Replacing expensive machine vision systems with low-cost 3D Time of Flight sensors in mobile robotics

Antti Alhonen, Pulu Robotics Oy, 2018

www.pulurobotics.fi

Definitions

Autonomous mobile robot – a robot that can move in its designated operating environment without causing damage to its surroundings (unless specified so), work for extended periods of time (i.e., charge itself)

SLAM – simultaneous localization and mapping: a sensory and computational act of building a map “from scratch”, by moving around, taking measurements, and later localizing itself in this map, without human intervention

Motivation for 3D

Machine vision for mobile robotics, especially autonomous, is of utmost importance. It is also considered one of the most complex and expensive areas.

Safety requirements necessitate that the robot can reliably detect close-by obstacles, such as people and property, in order not to hit them while moving through the operating environment.

Autonomous robots often provide mapping functionality, for example, simultaneous localization and mapping (SLAM). For this, precise distance measurements around the robot are needed in order to build the map with a minimal cumulative error.

Traditionally, mobile robots that perform SLAM utilize a 2D plane LIDAR, consisting of a rotating head, mounted with a laser range finder, typically based on either geometrical rangefinding (beam-sensor triangle) or time-of-flight principle.

While these 2D plane lidars are useful for navigating thin robots (approximating the two-dimensional shape), real robots often are tall. In these cases, 2D plane LIDAR is still often acceptable for mapping simple areas, but never good for safety: avoiding hitting obstacles.

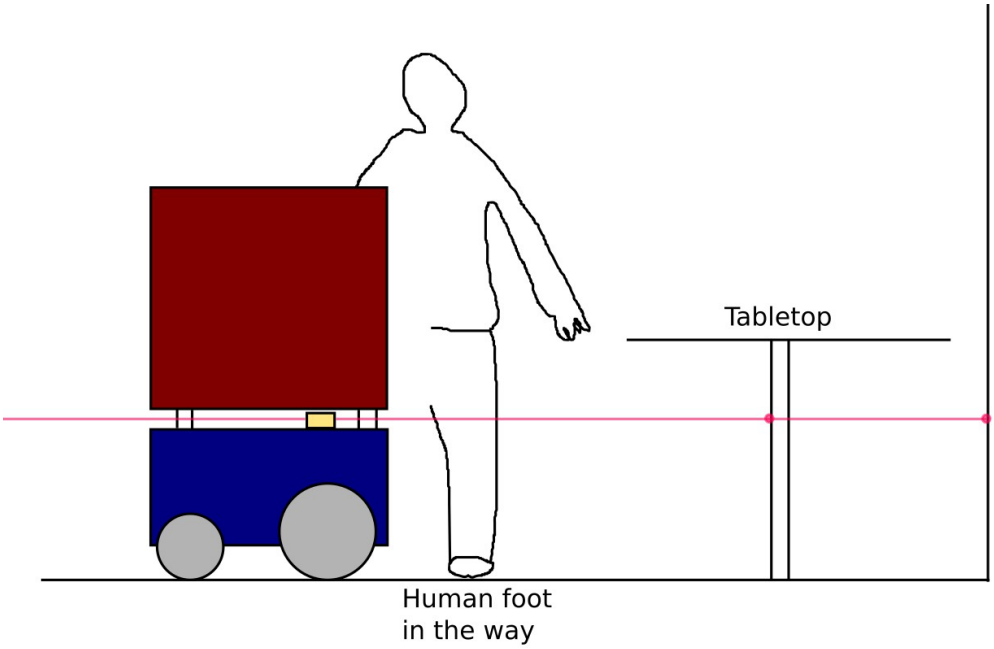

Figure 1. A 2D plane LIDAR cannot see the human foot in its way. The tabletop is also left unseen, so the tall robot will hit it unless equipped with secondary sensors. Only the table feet and the room wall geometry are seen: usable for localization and mapping.

Scanning 3D LIDARs do exist, but they are prohibitively expensive, or scan the environment slowly, so while they are good in mapping limited, static environment, they are no good in real-time obstacle avoidance.

Time-of-Flight distance measurement principle

Measuring distances by utilizing the speed of light (299 792 458 m/s) is not new, although considerably more challenging than utilizing the speed of sound (approx. 343 m/s, i.e., one-millionth of the speed of light).

Traditionally, pulsing light and measuring time to see the “echo” of the light has been impossible or prohibitively expensive. Instead, the light is modulated at a fairly high frequency (typically between

1 and 100 MHz). A receiver creates a slightly delayed signal, and the phase difference of the two is compared.

Many if not most 2D plane LIDARs utilize this technology. A narrow laser beam is sent, and a narrow beam optics detect the small laser dot drawn on the walls. This could be called “1D Time-of-Flight” - the only dimension is the depth. When this sensor is rotated to take multiple measurements, a 2D plane is being “scanned,” each sample at a different time.

3D Time-of-Flight

3D Time-of-Flight (3DTOF) is the logical next step from the 1D Time-of-Flight measurement, following from the advancements of integrated semiconductor technology and the birth of low-cost digital cameras.

3D Time-of-Flight sensor simply packs the distance measurement cells as a 2D array: instead of a single distance, or a 2D plane, it produces a three-dimensional point cloud (limited as a rectangular pyramid shape).

Internally, a 3DTOF sensor is like a digital camera sensor, but each pixel site implements two photon-counting wells instead of one. Incoming light is quickly demodulated (toggled) between these two wells, using the same signal that modulates the light source – typically, between 5 and 50 MHz. A differential amplifier is used to read out the difference in photons, rejecting DC and low-frequency ambient light.

Pros of 3DTOF

Positive aspects of the 3DTOF technology, in general, include:

1. Cheap to manufacture, especially per pixel – not too different from digital camera sensors
2. High resolution – sensors with 320*240 (76800) pixels (measurement points) are available.
3. Incredible measurement rate – typical products can provide over 50 full frames (up to tens of thousands of measurement points each) per second
4. All points are measured simultaneously – minimal motion artifacts
5. No moving parts, robust solid-state solution

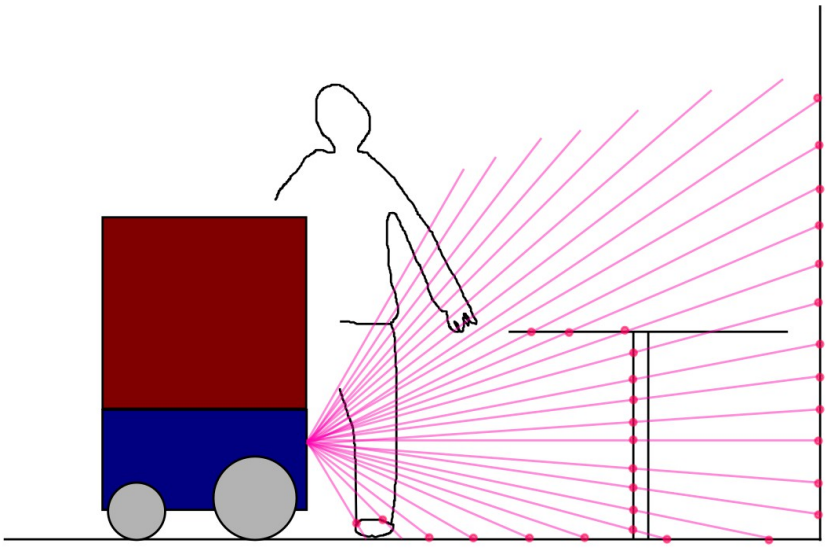

Figure 2. 3DTOF can see all obstacles at one glance – and also analyze the floor levelness.

Cons of 3DTOF

Negative aspects tend to cause a serious disillusion to first-time designers who try out the 3DTOF technology:

1. **Most importantly,** when the simultaneously modulated light pattern is wide, multipath effects can never be eliminated. At the same time, effects of modulated stray light in the lens are hard to minimize. Combined with aliasing (wrap-around), these effects are hard to predict and compensate for. These aspects and our solutions are discussed in more detail below.
2. To a lesser effect, but still significant: Since there are tens of thousands of distance-measuring pixels in a low-cost device, the quality of these pixels, analog-to-digital converter arrays, etc., is not as high as single-point solutions, even with state-of-the-art technology. This requires extra compensation and calibration against nonlinearity, temperature drift, etc.

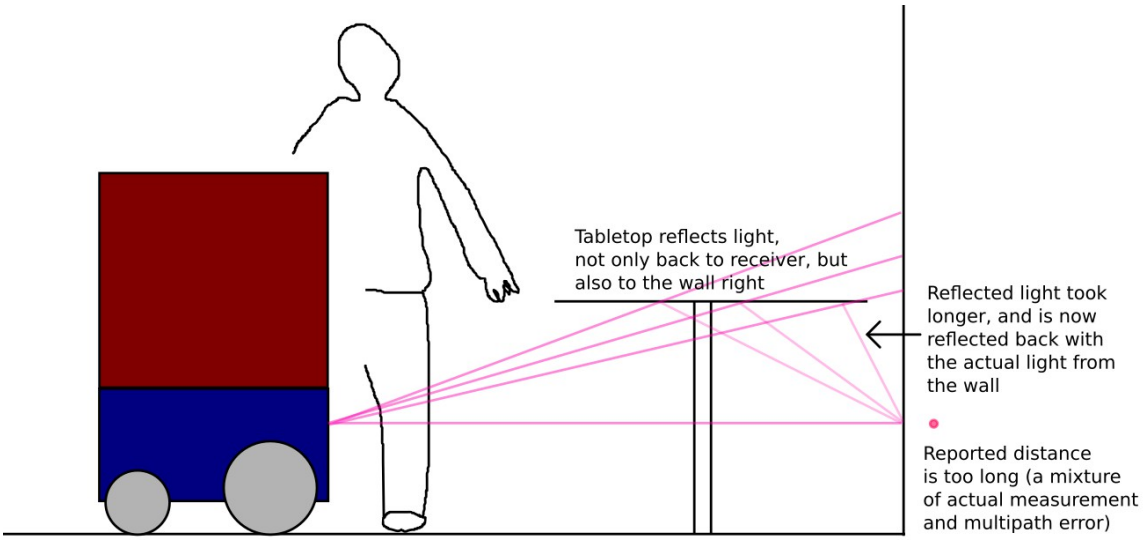

Figure 3. Multipath error in 3DTOF. Since the measurement (neither the modulated light beam nor the sensed area) is not a narrow cone, separate objects (e.g., the tabletop and the wall) are illuminated by the same modulated light, simultaneously. Instead of only reflecting the light back to the sensor, as desired, they also illuminate each other, causing distorted measurement.

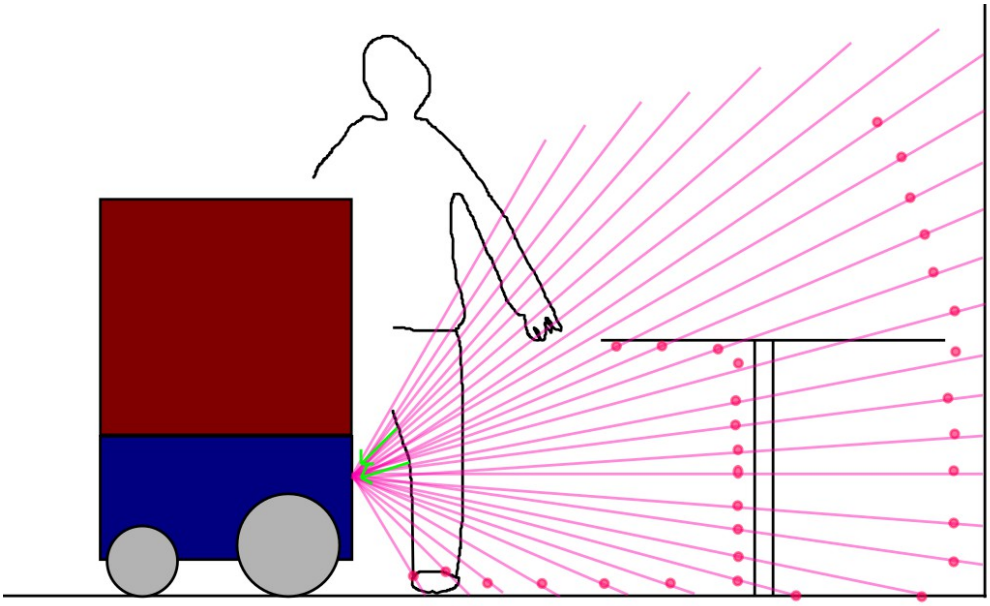

Figure 4. Modulated stray light error in 3DTOF. Highly IR-reflective trousers very close to the modulated illumination source reflect a very strong light, dispersing inside the lens, despite the lens coatings, causing the light to be measured at the pixel sites where it doesn't belong to. This close-range measurement, possibly just centimeters away, mixes with the actual readings at all pixels. Pixels with low-amplitude actual distance suffer more than those getting high signal level.

Why the robot vision problem hasn’t been solved using 3DTOF before?

Commercial 3DTOF camera modules have existed for more than a decade, and they have been used in mobile robots.

However, like all “robot modules”, these products carry significant price overhead:

- Image processing (CPU and/or FPGA) and memory
- USB or Ethernet interfacing electronics
- Power supplies
- Casing, possibly in undesirable shape
- API development effort, general company costs
- Profit their manufacturer needs to make

Additionally, these products are designed to be “general purpose”, which means their designers cannot make specific compromises to bring the cost down, nor they can do specific optimizations to bring performance up.

Generally, low-cost products have been useless (for example, with only 2-meter range, lacking de-aliasing), while better products have been very expensive, yet still suffer from the optical inaccuracies of 3DTOF without much effort put in minimizing them.

Due to the high expense of a single 3DTOF camera, and the fact that most products cannot be synchronized so that they would not interfere with each other, most robots only utilize one camera. This decision further limits the robot shape to be completely round, with its origin in the middle, so that it can turn around without hitting anything, even with limited vision. This decision, however, greatly limits the robot’s usefulness in real usage cases (delivery, for example).

What can we do differently, to enable 3DTOF as the primary (or the only) sensor?

First and foremost, we acquired a license to use sensor chip technology developed by Espros Photonics Corporation, for designing our 3DTOF camera modules from scratch.

Espros was chosen because of the well-specified, simple electrical interface, allowing us to bring the cost down by not using vendor-lock-in processing components, instead of implementing our image processing. Secondarily, Espros provides the best quantum efficiency of any 3DTOF sensor chip providers, meaning less illumination power required.

Some unique solutions were needed, nevertheless:

1. Multiplexing number of cameras

Multiplexing several cameras is essential for several reasons:

1. It is the only way to provide full 360-degree view around the robot, for best localization and mapping results, and maneuvers such as reversing.
2. It brings the unit cost down since most of the functions of a single 3DTOF camera don't need to be replicated; image processing, interfacing electronics, and power supplies can be shared.
3. Mitigating modulated stray light error (and other error sources) by overlapping: Using multiple cameras enables us to see beyond the borders of the image of one sensor: if there is an extremely reflective obstacle right next to the image field, but barely not within the image area, it is going to affect the measurements as explained in Figs. 4 and 5. Having another camera next to it, this object can be fully or mostly captured, then treated properly in a compensation algorithm.

   When the images slightly overlap, differences between the supposedly identical area can be analyzed. As the images are taken from a different angle, close-up obstacles (causing a modulated stray light error) are very likely to be in a completely different position, and only affect one of the images.

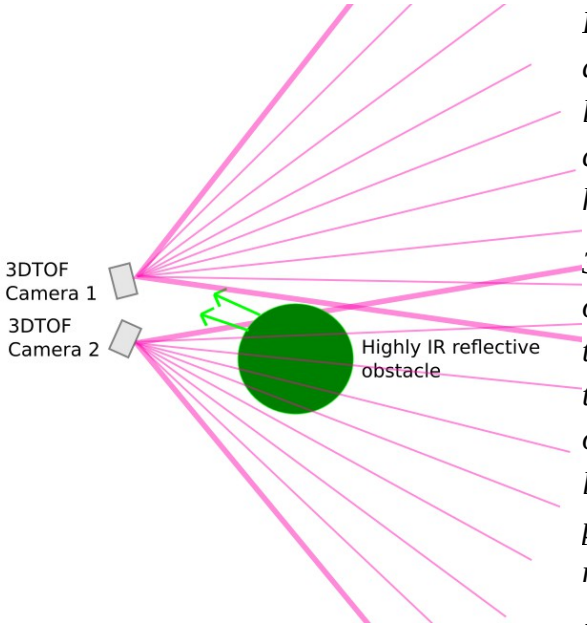

Figure 5. Highly IR-reflective close-by obstacle distorts 3DTOF camera 1, due to modulated stray light in its optics. As it is not seen directly by the camera, it's hard to know if any distortion is actually happening, how much, and why.

3DTOF Camera 2, however, directly sees the full obstacle. Camera 2 can simply shorten its exposure time (or utilize a High Dynamic Range mode) to get the amount of stray light down, then image the obstacle accurately, and model the amount of stray light generated by it. This information can be propagated to the calculation of Camera 1 distance map.

In the simplest implementation, the processing algorithm for Camera 2 could just decide to mark Camera 1 data "inaccurate for SLAM" instead of trying to correct it. For obstacle avoidance, the decision is correct anyway: we can't go further in that direction!

2. Secondary narrow-beam imaging for error compensation

As explained in Figures 3 and 4, serious, hard to predict errors appear due to modulated stray light and multipath effect. Both problems are fundamentally caused by using wide light source pattern, the very same appeal of 3DTOF that enables us to do wide imaging at a high frame rate and low cost.

In addition to the means of compensation explained above (overlapping multiple cameras), we introduce another one.

Adding a more accurate extra sensor would make sense: just add another laser rangefinder in the middle of the image field. By utilizing “traditional” narrow beam of illumination, stray light or multipath do not happen (or are limited to very low levels): the result is an accurate 1D LIDAR.

Adding an extra sensor would increase cost and complexity, however. Luckily, the 3DTOF camera module already has the sensor: we can utilize the same sensor chip and optics, but only read out the middle of the frame. The only thing left is to design an alternative light source, providing the narrow beam, and a way to switch between these two light sources. This is fairly cost-effective, as the secondary beam requires considerably lower power due to the small illuminated area. The complexity increase is very manageable, as well. All this means about $2 BOM increase per camera module.

Accurate midpoint data has to be imaged quickly after (or before) the full image, so it represents the same objects while everything moves in reality. In practice, 10ms interval is well achievable.

Midpoint data can be compared to the full data at the same location. Good narrow-beam data can be used in the SLAM, and the full data around it can be compensated to match in the middle. If it is corrected to match in the middle, and the full frame is corrected similarly, the narrow beam can be extended. The further we get from the middle, the less accurate, and more uncertain the data becomes.

Finally, if there is a considerable difference between the accurate narrow-beam data and the full sensor data, the full data can be only used for live obstacle avoidance and marked invalid for producing accurate maps. Mapping can still work out with less data, just more slowly.

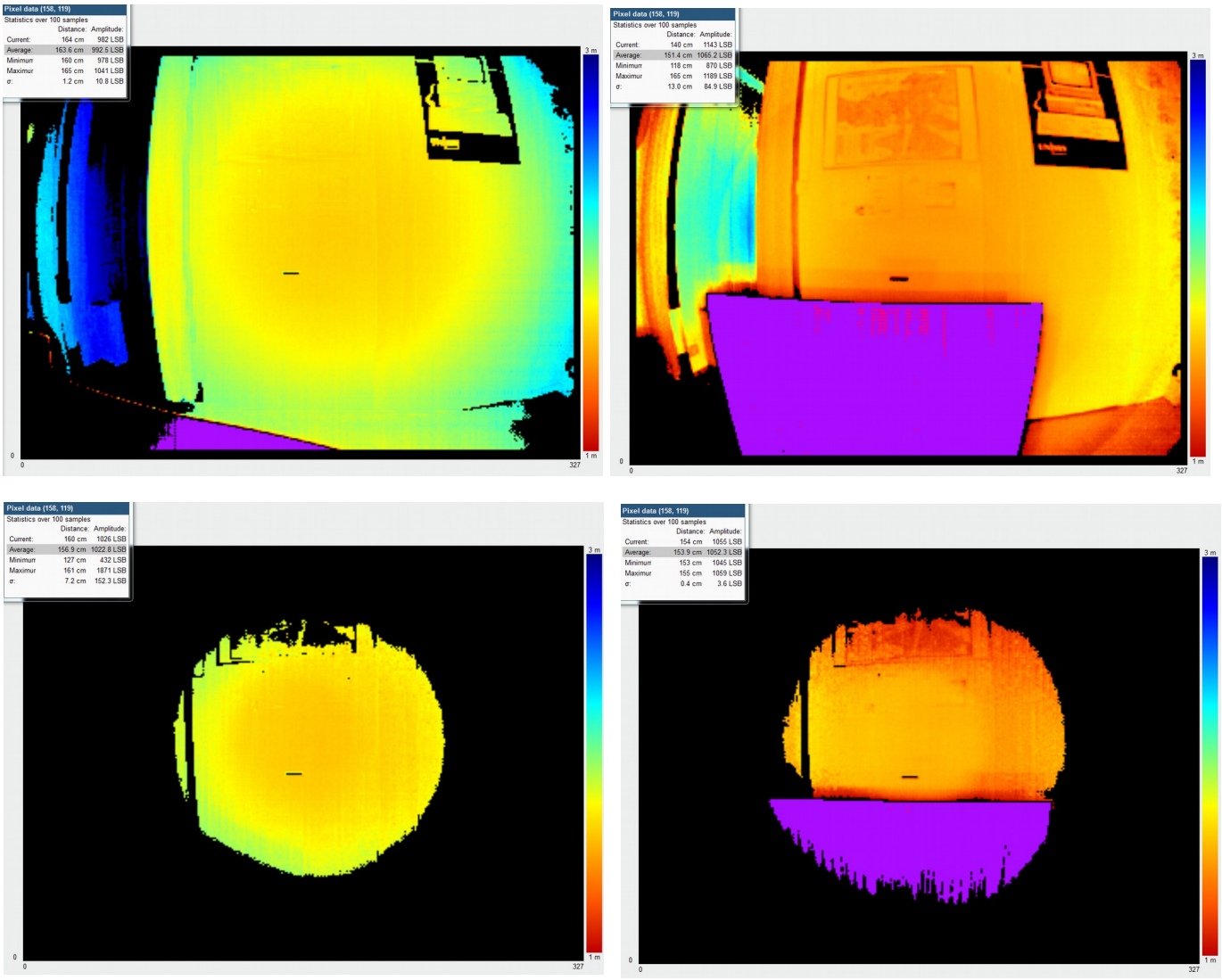

Figure 6. Right after acquiring the 3DTOF development kit, we tested our narrow beam hypothesis. A bright, IR-reflective white cardboard was brought into close proximity while measuring the distance to a marked point in the wall (black tape in the middle of the images). Pink indicates overexposed pixels.

Top row: the development kit LED illumination source is unmodified, providing approx. 90 deg (+/- 45 deg) half-intensity angle of illumination. The bright obstacle causes a wall measurement distortion from 164cm to 140cm, or -14.6% error.

Bottom row: the development kit LEDs are equipped with short black pipes to narrow down the light source cone shape, without affecting the sensor in any other way. The half-intensity angle of illumination is now approx. 30 deg (+/- 15 deg). The bright obstacle now causes a distortion from 159cm to 146cm, or -8.2% error. The error is reduced to 56%.

This experiment made us confident in the effectiveness of using 8 deg (+/-4 deg) half-intensity angle of illumination as our narrow-beam source.

3. Integrating 3DTOF to the rest of the robot platform

In addition to multiplexing several camera modules with one controller, the cost can be further reduced – and the system further simplified, by integrating the 3DTOF processing within the same microcontroller that is already present in the system.

In the case of Pulu Robotics' RobotBoard, this means upgrading from the existing ARM Cortex M3 CPU to an ARM Cortex M7 flagship MCU, only a $3 increase in the bill of materials (BOM). This $3 increase is offset by another $4 decrease, as this more capable microcontroller can take over the task of separate motor control MCUs as well!

Integrating the sensor access and image processing with existing microcontroller offers the following benefits:

1. Cost reduction: only one microcontroller is needed, on one printed circuit board (PCB) only
2. Space reduction: fewer parts, smaller solution
3. Less interfacing, fewer APIs: less specification and implementation work with fewer intermediate communication links – also meaning shorter time-to-market with limited resources
4. Quick response, robustness, reliability, safety: measured distances are immediately available for low-level obstacle avoidance: no need to transfer them first from one module to another.

Conclusion

We claim that a carefully designed and well integrated 3D Time Of Flight sensor array of about five to ten sensors, specifically designed to work together, can replace all or most other sensors for avoiding challenging obstacles such as tabletops, human feet, even hanging cables.

We further claim that such a setup can be used as the only visual source for Simultaneous Localization and Mapping. (SLAM should, naturally, still use other types of sensor data, such as wheel odometry and inertial measurements, which can be implemented very cheaply.)

Utilizing 3DTOF as the only visual source of data for SLAM, however, requires carefully designed compensation methods. We presented two such methods we have never seen discussed before; these methods become apparent only when we look at the 3DTOF as an integrated part of a robot, as a "total SLAM component," instead of a standalone single-camera product.

Exact algorithms for implementing the presented correction principles are worked on. We believe such algorithms are best developed by prototyping on an actual system, driven in real-world environments.

Version history of this document

2018-02-01: Initial release by Antti Alhonen