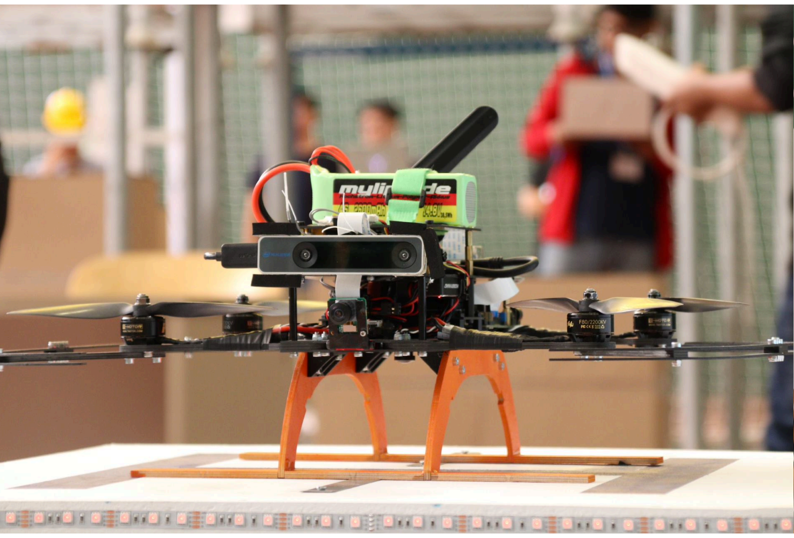
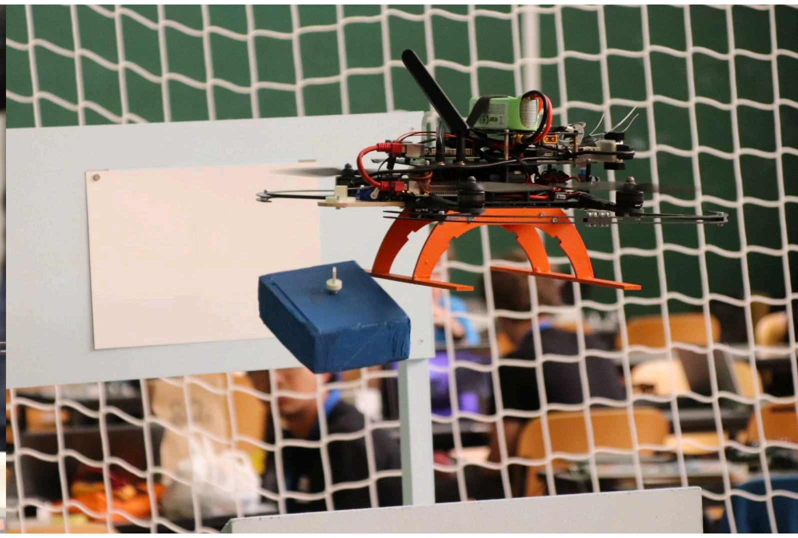
IMAV 2019 Technical Report

Introduction

With the help of our partners and sponsors it was possible for 7 members of the Aachen Drone Development Initiative (ADDI) to participate in the 2019 International Micro Aerial Vehicle (IMAV) Conference and Competition. IMAV was held at the Universidad Politécnica de Madrid in Spain from Sept. 29th to Oct. 5th 2019. Goal of the competition was to develop an autonomous UAV for a warehouse environment. Scored tasks included taking an inventory of the warehouse by scanning shelves and parcels as well as picking up packages and delivering them to different drop off spots.

ADDI participated in the IMAV competition for the first time and placed 3rd overall against 12 teams from 4 continents. Furthermore a special award for the best package handling was given to the ADDI team by the committee.





Hardware Development - Raupe Nimmersatt

A compact quadcopter was developed from ground by our mechanical and electrical engineers. Subsequently two almost identical versions were built to perform tasks in parallel by having both drones fly at the same time at the competition. This also created redundancy in case of software or hardware defects.

The focus was put on incorporating all the required components for full onboard computation of the computer vision tasks. Another design factor was the ability to carry parcels to target locations and drop them. The main material used for the structural components was carbon fibre plates with the undercarriage being constructed of lightweight wood. All parts were custom designed and CNC milled. Mounts for the cameras, the undercarriage and the package drop were designed and 3D printed by our team.

Hardware components of the Raupe Nimmersatt I and II drones	
Flight system	
Motors (4x)	T-Motor F80 Pro with Gemfan 7038 Props
ESCs (4x)	HGLRC 50A BLHELI_32
Flight Controller	Pixhawk FMUv4/FMUv5
BEC	PM06 Power Module
Battery (2x)	2600mAh LIPO (mylipo.de)
Navigation sensors and visual system	
Visual Odometry	Intel Realsense T265
Aruco Tracking	Raspberry Pi Camera V2.1 with 160° FOV lens
Front-facing visuals	Raspberry Pi Camera V2.1 with 160° FOV lens
Actuators	
Package release	9g model building servo motor
Computational hardware	
Main (navigation and control)	nVidia Jetson Nano Development Board
Support	Raspberry Pi 4B w/ 4GB RAM

Software Development

The software stack of the Raupe Nimmersatt drones was built upon **ROS** (Robot Operating System) the most common platform for developing distributed robotics applications. Additionally several packages like **MAVROS** were used for the incorporation of the flight hardware into the navigation and control system.

Early on in the design phase it was determined that the computational power of one onboard computer would not be enough to fulfil all tasks and subsequently the decision was made to split the load between a **NVIDIA Jetson Nano** and a **Raspberry Pi 4**. The **Jetson Nano** being the main device also ran the roscore while the **Raspberry Pi** communicated with the **Jetson Nano** via Ethernet. To control the flight hardware, the **pixhawk flight controller** was connected to the **Jetson** via standard USB.

For rapid testing of our software stack and **ROS** nodes the whole environment was modelled for use in a **Gazebo** simulation.

The **T265** as well as a **picam** were connected to the **Jetson**. To verify the position data provided by the **T265** and to compensate for drift, aruco markers were used as fixed reference points in the world. For marker detection a wide FOV birds-eye view of the floor of the environment was used. This way a global position in the environment could always be determined and verified with good accuracy. Two nodes to fuse this data were developed by the team (Mapper and Fusion).

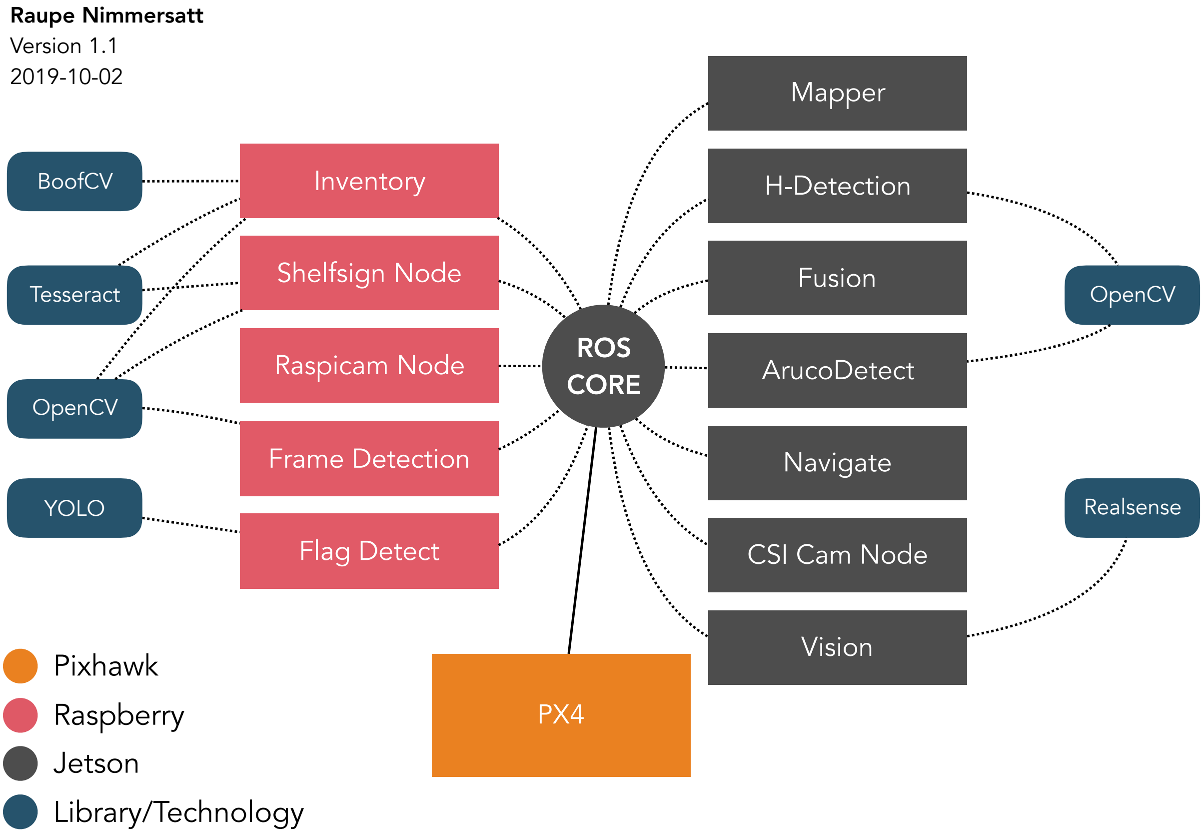


Diagram showing the nodes developed and used by the team, the target hardware of the nodes, interplay of the nodes and the libraries/technologies in use.

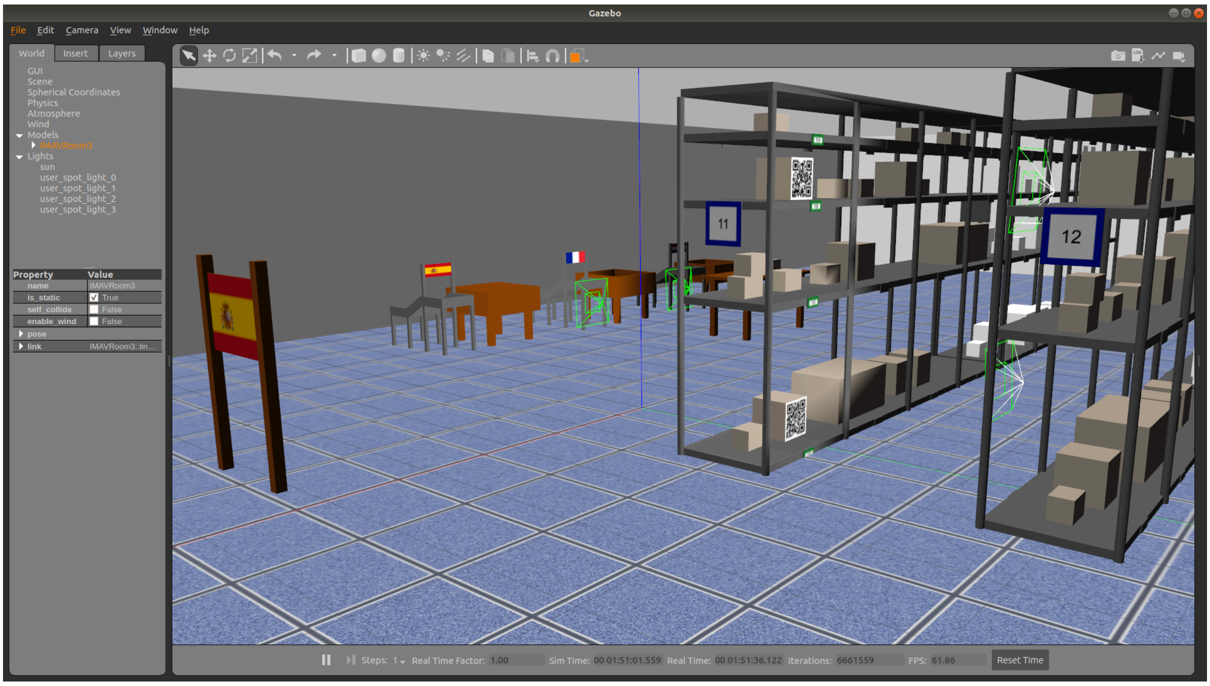
Software for Mission-Tasks

Inventory

The inventory was completely computed on the **Raspberry Pi** by analysing the video feed of the front camera. Packages were localised by scanning QR Codes using **BoofCV** on them and matching them with nearby alphanumerical signs. The signs were detected using edge hierarchies and their alphanumeric code. To read these codes, **Tesseract** was used to decode the text on the signs.

Flag Detection

To drop off the package at the right spot, a flag of a participating country had to be detected in the take off area of the competition environment. At the drop-off location three different spots with different flags were provided and the right one had to be determined.
Training data for this task was generated by our team using the **MIT Indoor Scene Recognition** dataset to which flags were added in random positions. Subsequently a tiny **YOLO** network was trained on the **RWTH** Cluster.



A screenshot of the simulated environment in Gazebo used by the ADDI IMAV team for testing purposes. The scene includes the warehouse shelves, drop-off locations and the flag at the take-off area.