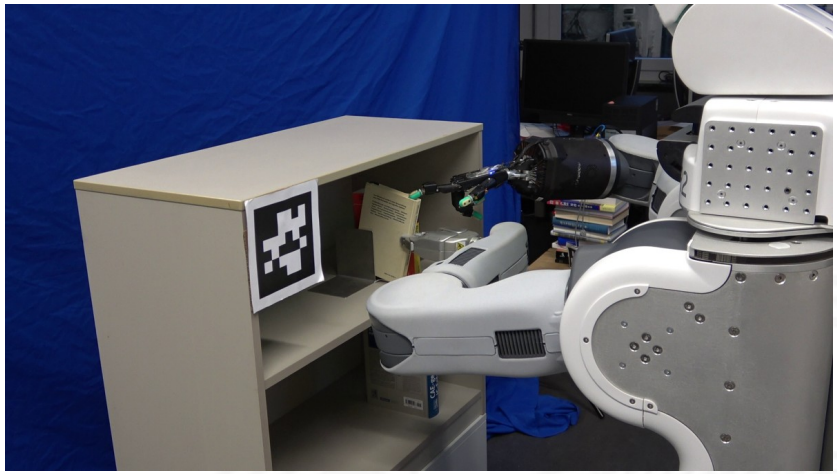


(b) Moving the finger on the book to make contact for tilting.

Figure: The process of extracting the book.

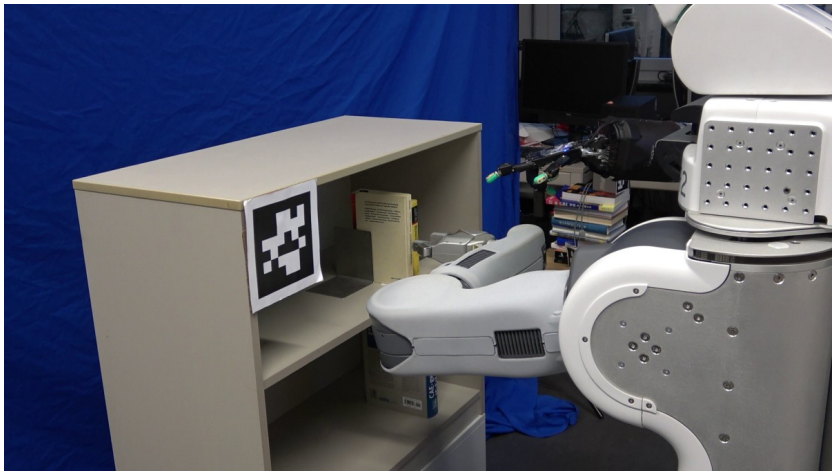


(c) Pulling back the shadow hand to tilt the book.

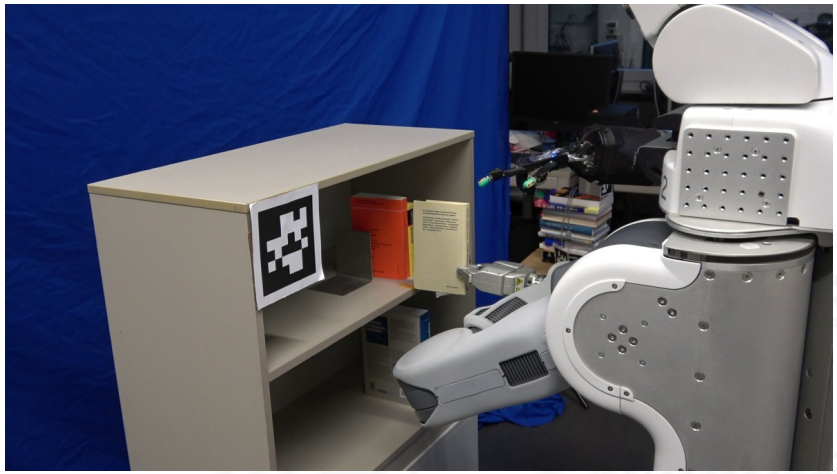


(d) Use the gripper to grasp the book while it is tilted by the shadow hand.

Figure: The process of extracting the book.

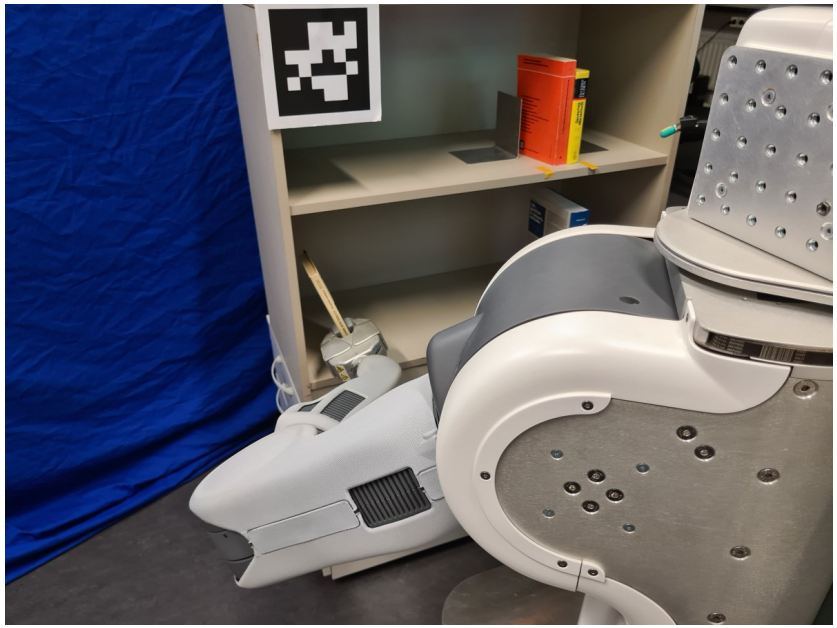


(e) Gripper after pulling the book out a bit and releasing it.

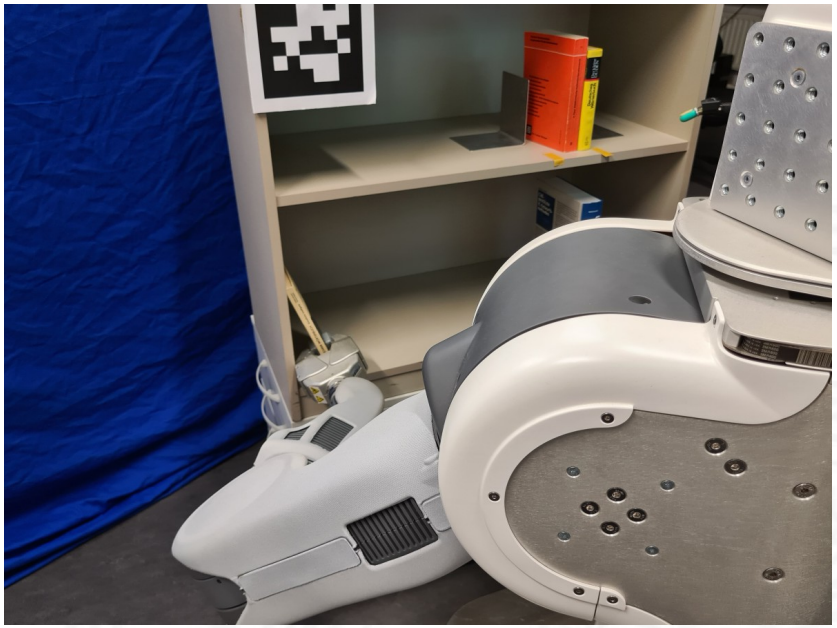


(f) Pulled out book after the gripper has grasped it again.

Figure: The process of extracting the book.



(a) Moving the book into its initial position in the shelf before placing.



(b) Leaning the book against the shelf to the left.

Figure: The process of placing the book.

2. Placing

(c) Positioning next to the book to prepare pushing the book.

(d) Pushing the book against the shelf to let it stand upright.

Figure: The process of placing the book.

Results

Results can be seen in the demo.

For the future work, each of the pipelines have can be improved in terms of the performance, or functionality:

- more robust perception
- other motions for the manipulation to enable more complicated behavior (e.g. error correction)
- more sophisticated task planning system (incorporation of the recovery from a fallback, new scenarios, tighter integration with manipulation etc.)

Results

Thank you for your attention!
Questions?