1 Navigation system for autonomous forklift storage system

The navigation system for the autonomous forklift should fulfill the following requirements:

Based upon these requirements the solutions which are marked **yellow** are the most promising. Entries marked **blue** represent ideas that can extend/aid a solution.

Different colored lines on the floor hereby represents the most easy to implement solution for a reasonable price and **Sensor fusion (IMU + GPS)** represents a more sophisticated but more flexible solution with the higher learning outcome.

Approaches:	**Advantages**	**Disadvantages**	**Hardware-Requirements + price range**
Different colored lines on the floor	• Different colored lines allow easy target management • "Easy" algorithms for car control + obstacle avoidance • Cheap price • Customer can lay out a custom pattern • Easy to debug and test	• Calibration of color sensor – shifts with different light levels (especially with respect to TEKExpo with changing lights) • Not that challenging – the solution has been seen/coded before – still relevant learning experience as it is a valid solution • Not flexible – long-term	Color sensor: 80 – 140kr

		changes can only be made by laying different tracks	
Rough GPS location + pallet detection Finding rough area in which the pallet must be and then switch over to pallet detection algorithm	• Simple implementation	• GPS alone is not precise enough, especially indoors (meter precision, but cm is desired) • No precise route planning possible – which is however necessary in warehouses – forklift should not run into other storage shelfs	GPS:59-200kr
Preprogrammed map		• Reliance on some form of feedback from the system – just part of another solution	
Bluetooth RTLS	• Depending on Bluetooth version high precision is possible	• Deployment of beacons – need to be positioned + other groups might use similar technologies, which	Beacons + reciever

		could interfere • Limited range	
Sensor fusion (IMU + GPS) "**Sensor fusion** is the process of combining sensor data or data derived from disparate sources such that the resulting information has less uncertainty than would be possible when these sources were used individually." - Sensor fusion - Wikipedia **https://se.mathworks.com/help/fusion/inertial-sensor-fusion.html** Navigation Kalman Filter with Accelerometer, Gyroscope and GPS - YouTube https://se.mathworks.com/help/nav/ug/imu-and-gps-fusion-for-inertial-navigation.html Possibly using MATLAB codegen Or MATLAB Embedded Coder for sensor fusion: Understanding Sensor Fusion and Tracking, Part 2: Fusing a Mag, Accel, & Gyro Estimate - YouTube Understanding Sensor Fusion and Tracking, Part 3: Fusing a GPS and IMU to Estimate Pose - YouTube **https://www.youtube.com/watch?v=UZsxFpjmdAs** **Video by MATLAB claiming Embedded compatibility and explaining general concept** Possible also not to use MATLAB: https://www.youtube.com/watch?v=hQUkiC5o0JI	• Highly precise and self-correcting • No range limitations • High learning curve – filters (Kalman-Filter + implementation of advanced mathematical models in C code, possibly using MATLAB) • High flexibility – any point in space can be targeted • Good in combination with programmed map • Suitable for fast applications – including fast drones	• Pricy – navigation system could cost around ca. 150-350kr • More **complex** implementation • Reliance on several sensors • Possible calibration time • Harder to debug (Assumption) • Algorithm by MATLAB might stress the computing power of the selected microcontroller • I could not find examples of previous implementations on an MCU using Matlab generated C code	Gyroscope: 64 to 149 kr Magnetometer: 46-61kr Gyro + Accelerometer: 70kr Gyro, Acc + mag often come combined GPS:59-200kr Required Matlab toolboxes are included in our license
ESP32 UWB	• Possible high precision	• Range limited • Reliance on specific	Around 270kr each – too pricy

		version of ESP32-microcontroller • At least 3 MCUs needed - pricy	