

## Semester Thesis

# Evaluation of VIO Methods for Wheeled Agricultural Robots

Spring Term 2021



## **Declaration of Originality**

I hereby declare that the written work I have submitted entitled

Evaluation of VIO Methods for Wheeled Agricultural Robots

is original work which I alone have authored and which is written in my own words.<sup>1</sup>

## Author(s)

Shane Kelly

### Committee members(s)

Edo Jelavic  
Johannes Pankert

### **Supervising lecturer**

Marco Hutter

With the signature I declare that I have been informed regarding normal academic citation rules and that I have read and understood the information on ‘Citation etiquette’ (<https://www.ethz.ch/content/dam/ethz/main/education/rechtliches-abschluesse/leistungskontrollen/plagiarism-citationetiquette.pdf>). The citation conventions usual to the discipline in question here have been respected.

The above written work may be tested electronically for plagiarism.

California, USA 13.07.2021

### Place and date

Shane Kelly

Signature

---

<sup>1</sup>Co-authored work: The signatures of all authors are required. Each signature attests to the originality of the entire piece of written work in its final form.

# Intellectual Property Agreement

The student acted under the supervision of Prof. Hutter and contributed to research of his group. Research results of students outside the scope of an employment contract with ETH Zurich belong to the students themselves. The results of the student within the present thesis shall be exploited by ETH Zurich, possibly together with results of other contributors in the same field. To facilitate and to enable a common exploitation of all combined research results, the student hereby assigns his rights to the research results to ETH Zurich. In exchange, the student shall be treated like an employee of ETH Zurich with respect to any income generated due to the research results.

This agreement regulates the rights to the created research results.

## 1. Intellectual Property Rights

1. The student assigns his/her rights to the research results, including inventions and works protected by copyright, but not including his moral rights ("Urheberpersönlichkeitsrechte"), to ETH Zurich. Herewith, he cedes, in particular, all rights for commercial exploitations of research results to ETH Zurich. He is doing this voluntarily and with full awareness, in order to facilitate the commercial exploitation of the created Research Results. The student's moral rights ("Urheberpersönlichkeitsrechte") shall not be affected by this assignment.
2. In exchange, the student will be compensated by ETH Zurich in the case of income through the commercial exploitation of research results. Compensation will be made as if the student was an employee of ETH Zurich and according to the guidelines "Richtlinien für die wirtschaftliche Verwertung von Forschungsergebnissen der ETH Zürich".
3. The student agrees to keep all research results confidential. This obligation to confidentiality shall persist until he or she is informed by ETH Zurich that the intellectual property rights to the research results have been protected through patent applications or other adequate measures or that no protection is sought, but not longer than 12 months after the collaborator has signed this agreement.
4. If a patent application is filed for an invention based on the research results, the student will duly provide all necessary signatures. He/she also agrees to be available whenever his aid is necessary in the course of the patent application process, e.g. to respond to questions of patent examiners or the like.

## 2. Settlement of Disagreements

Should disagreements arise out between the parties, the parties will make an effort to settle them between them in good faith. In case of failure of these agreements, Swiss Law shall be applied and the Courts of Zurich shall have exclusive jurisdiction.

California, USA 13.07.2021

Place and date

  
Signature

# Contents

<b>Abstract</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 VIO Methods</b>	<b>3</b>
2.1 OpenVINS . . . . .	3
2.2 ORB-SLAM3 . . . . .	3
2.3 ROVIO . . . . .	4
2.4 VINS-Fusion . . . . .	4
2.5 Summary . . . . .	5
<b>3 Sensors</b>	<b>7</b>
3.1 Visual-Inertial Sensors . . . . .	7
3.1.1 T265 . . . . .	8
3.1.2 Alphasense . . . . .	8
3.1.3 Summary . . . . .	8
3.2 Groundtruth Sensor . . . . .	9
<b>4 Datasets</b>	<b>11</b>
4.1 Eschikon . . . . .	11
4.2 Rosario . . . . .	12
<b>5 Evaluation Metrics</b>	<b>13</b>
5.1 Accuracy . . . . .	13
5.1.1 Accuracy for Online State Estimation . . . . .	13
5.1.2 Effect of Loop Closures . . . . .	13
5.2 Robustness . . . . .	13
5.3 Computational Performance . . . . .	14
5.4 Code Documentation and Code Extensibility . . . . .	14
<b>6 Adaptations for Agricultural Environments</b>	<b>15</b>
6.1 Tracking Inconsistent Features . . . . .	15
6.2 Poor Exposure . . . . .	16
6.3 Mono vs. Stereo . . . . .	17
6.4 T265 Mounting Angle . . . . .	18
6.5 T265 vs. Alphasense . . . . .	19
6.6 Summary . . . . .	20
<b>7 Evaluation Results</b>	<b>21</b>
7.1 Accuracy . . . . .	21
7.1.1 Accuracy for Online State Estimation . . . . .	21
7.1.2 Effect of Loop Closures . . . . .	22
7.2 Robustness . . . . .	23

7.3	Computational Performance . . . . .	24
7.4	Code Documentation and Code Extensibility . . . . .	25
7.5	Summary . . . . .	25
<b>8</b>	<b>Future Work</b>	<b>27</b>
	<b>Bibliography</b>	<b>30</b>
	<b>A Datasheets</b>	<b>31</b>

# Abstract

Precision agricultural methods that rely on mobile robots are gaining increasing attention as a means for reducing labor costs, increasing crop yield, and correcting environmentally harmful farming practices. As this trend continues, robot localization in agricultural settings becomes of critical importance to ensure accurate state estimation for robotic navigation and mapping. This project explores visual-inertial odometry as a promising method for mobile robot localization in the context of wheeled agricultural robots. Agricultural environments present many unique challenges such as minimal and repetitive visual texture, extreme lighting conditions, and distant horizons with few nearby objects. State-of-the-art VIO methods OpenVINS, ORB-SLAM3, ROVIO, and VINS-Fusion are evaluated on data from a wheeled agricultural robot for metrics of accuracy, robustness, computational performance, and code documentation and extensibility. Finally, a recommendation is made as to which VIO method shows the most promise in the tested agricultural environments.



# Chapter 1

## Introduction

The global agricultural industry is critically important and faces many challenges. The number of people who do not have enough food to eat each year is approaching one billion and increasing annually. Current crop inspection procedures are labor intensive and expensive. Around the world, agricultural chemical runoff is a leading contributor to impaired water quality, which negatively affects human health and endangers wild ecosystems.

Current trends in precision agriculture have begun to develop solutions to these problems that leverage mobile robotic technologies for precise and automated completion of agricultural tasks. As a part of this trend I have partnered with the Rowesys team, an ETH focus project that has created a wheeled precision agricultural robot, named Rosie, shown in figure 1.1. Rosie aims to transform the modern agricultural industry into a highly efficient and environmentally friendly food production system. To do this, Rosie uses physical tools to remove weeds from farmland rather than harsh herbicides. Rosie also uses onboard sensors to take precision measurements of each crop to provide information about current size and present diseases to allow farmers to apply treatments exactly where and when they need it. Changes like these make the agricultural industry more sustainable and better able to meet global food demands without increasing already expensive labor costs.



Figure 1.1: Rosie, the wheeled precision agricultural robot from the Rowesys ETH focus project.

However, the Rowesys team still has a few challenges remaining before Rosie can perform her missions entirely without human intervention. In order to remove weeds from a specific row in the field or capture images of a certain region, Rosie will need to know exactly where she is located within her environment. Currently, Rosie relies heavily on GPS data for localization, which can suffer from connection loss or multi-path issues. Another common means of localization is lidar, but in an agricultural setting lidar localization may be unreliable due to the lack of nearby physical structure in typical crop fields. I propose using camera image streams and IMU measurements to localize with visual-inertial odometry, which has been shown to be a highly accurate and robust localization strategy in many difficult environments. VIO can also be fused with other sensor modalities such as GPS or wheel odometry to further increase accuracy and robustness.

However, VIO in agricultural settings has not yet been fully explored and there are a unique set of challenges to overcome in order to achieve robust performance. Common VIO challenges in agricultural environments are depicted in figure 1.2 and include repetitive visual texture from crop rows, direct sunlight that leads to lens flare, extreme lighting conditions that cause poor image exposure, and moving objects such as clouds or plants swaying in the wind that can cause difficulty when the scene is assumed to be static. With these challenges in mind, it is clear that some modern VIO methods may not easily adapt to agricultural environments. Therefore, the goal of this project is to evaluate state-of-the-art VIO methods in agricultural settings using datasets collected from Rosie, tune the parameters of each method to ensure the best performance, modify methods as needed to adapt them to agricultural environments, select the method that shows the most promise in this challenging context, and propose this method to the Rowesys team to use for localization.

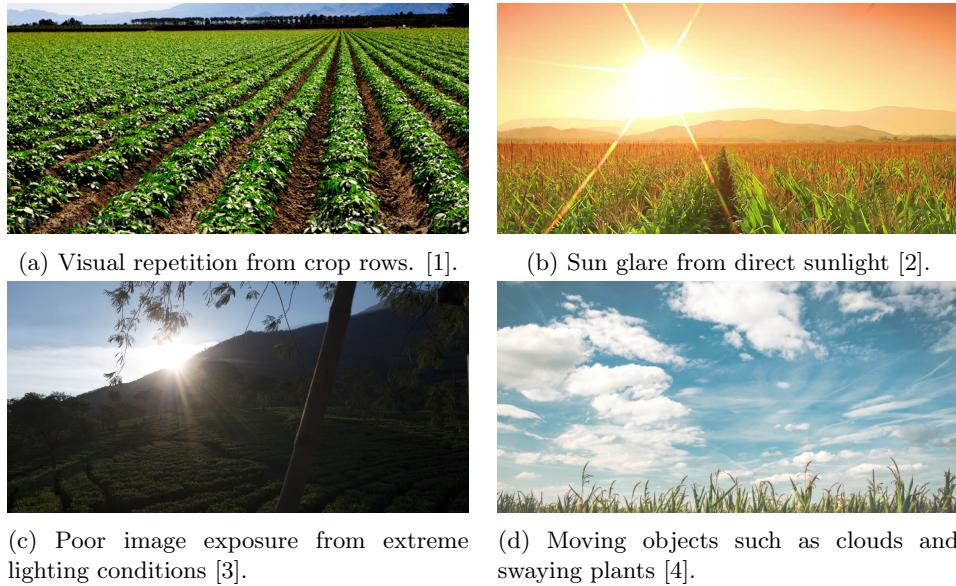


Figure 1.2: Depictions of common VIO challenges in agricultural environments.

# Chapter 2

## VIO Methods

I conducted a thorough literature review and carefully selected a set of 4 VIO methods for evaluation to maximize the chances of making a useful VIO system recommendation to the Rowesys team. In my selection process, I chose methods based on demonstrated accuracy, demonstrated robustness, and diversity in the approach each method took to solving the VIO problem. When evaluating a VIO method for diversity in its VIO implementation, it is useful to use the typical decomposition of a VIO method into a front-end system and a back-end system. The front-end makes observations about the world such as detecting features in an image frame and tracking them by associating features between frames. The back-end consumes these observations and computes an estimate of the sensor motion through the environment. Another area where VIO methods can have diversity is in whether or not they support loop closing functionality. If a VIO method has loop closing functionality, then it is able to detect when the sensor is in the same location twice and use this knowledge to remove some accumulated errors in the estimated trajectory.

### 2.1 OpenVINS

The first method that I selected was OpenVINS [5], which was developed as a research platform to be used and built upon by the VIO community. OpenVINS' front-end uses the FAST [6] feature detector to pick pixels in the image to track, which works by considering a small circle of pixels around each candidate pixel and marks it as a feature if there is a large enough continuous segment of the circle that is lighter or darker than the candidate pixel. OpenVINS then tracks the apparent motion of all detected features with KLT [7] optical flow, which assumes that the pixel intensity of features will be approximately constant as they move through the image and that the motion in a window around each feature is small and nearly constant and it then finds the least squares solution for the apparent motion based on these constraints. In the back-end, OpenVINS uses a multi-state constrained Kalman Filter [8], MSCKF, that allows for including visual features in the filter's measurement model without storing the features in the filter state, which can be computationally expensive. OpenVINS does not have loop closing functionality.

### 2.2 ORB-SLAM3

ORB-SLAM3 [9] was built off of the very successful ORB-SLAM [10] and ORB-SLAM2 [11] systems. In the front-end, the ORB feature detector is used and feature association is done with the ORB descriptor [12]. The ORB descriptor is built by

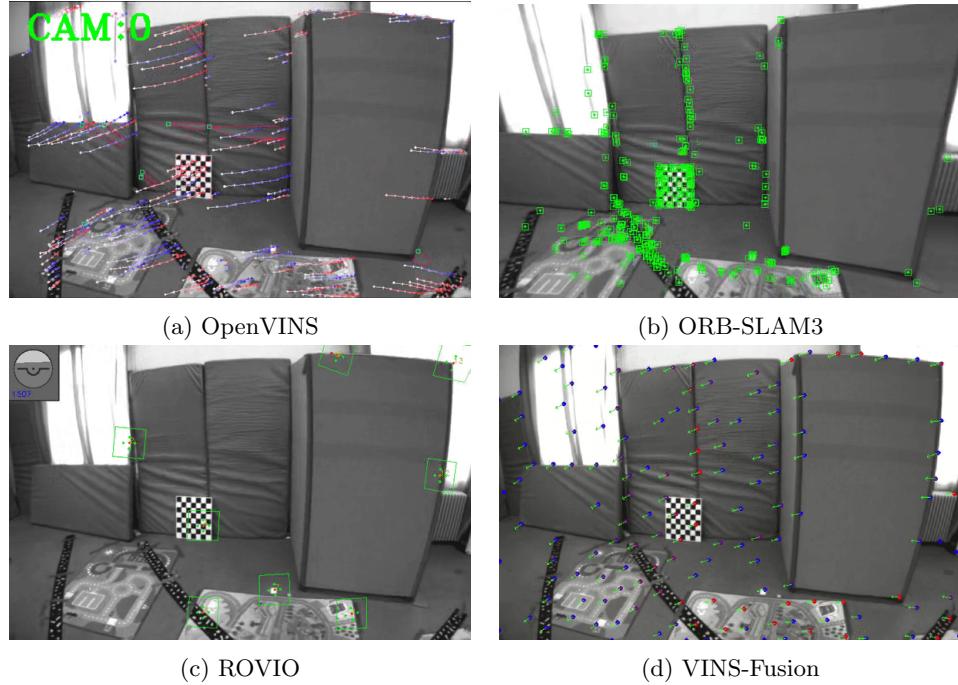


Figure 2.1: A visualization of the front-end of all four VIO methods during a baseline comparison I ran on the EuRoC V101 dataset.

concatenating the binary results of many pixel intensity value comparisons between pre-defined pixel pairs around the feature. The back-end uses bundle adjustment, which does batch optimization over the local feature detections and the trajectory estimate to minimize error between the estimated 3D world features and the 2D observations of those features in the local image frames. Loop closing is built into ORB-SLAM3, but I modified the method such that it could optionally be turned off to facilitate comparisons to other methods without loop closures.

### 2.3 ROVIO

ROVIO [13] uses the Shi Tomasi feature detector to select features for tracking. It then tracks the selected features using direct pixel intensity values of a patch extracted around each feature. ROVIO uses an iterated extended Kalman Filter (IEKF) in the back-end. Unlike OpenVINS, ROVIO includes the tracked features in the filter state. This allows for very tight integration of landmarks into the filter, but places a limit on the number of features that can be tracked since a high dimensional filter state space can become computationally intractable very quickly. By default ROVIO tracks 25 features, compared to the hundreds of features typically tracked by other methods. ROVIO does not have loop closing functionality.

### 2.4 VINS-Fusion

The final method I chose is VINS-Fusion [14], a stereo-camera extension of the VINS-Mono [15] system. Like ORB-SLAM3, the back-end is based around bundle adjustment, but instead of using feature descriptors for association in the front-end, VINS-Fusion uses optical flow similar to OpenVINS. The tracked features are

selected using the Shi Tomasi detector, which is similar to ROVIO. VINS-Fusion comes with the option to enable loop closing.

## 2.5 Summary

I considered other VIO methods for this evaluation, such as OKVIS [16], BASALT [17], and Kimera [18], but did not include them since I found they had less impressive accuracy claims or added less diversity to my selected methods. Table 2.1 contains a summary of the key characteristics of the methods I selected, which shows the diverse set of strategies employed by these methods in solving the VIO problem. The methods include three different types of features, direct and indirect association strategies, two methods with bundle adjustment based estimation and two methods with filter-based estimation (with different Kalman Filter variants), and two systems with relocalization and loop closures and two systems without relocalization and loop closures. This carefully selected diversity increases the chances of finding a method that can adapt well to agricultural settings and increases the chances of making a useful VIO method recommendation to the Rowesys team. I ran a baseline comparison of all methods on the EuRoC datasets [19] to validate the accuracy described in the VIO method publications and to ensure that I could run all methods successfully on my machine. A visualization of the front-end of all methods during this baseline EuRoC comparison is shown in figure 2.1.

	<b>Pixel / Feature</b>	<b>Association</b>	<b>Estimation</b>	<b>Relocation</b>	<b>Loop Closing</b>
<b>OpenVINS</b>	S: FAST	S: KLT	MSCKF	<i>None</i>	<i>None</i>
<b>ORB-SLAM3</b>	S & L: ORB	S & L: Descriptor	Local BA	DBoW2	DBoW2 PG + BA
<b>ROVIO</b>	S: Shi Tomasi	S: Direct	IEKF	<i>None</i>	<i>None</i>
<b>VINS-Fusion</b>	S: Shi Tomasi L: BRIEF	S: KLT L: Descriptor	Local BA	DBoW2	DBoW2 PG

Table 2.1: Summary of the key characteristics of the selected VIO methods. S: short-term, e.g. frame-to-frame. L: long-term, e.g. loop closure. BA: bundle adjustment. PG: pose graph.



# Chapter 3

## Sensors

### 3.1 Visual-Inertial Sensors

I selected two visual-inertial sensors for data collection, the T265 from Intel and the Alphasense from Sevensense, such that I could collect VIO data on both sensors and compare VIO results between the two. To ensure the best possible VIO results, I calibrated both sensors myself using Kalibr<sup>1</sup>, rather than using the factory-provided calibrations.



(a) The T265 mounted on Rosie. The custom 3D printed mount allows for easily changing the pitch of the sensor.

(b) The Alphasense mounted on Rosie using a fixed-angle custom 3D printed mount.

Figure 3.1: The T265 and the Alphasense mounted on Rosie with custom 3D printed mounts.

<sup>1</sup><https://github.com/ethz-asl/kalibr>

### 3.1.1 T265

The first visual-inertial sensor I used was the Intel Realsense Tracking Camera T265<sup>2</sup>, which had been used by the Rowesys team before on Rosie, so it was readily available and the software integration work had been done before. Therefore, I only had to mechanically integrate the sensor before collecting data. The T265 comes with on-device VIO capabilities, which the Rowesys team has tried using in the past, but they were unable to get it working robustly in agricultural settings and abandoned it. The T265 is relatively affordable compared to the Alphasense, has an ultra-wide FOV, and produces a nearly square image which is slightly larger in both dimensions than the Alphasense image. The T265 provides lower quality data than the Alphasense, with bigger camera-IMU time offsets. Additionally, the accelerometer and gyroscope inside of the T265 IMU publish at very different rates, 62.5 Hz and 200 Hz respectively. Since all of the evaluated VIO methods expect a combined IMU measurement, rather than individual accelerometer and gyroscope measurements, the accelerometer measurements must be interpolated to match the frequency of the gyroscope, which introduces a source of error in the IMU measurements. Lastly, the ROS driver for the T265 allows for very little configurability. For example, the image exposure and camera framerate cannot be modified. As a result, I had to use the T265 at 30 frames-per-second although I would have preferred 20 frames-per-second which I found to be more commonly used for VIO. Figure 3.1a shows the T265 mounted on Rosie.

### 3.1.2 Alphasense

The second visual-inertial sensor that I used was the Sevensense Alphasense<sup>3</sup>. The Alphasense had never been used on Rosie before, so I needed to do the full software and hardware integration. The Alphasense is more expensive than the T265, has a wide FOV, though not as wide as the T265, and has a more standard 4:3 image aspect ratio that is slightly smaller than the T265 image in both dimensions. The Alphasense also provides higher quality data than the T265 with smaller camera-IMU time offsets. The Alphasense accelerometer and gyroscope publish at the same rates and therefore the IMU measurements do not rely on interpolation. Lastly, the ROS driver for the Alphasense allows for a relatively high degree of configurability compared to the T265. For example, the exposure, camera framerate, IMU frequency, and much more can be modified. Figure 3.1b shows the Alphasense mounted on Rosie.

### 3.1.3 Summary

Table 3.1 summarizes the key characteristics of the T265 and the Alphasense side-by-side. Including both of these sensors in the evaluation makes the results relevant to a wider range of agricultural robotic systems, from hobbyist-level systems equipped with sensors like the T265 to research-grade systems equipped with sensors like the Alphasense.

---

<sup>2</sup><https://www.intelrealsense.com/tracking-camera-t265/>, datasheet in appendix A.

<sup>3</sup>[https://github.com/sevensense-robotics/alphasense\\_core\\_manual](https://github.com/sevensense-robotics/alphasense_core_manual), datasheet in appendix A

	<b>T265</b>	<b>Alphasense</b>
<b>Cost</b>	~300 CHF	~3,000 CHF
<b>FOV</b>	Ultra-wide: $163^\circ \times 163^\circ$	Wide: $126^\circ \times 92^\circ$
<b>Resolution</b>	$848 \times 800$	$720 \times 540$
<b>Camera-IMU Time Offset</b>	< 5 ms	< 1 ms
<b>IMU Interpolation</b>	Required	Not required
<b>Degree of Customizability</b>	Low	High

Table 3.1: Summary of the key characteristics of the T265 and Alphasense.

## 3.2 Groundtruth Sensor

I used RTK GPS to serve as groundtruth data for comparing VIO trajectory estimates against. I used the SwiftNavigation GNSS GPS500 Antenna<sup>4</sup>, which was already fully integrated into Rosie and ready for use. All trajectory visualizations shown in this report are visualized in the frame of the GPS antenna. Figure 3.2 shows the GPS antenna mounted on Rosie’s chassis.



Figure 3.2: GPS antenna (white) mounted on Rosie.

---

<sup>4</sup><https://www.swiftnav.com/store/accessories/gnss-antenna-pack>, datasheet in appendix A.



# Chapter 4

## Datasets

### 4.1 Eschikon

To test the selected VIO methods, I put a large focus on collecting a diverse set of high quality data on Rosie. I collected all datasets used for this project at the Eschikon ETH agricultural research center on wheat and corn fields. With the T265, the Alphasense, and the RTK GPS antenna on Rosie, I collected a large variety of agricultural VIO datasets to cover as many use-case scenarios as possible. The data I collected includes different weather conditions, times of day, sensors, sensor mounting positions, driving patterns, and crop fields. As shown in figure 4.1, I put a focus on gathering data that simulates typical crop inspection missions, where Rosie would navigate a whole field row-by-row, and then return to her starting position.

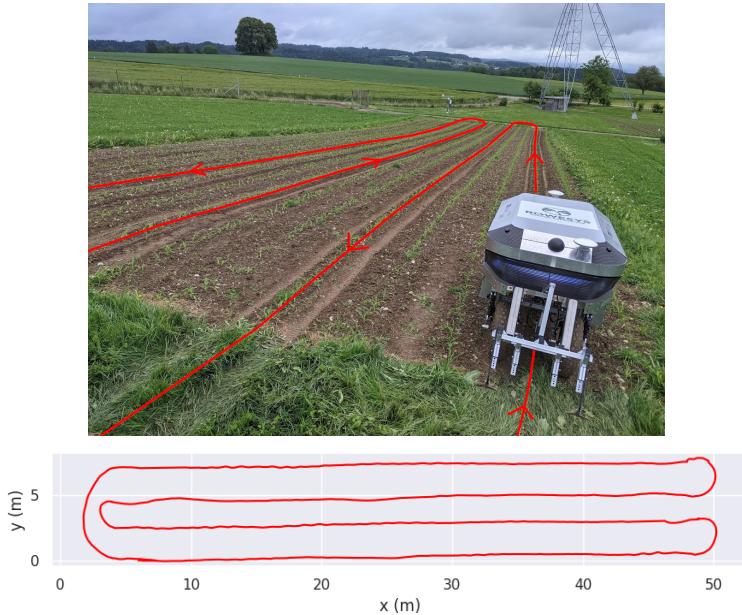


Figure 4.1: A driving pattern that I commonly used when collecting VIO datasets. The pattern simulates a typical crop inspection mission that traverses the whole field and returns to the starting position.

A small sample of images from an Eschikon dataset are shown in figure 4.2. The top row of images were captured while Rosie was traversing along crop rows. The

bottom row of images were captured while Rosie was transitioning between crop rows. The dataset has a 4-row-loop driving pattern similar to the dataset shown in figure 4.1 and was collected on the corn field on an overcast day with the Alphasense.

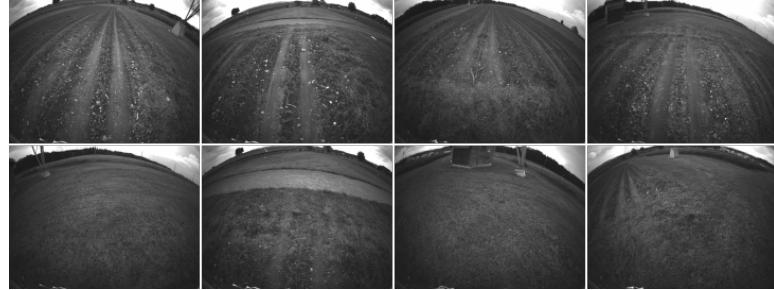


Figure 4.2: Some sample images from an Eschikon dataset. The top row of images were captured while Rosie was traversing along crop rows. The bottom row of images were captured while Rosie was transitioning between crop rows. The dataset has a 4-row-loop driving pattern and was collected on the corn field on an overcast day with the Alphasense.

## 4.2 Rosario

The Rosario datasets [20] provide publicly available agricultural robotics datasets with stereo camera images, IMU measurements, and GPS readings in a similar fashion to the datasets that I have collected for this project. At the beginning of this project, I had hoped to evaluate all VIO methods on the Rosario datasets in addition to Eschikon datasets. I set up all VIO methods to run on the Rosario datasets, but encountered many issues and eventually discovered after deeper debugging and discussions with the authors of the Rosario paper that there are issues with the Rosario data that make it unusable with visual-inertial odometry methods. The IMU data is corrupted and therefore only visual odometry methods can run properly on the datasets. The issue is being tracked on their Github repository<sup>1</sup> and will be updated when the problem is fixed. As a result of these issues, I did not use the Rosario datasets in this project.

---

<sup>1</sup><https://github.com/CIFASIS/dataset-processing/issues/2#issuecomment-686053266>

# Chapter 5

## Evaluation Metrics

To evaluate the selected VIO methods, I compared them on metrics defined by the needs of the Rowesys team, which are: accuracy, robustness, computational performance, and code documentation and extensibility.

### 5.1 Accuracy

#### 5.1.1 Accuracy for Online State Estimation

To ensure accurate state estimation, the trajectory estimated by a VIO method must closely match the true trajectory. In their localization system, the Rowesys team will be incorporating each relative VIO motion estimate into their state estimation pipeline incrementally along with information from other sensors. Therefore, the Rowesys team cares most about local consistency (rather than the global consistency) of a VIO method's estimated trajectory compared to the groundtruth, since only the relative motion estimates will be used. Typically, the well-known relative position error metric would be used to measure local consistency, but that requires groundtruth orientations to compute, which I do not have since I am using position-only RTK GPS data as groundtruth. Instead of relative position error, I use segment-wise absolute position error to measure local consistency. Segment-wise APE breaks the groundtruth trajectory into 5 meter segments, aligns each estimated trajectory segment to the associated groundtruth segment, computes the absolute position errors from each segment pair, and then returns the maximum APE from each segment pair.

#### 5.1.2 Effect of Loop Closures

In the future, the Rowesys team may use estimated trajectories from VIO systems to create maps of the environment. For this use-case, loop closing functionality will prove useful for removing accumulated drift in trajectory estimates. I use a relative error metric (segment-wise APE) and an absolute error metric (APE) to evaluate the effect of loop closures for the VIO methods that have loop closing functionality.

### 5.2 Robustness

The Rowesys team will use VIO in a variety of difficult scenarios and requires a VIO method that can perform well under many challenging contexts. For example, the Rowesys team may not have perfect sensor calibration information or may imperfectly tune a VIO method and they still need the VIO method to produce

reliably accurate results under these imperfect conditions on long duration and challenging missions. To evaluate the robustness of each VIO method, I ran each VIO method under many stressful conditions such as fast angular velocity, highly bumpy terrain, and direct sunlight and evaluated how robust each method was to these difficult scenarios by looking at the accuracy of the estimated trajectory over repeated runs.

### 5.3 Computational Performance

Most VIO systems suggest processing at least 10 image frames per second to achieve good performance, which makes it important to measure processing times per image frame for each VIO method. Additionally, it is important to measure the computation time per frame for each VIO method to better understand their respective computational loads and to know how much computation time Rosie has remaining to use for other computationally-intensive tasks. To evaluate computational performance, I timed how long it takes each method to process each frame over a typical dataset using a machine with computational resources comparable to the computer on Rosie.

### 5.4 Code Documentation and Code Extensibility

No VIO system will perfectly meet the needs of the Rowesys team in every way, so it is important that a VIO system's software is well documented and has cleanly-written code so that it is easy to customize and extend. I spent time reading through and modifying the code for each VIO method and then evaluated them on the quantity and quality of the code documentation and how easy the code was to modify and extend.

# Chapter 6

## Adaptations for Agricultural Environments

Despite selecting four very impressive VIO methods to evaluate, none of them were specifically built for the difficult conditions in agricultural settings, and I needed to make some adaptations before I could get results that were good enough for a high-quality comparison. Figure 6.1 shows poor initial results without any adaptations where the dashed line is the ground truth trajectory and the colored lines are the trajectory estimates from each method. Before discussing the VIO evaluation results, I'll describe some of the issues I encountered while using VIO in agricultural environments and the corresponding adaptations that I made to overcome them.

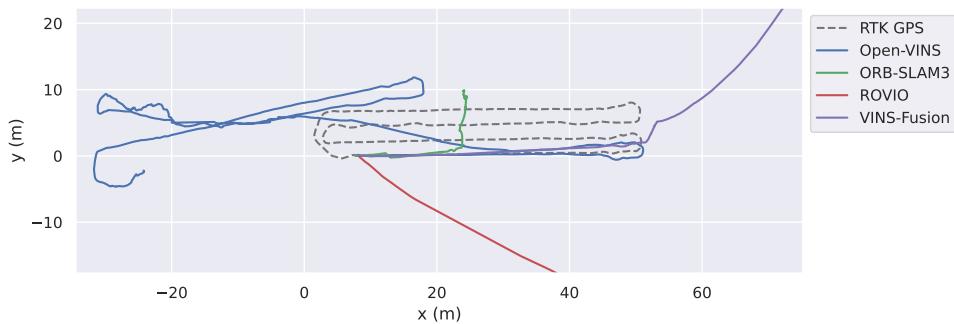


Figure 6.1: Poor initial VIO results before any adaptations were made for agricultural environments.

### 6.1 Tracking Inconsistent Features

The first issue I noticed was the tracking of inconsistent features. Rosie's chassis and the clouds in the sky would often be in the camera FOV and the VIO methods would attempt to track features here, which would lead to lower quality tracking. When using cameras with very wide FOVs, it is impossible to mount the sensor on the front of Rosie such that both the chassis and the sky are not in the image. To fix this, I modified each method to accept a predefined image mask and then reject all features detected outside of the mask, such as on Rosie's chassis or in the sky. Figure 6.2 shows green features tracked by ORB-SLAM3 without image masking on the left with many inconsistent features, and with image masking on the right

with all consistent features. The portion of the image included in the image mask, where features are allowed to be tracked, is shown in the middle.

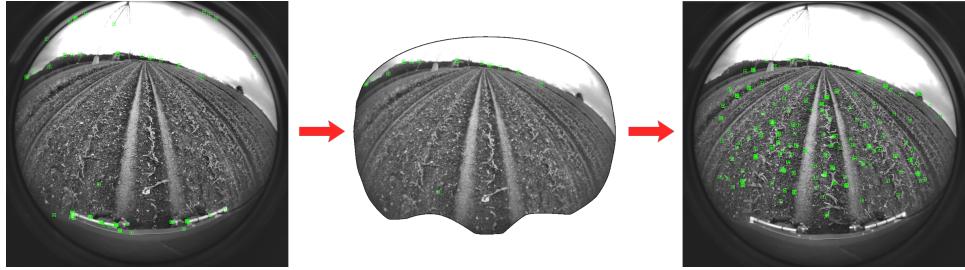


Figure 6.2: Left: before image masking is applied with inconsistent features being tracked. Center: the region of the image within the image mask to be applied (where features are allowed to be tracked). Right: after image masking is applied with consistent features being tracked.

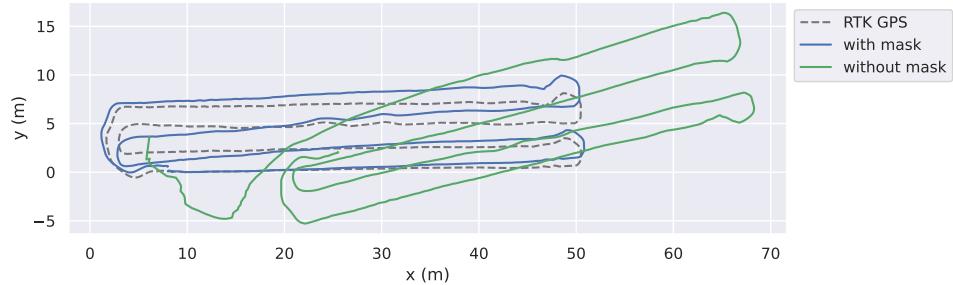


Figure 6.3: OpenVINS trajectories with and without image masking. The estimated trajectory with image masking aligns much more closely to the groundtruth than the trajectory without image masking.

Using masking resulted in performance improvement in all methods, as shown in table 6.1. Figure 6.3 shows OpenVINS’ estimated trajectories with and without image masking compared to the groundtruth trajectory, where the estimated trajectory with image masking follows the groundtruth much more closely.

	Without Masking	With Masking
<b>OpenVINS</b>	5.66	0.40
<b>ORB-SLAM3</b>	—	0.22
<b>ROVIO</b>	0.97	0.52
<b>VINS-Fusion</b>	1.48	0.14

Table 6.1: Max segment-wise APE (m), with and without image masking. ORB-SLAM3 did not complete the dataset without image masking.

## 6.2 Poor Exposure

The next issue that I noticed was poor image exposure. Most of the useful visual texture for tracking in agricultural environments is on the ground, so if it is too under-exposed then that may lead to degraded VIO performance. To investigate this problem, I wrote a ROS node to pre-process all images with a histogram equalization algorithm called CLAHE [21] before being consumed by the VIO methods.

Figure 6.4 shows the effect of CLAHE on an image with poor exposure. Histogram equalization attempts to spread out pixel intensity values and increase contrast in an image by linearizing the CDF of an image’s pixel intensities, shown as black lines in the figure. This effectively flattens out an image’s pixel intensity histogram, which are shown in red in the figure. In the example image in the top left of the figure, the large quantity of dark pixels on the ground are spread out into lighter intensity bins, which increases the contrast of the ground’s visual texture.

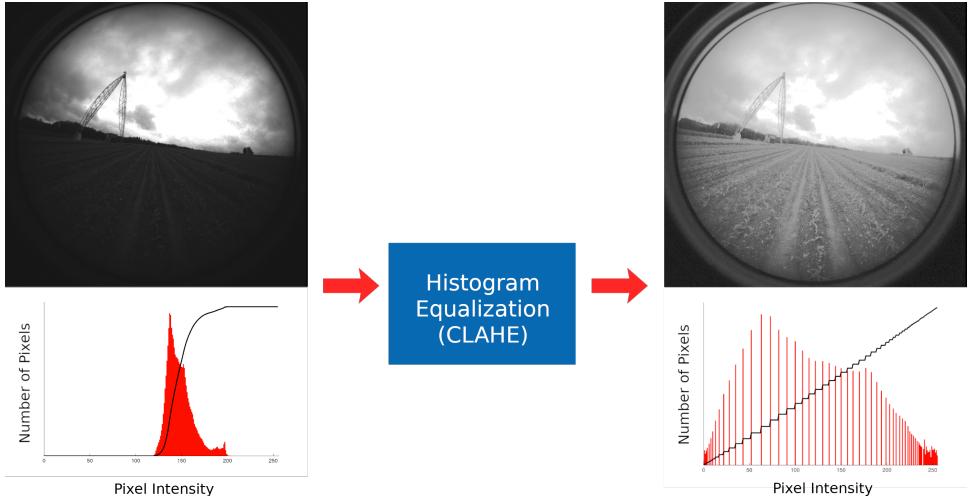


Figure 6.4: An image before and after CLAHE is applied. The pixel intensity histograms are shown in red and the pixel intensity CDFs are shown in black. CLAHE redistributes pixel intensities to make the CDF more linear, which flattens out the pixel intensity histogram.

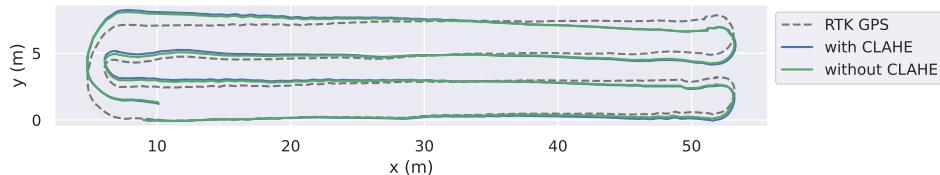


Figure 6.5: VINS-Fusion estimated trajectories with and without CLAHE. CLAHE has a negligible effect on VINS-Fusion’s optical flow-based front-end, so both trajectories are nearly identical.

As shown in table 6.2, CLAHE had a large positive effect on ORB-SLAM3, a feature-based method, and little or no effect on the other methods. I believe that CLAHE leads to more descriptive ORB feature descriptors since higher contrast would change the results of its pair-wise pixel comparisons. However, for the other methods, which rely only on image intensity differences for tracking, there would be a drastically smaller effect, as shown with VINS-Fusion in figure 6.5 where the estimated trajectories with and without CLAHE are very similar.

### 6.3 Mono vs. Stereo

I also ran experiments to evaluate performance differences between monocular and stereo VIO. There was an increase in accuracy and robustness on all methods when

	<b>Without CLAHE</b>	<b>With CLAHE</b>
<b>OpenVINS</b>	0.23	0.23
<b>ORB-SLAM3</b>	2.64	0.34
<b>ROVIO</b>	0.82	0.65
<b>VINS-Fusion</b>	0.14	0.14

Table 6.2: Max segment-wise APE (m), with and without CLAHE.

switching to stereo, which is shown in table 6.3. Notably, VINS-Fusion’s performance increase was the smallest due to its already very impressive performance in monocular mode. I also investigated IMU bias estimates between monocular and stereo modes, which are shown in figure 6.6 for ROVIO. Note that the IMU bias estimates are nearly identical for mono and stereo, which suggests that the increase in accuracy in stereo mode is from the ability to do direct stereo triangulation of features, rather than from significantly improved scale estimation.

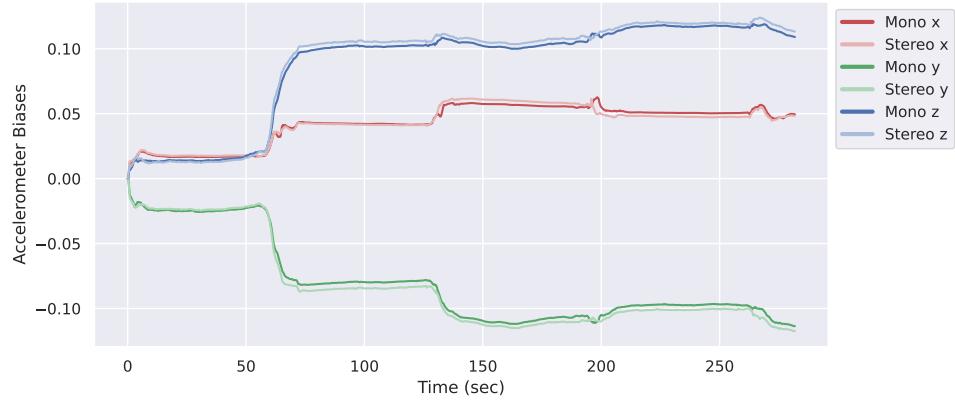


Figure 6.6: ROVIO accelerometer biases in mono and stereo mode.

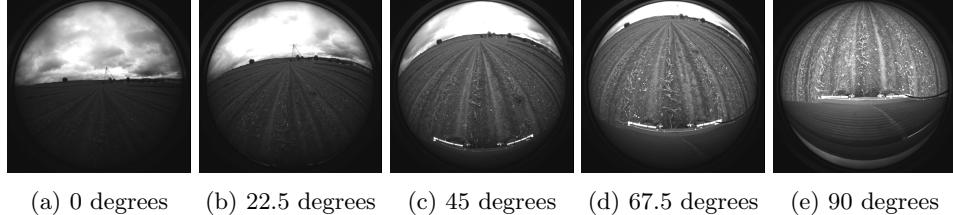
	<b>Mono</b>	<b>Stereo</b>
<b>OpenVINS</b>	1.93	0.12
<b>ORB-SLAM3</b>	0.55	0.22
<b>ROVIO</b>	0.43	0.34
<b>VINS-Fusion</b>	0.17	0.14

Table 6.3: Max segment-wise APE (m), with mono and stereo.

## 6.4 T265 Mounting Angle

The custom 3D printed bracket that I used to mount the T265 allowed for easy modification of the sensor’s pitch angle. To see if a certain pitch angle produced better performance than other angles, I collected 5 datasets back-to-back with different T265 mounting angles and evaluated OpenVINS’ performance with each mounting angle. Figure 6.7 shows example images from the T265 while mounted at the 5 different angles that I evaluated. The most notable differences between the mounting angles are the quality of the exposure of the ground and the amount of robot chassis and sky included in the image. After using histogram equalization and image masking, the effects of these differences were almost entirely mitigated. As a result,

the VIO accuracies for each mounting angle were very similar, as shown in figure 6.8.



(a) 0 degrees (b) 22.5 degrees (c) 45 degrees (d) 67.5 degrees (e) 90 degrees

Figure 6.7: Example images taken from the T265 with different mounting angles.

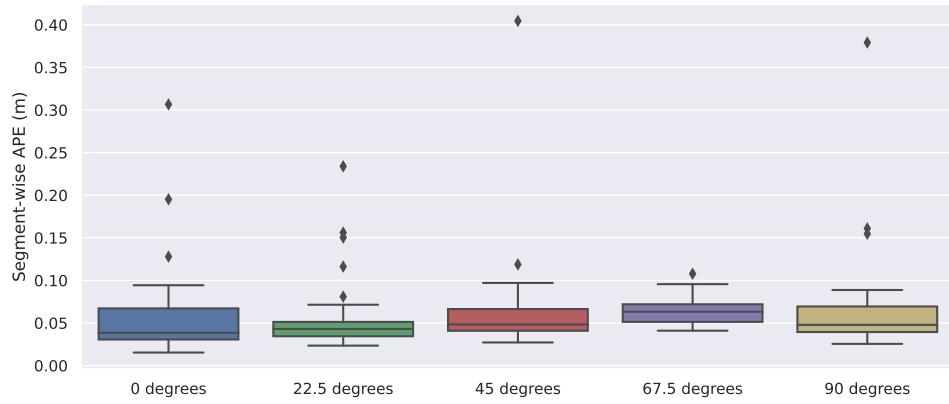


Figure 6.8: The distribution of errors for OpenVINS' trajectory estimate with different T265 mounting angles. The performance did not significantly differ between mounting angles.

## 6.5 T265 vs. Alphasense

I collected a dataset with both the T265 and the Alphasense mounted on Rosie at the same time and compared the results of processing that dataset with all VIO methods on both sensors with all aforementioned adaptations. Figure 6.9 shows an example image from each sensor from the sensor comparison dataset. As shown in table 6.4, all methods improved when using the Alphasense due to the more tightly timestamped data it provides without relying on interpolation for IMU measurements. OpenVINS had especially accurate results on the T265 compared to other methods, which is a testament to its robustness to lower quality data due in part to its built-in online time offset correction.

	<b>T265</b>	<b>Alphasense</b>
<b>OpenVINS</b>	0.31	0.15
<b>ORB-SLAM3</b>	2.14	0.63
<b>ROVIO</b>	1.33	0.30
<b>VINS-Fusion</b>	1.12	0.11

Table 6.4: Max segment-wise APE (m), with the T265 and the Alphasense.

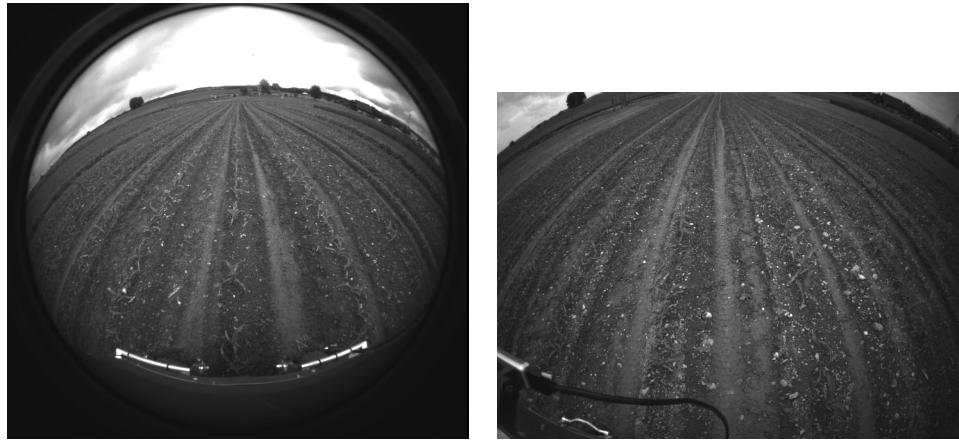


Figure 6.9: An example image from both the T265 and the Alphasense from the sensor comparison dataset. The left image is from the T265 and the right image is from the Alphasense.

## 6.6 Summary

To achieve the best possible performance with VIO methods in agricultural environments, I applied image masking to ensure consistent tracking of features, used histogram equalization to improve image contrast and exposure, used methods in stereo mode for better feature triangulation, and used the Alphasense for smaller camera-IMU time offsets without depending on interpolation for IMU measurements. I recommend using all of these adaptations when using VIO in agricultural environments. However, if a direct pixel-based VIO method is being used on a very compute-limited platform, then CLAHE could be removed since I did not observe significant performance improvement on pixel-based methods and it does require a small amount of computational resources to process each image with histogram equalization.

# Chapter 7

## Evaluation Results

After applying image masking, histogram equalization, setting each VIO method to stereo mode, and collecting data from the Alphasense, all VIO methods were producing results that were accurate enough for a comparative evaluation in the given difficult environmental conditions. All of the following results use all of these adaptations unless explicitly mentioned otherwise.

### 7.1 Accuracy

#### 7.1.1 Accuracy for Online State Estimation

To evaluate accuracy for online state estimation, I ran each VIO method on the same dataset with all aforementioned adaptations applied. Note that I include results for ROVIO mono among all of the other stereo methods since ROVIO stereo does not run in realtime with the computational resources available on Rosie since bugs have been introduced in recent versions of the software. Therefore, ROVIO stereo will not be in the final evaluation, but is included in the accuracy evaluation for completeness. Older versions of ROVIO stereo without bugs are not included in the evaluation since those versions are no longer maintained. Figure 7.1 and figure 7.2 show the estimated trajectories compared to groundtruth for each VIO method. Figure 7.3 shows the distribution of segment-wise APEs for each VIO method and figure 7.4 shows statistics about the error distributions. As summarized in table 7.1, OpenVINS and VINS-Fusion achieved the highest accuracy, followed by ORB-SLAM3, and then ROVIO mono.

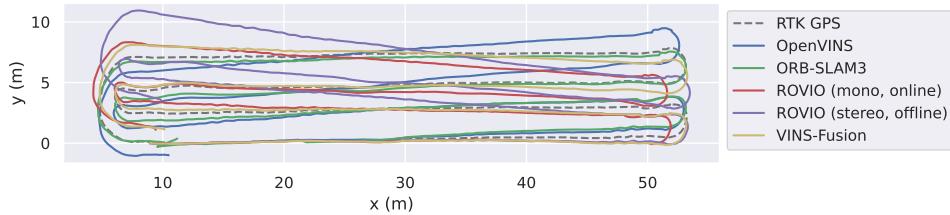


Figure 7.1: A top-view of the estimated trajectories from all VIO methods after all adaptations were applied.

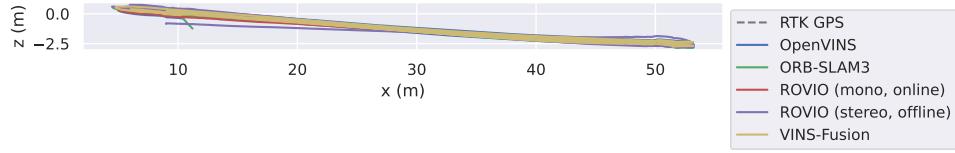


Figure 7.2: A side-view of the estimated trajectories from all VIO methods after all adaptations were applied. The crop field is sloped, which is reflected in the estimated trajectories.

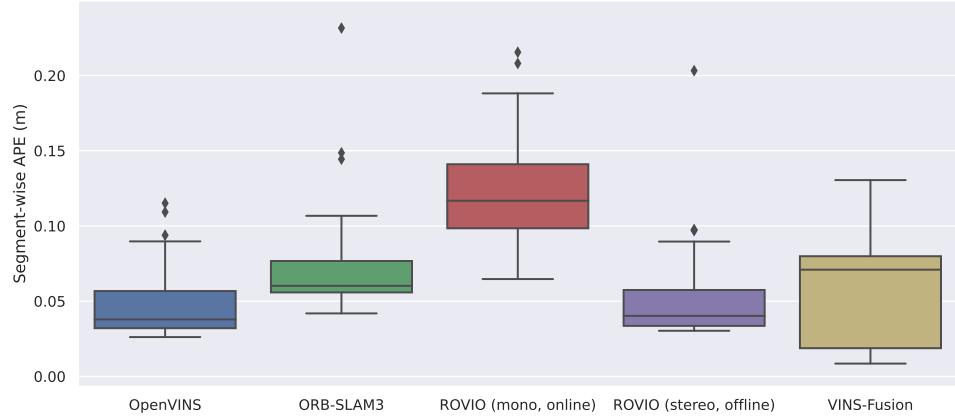


Figure 7.3: The distribution of errors for all VIO methods.

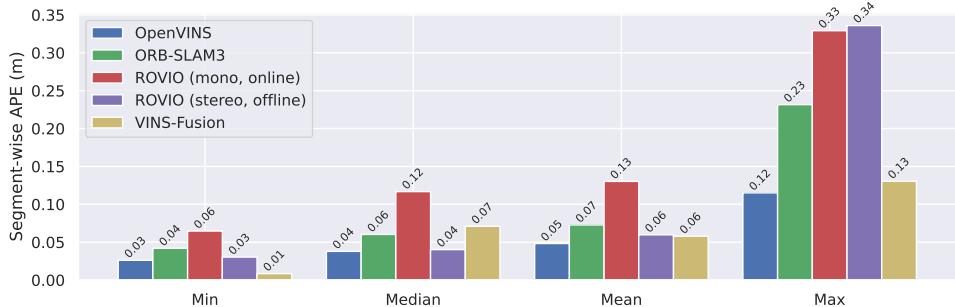


Figure 7.4: Statistics about the distribution of errors for all VIO methods.

### 7.1.2 Effect of Loop Closures

For the two methods with loop closure capabilities, ORB-SLAM3 and VINS-Fusion, I evaluated their ability to eliminate drift with loop closure functionality enabled over a long dataset that travels the same looped path twice. Both methods detected and closed a large numbers of loops throughout the dataset and produced results with almost no drift as shown in figure 7.5. Figure 7.6 shows the distribution of relative errors (segment-wise APE) for both VIO methods and figure 7.7 shows the distribution of absolute errors (APE). The small amount of the drift in the estimated trajectories results in low relative errors. VINS-Fusion has slightly lower relative and absolute errors than ORB-SLAM3 and so VINS-Fusion receives a higher rating as shown in table 7.2.

OpenVINS	ORB-SLAM3	ROVIO (mono)	VINS-Fusion
Excellent	Good	Fair	Excellent

Table 7.1: Ratings for all VIO methods on the metric of accuracy for online state estimation.

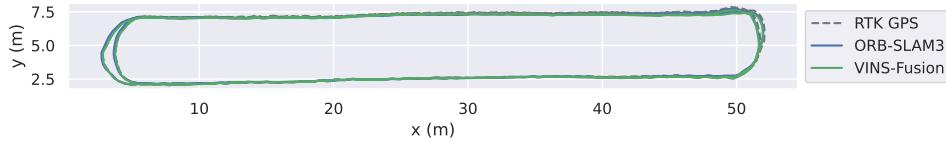


Figure 7.5: Trajectories estimated by ORB-SLAM3 and VINS-Fusion with loop closing enabled.

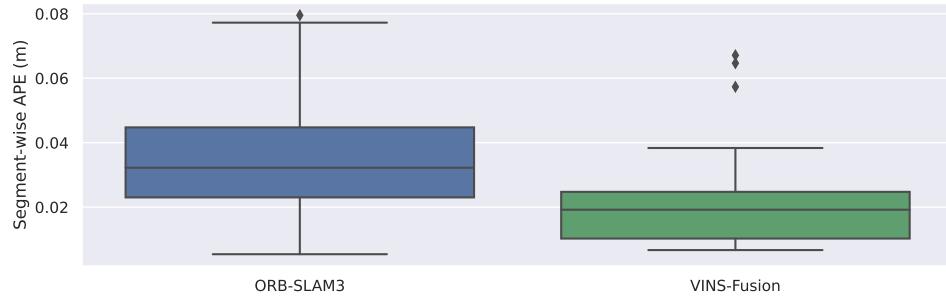


Figure 7.6: Relative errors (segment-wise APE) for ORB-SLAM3 and VINS-Fusion with loop closing enabled.

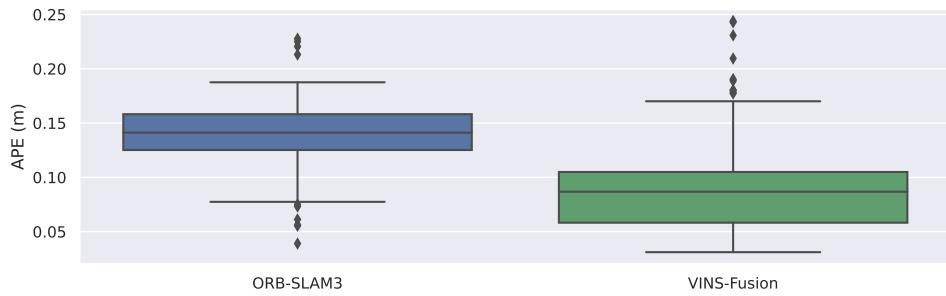


Figure 7.7: Absolute errors (APE) for ORB-SLAM3 and VINS-Fusion with loop closing enabled.

## 7.2 Robustness

Throughout my testing, OpenVINS and VINS-Fusion have shown above-average robustness to difficult situations, such as having an inaccurate calibration or handling especially difficult environmental situations such as large bumps and high velocity turns. ORB-SLAM3 and ROVIO have shown less robustness to these situations. Therefore, as shown in table 7.3, OpenVINS and VINS-Fusion receive a Good rating for robustness, followed by ORB-SLAM3 and ROVIO with a Fair rating.

<b>OpenVINS</b>	<b>ORB-SLAM3</b>	<b>ROVIO (mono)</b>	<b>VINS-Fusion</b>
<i>N/A</i>	Good	<i>N/A</i>	Excellent

Table 7.2: Ratings for all VIO methods on the metric of effect of loop closures.

<b>OpenVINS</b>	<b>ORB-SLAM3</b>	<b>ROVIO (mono)</b>	<b>VINS-Fusion</b>
Good	Fair	Fair	Good

Table 7.3: Ratings for all VIO methods on the metric of robustness. OpenVINS and ROVIO do not have loop closing functionality.

### 7.3 Computational Performance

I timed how long each method takes to process each pair of image frames on a typical dataset to evaluate their computational performance. The timings were done on an Intel Core i7-8550U CPU, which is comparable in computational resources to the machine inside of Rosie, and are shown in figure 7.8 and figure 7.9. OpenVINS and ROVIO mono are able to run at 20 Hz, ORB-SLAM3 and VINS-Fusion are able to run at 10 Hz, and ROVIO stereo had to be run offline. The resulting evaluation rankings are shown in table 7.4. Note that ORB-SLAM3 always has a single frame that takes around half a second to process due to a lengthy global refinement step that it runs once the map reaches a certain size.

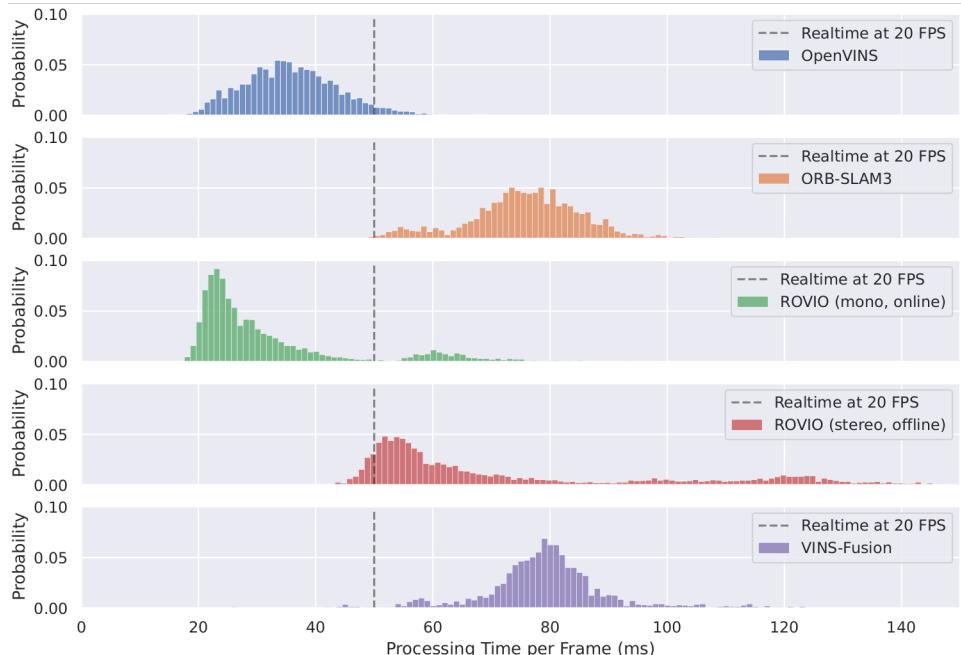


Figure 7.8: Distributions for how long each VIO method takes to process a single image frame.

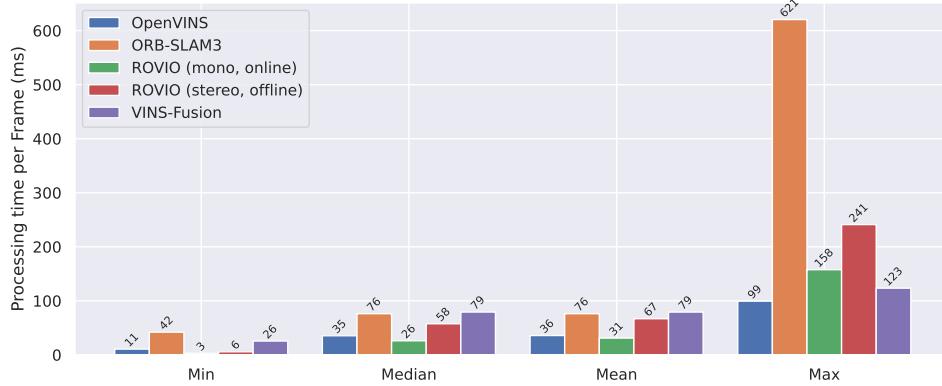


Figure 7.9: Statistics about the distributions for how long each VIO method takes to process a single image frame. Note that ORB-SLAM3’s global refinement step, which runs once per dataset after the map reaches a certain size, results in a large maximum processing time.

OpenVINS	ORB-SLAM3	ROVIO (mono)	VINS-Fusion
Good	Fair	Good	Fair

Table 7.4: Ratings for all VIO methods on the metric of computational performance.

## 7.4 Code Documentation and Code Extensibility

I spent a lot of time reading and modifying the code for each VIO method, which allowed me to understand how well-documented the code is for each method and how understandably-written the code is for each method. OpenVINS has extensive and understandable documentation paired with a very cleanly-written codebase that makes it very extensible. ROVIO’s code, although very intelligently designed, was relatively difficult for me to understand and extend. ORB-SLAM3 and VINS-Fusion have moderately understandable code, but are not very thoroughly documented. The code documentation quality and code extensibility for each method are summarized in table 7.5.

	OpenVINS	ORB-SLAM3	ROVIO (mono)	VINS-Fusion
Documentation	Excellent	Fair	Fair	Fair
Extensibility	Excellent	Good	Poor	Good

Table 7.5: Ratings for all VIO methods on the metrics of code documentation and code extensibility.

## 7.5 Summary

In conclusion, I rank OpenVINS, ORB-SLAM3, ROVIO, and VINS-Fusion on the metrics of accuracy (with and without loop closing), computational performance, robustness, and code documentation and extensibility as shown in table 7.6. My final recommendation for the Rowesys team, and any other precision agricultural systems interested in VIO for online state estimation, is to use OpenVINS for its top-tier accuracy, impressive robustness, low computational cost, and thorough documentation. If a system with loop closures is absolutely necessary, then I recommend

VINS-Fusion which, compared to OpenVINS, achieves equally impressive accuracy, but has a slight increase in computational cost and is not as well documented.

	OpenVINS	ORB-SLAM3	ROVIO (mono)	VINS-Fusion
<b>Accuracy for Online State Estimation</b>	Excellent	Good	Fair	Excellent
<b>Effect of Loop Closures</b>	N/A	Good	N/A	Excellent
<b>Computational Performance</b>	Good	Fair	Good	Fair
<b>Robustness</b>	Good	Fair	Fair	Good
<b>Code Documentation</b>	Excellent	Fair	Fair	Fair
<b>Code Extensibility</b>	Excellent	Good	Poor	Good

Table 7.6: Summary of all VIO method rankings on all evaluation metrics.

## Chapter 8

# Future Work

Now that a promising VIO method has been identified, it will need to be fully integrated into Rosie and tested in real-time on the crop fields. Another ongoing semester project is focused on a pose graph-based sensor fusion state estimation pipeline for Rosie. Once that project is completed, VIO will need to be integrated into the pipeline to be fused into the final state estimate.

There is also much more work to be done in adapting VIO systems for agricultural robotics. I see large areas of opportunity for deeper investigation of lighting and exposure issues. Typically, the most difficult lighting challenges with VIO result from directly viewing the sun in an image, so it could prove useful for future work to investigate online glare detection, or to use multi-camera rigs, such that if one camera is facing into the sun, another can be used to continue accurate tracking.



# Bibliography

- [1] “Barn owl box company,” <https://www.barnowlbox.com/crops-and-barn-owls/>, accessed: 21.06.2021.
- [2] “Shutterstock,” <https://www.shutterstock.com/video/clip-3451814-corn-farm-thailand--crane-shot-flare>, accessed: 21.06.2021.
- [3] “Getty images,” <https://www.gettyimages.it/video/sunrise-farm>, accessed: 21.06.2021.
- [4] “Pixnio,” <https://pixnio.com/flora-plants/crops/corn-field-corn-crops-field-countryside-farm-clouds-blue-sky>, accessed: 21.06.2021.
- [5] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. Huang, “Openvins: A research platform for visual-inertial estimation,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4666–4672.
- [6] E. Rosten and T. Drummond, “Machine learning for high-speed corner detection,” in *European conference on computer vision*. Springer, 2006, pp. 430–443.
- [7] B. D. Lucas, T. Kanade *et al.*, “An iterative image registration technique with an application to stereo vision.” Vancouver, British Columbia, 1981.
- [8] A. I. Mourikis and S. I. Roumeliotis, “A multi-state constraint kalman filter for vision-aided inertial navigation,” in *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE, 2007, pp. 3565–3572.
- [9] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, “Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam,” *IEEE Transactions on Robotics*, 2021.
- [10] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, “Orb-slam: a versatile and accurate monocular slam system,” *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.
- [11] R. Mur-Artal and J. D. Tardós, “Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras,” *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.
- [12] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “Orb: An efficient alternative to sift or surf,” in *2011 International conference on computer vision*. Ieee, 2011, pp. 2564–2571.
- [13] M. Bloesch, M. Burri, S. Omari, M. Hutter, and R. Siegwart, “Iterated extended kalman filter based visual-inertial odometry using direct photometric feedback,” *The International Journal of Robotics Research*, vol. 36, no. 10, pp. 1053–1072, 2017.

- [14] “Github repository for vins-fusion,” <https://github.com/HKUST-Aerial-Robotics/VINS-Fusion>, accessed: 21.06.2021.
- [15] T. Qin, P. Li, and S. Shen, “Vins-mono: A robust and versatile monocular visual-inertial state estimator,” *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.
- [16] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, “Keyframe-based visual-inertial odometry using nonlinear optimization,” *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.
- [17] V. Usenko, N. Demmel, D. Schubert, J. Stückler, and D. Cremers, “Visual-inertial mapping with non-linear factor recovery,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 422–429, 2019.
- [18] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, “Kimera: an open-source library for real-time metric-semantic localization and mapping,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1689–1696.
- [19] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achterlik, and R. Siegwart, “The euroc micro aerial vehicle datasets,” *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.
- [20] T. Pire, M. Mujica, J. Civera, and E. Kofman, “The rosario dataset: Multi-sensor data for localization and mapping in agricultural environments,” *The International Journal of Robotics Research*, vol. 38, no. 6, pp. 633–641, 2019.
- [21] K. Zuiderveld, “Contrast limited adaptive histogram equalization,” *Graphics gems*, pp. 474–485, 1994.

# **Appendix A**

## **Datasheets**



## Intel® RealSense™ Tracking Camera

Datasheet

---

*Intel® RealSense™ Tracking Camera T265, Intel® RealSense™ Tracking Module T261*

*September 2019*

*Revision 004*

*Description and Features*

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at [intel.com](http://intel.com), or from the OEM or retailer.

No computer system can be absolutely secure. Intel does not assume any liability for lost or stolen data or systems or any damages resulting from such losses.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at [intel.com](http://intel.com), or from the OEM or retailer.

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or visit [www.intel.com/design/literature.htm](http://www.intel.com/design/literature.htm).

By using this document, in addition to any agreements you have with Intel, you accept the terms set forth below.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Intel, RealSense and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.

\*Other names and brands may be claimed as the property of others.

Copyright © 2019, Intel Corporation. All rights reserved.

*Description and Features*

## Contents

1	Description and Features .....	8
2	Introduction.....	9
	2.1 Purpose and Scope of this Document.....	9
	2.2 Terminology.....	9
	2.3 Tracking Camera Technology Overview.....	9
	2.4 Tracking System .....	10
	2.5 Intel® RealSense™ Tracking Module T261 .....	10
	2.6 Intel® RealSense™ Tracking Camera T265 .....	10
3	Component Specification.....	11
	3.1 Tracking Camera system Components.....	11
	3.2 Intel® RealSense™ Tracking Module T261 .....	11
	3.2.1 Inertial Measurement Unit.....	12
	3.2.2 Fisheye Imagers .....	12
	3.2.3 Tracking Module Connector Plug.....	13
	3.2.4 Tracking Module Label .....	15
	3.2.5 Stiffener .....	16
	3.2.6 Mechanical Dimensions.....	16
	3.2.7 Tracking Module Storage and Operating Conditions .....	16
	3.2.8 Tracking Module Power Requirements.....	16
	3.3 Intel® RealSense™ Tracking Camera T265 Device.....	17
	3.3.1 Intel® RealSense™ Tracking Camera T265 Mechanical Dimensions .....	17
	3.3.2 Intel® RealSense™ Tracking Camera T265 Thermals ....	17
	3.3.3 Intel® RealSense™ Tracking Camera T265 Storage and Operating Conditions .....	18
	3.3.4 Product Identifier and Material Code.....	18
4	Functional Specification .....	19



*Description and Features*

	4.1	Boot Device Information.....	19
5	Software.....	20	
	5.1	Intel® RealSense™ Software Development Kit 2.0 .....	20
6	System Integration.....	21	
	6.1	System Level Block Diagram .....	21
	6.2	Intel® RealSense™ Tracking Module T261 Module Flex.....	21
	6.3	Thermals .....	21
	6.3.1	Passive Heat Spreader.....	22
	6.4	Cover Design Guidance .....	23
	6.5	Mounting Guidance .....	24
	6.5.1	Screw Mount .....	24
	6.5.2	Occlusion Avoidance .....	25
	6.6	Gaskets.....	26
	6.6.1	Dust Protection .....	27
	6.7	Center of Tracking Location.....	27
	6.8	Tracking System Coordinate System.....	28
7	Product Regulatory .....	30	
	7.1	Manufacturer's Information .....	30
	7.2	NRTL Statement.....	30
	7.3	Ecology Compliance.....	31
	7.3.1	RoHS Declaration .....	31

## Figures

Figure 3-1.	Intel® RealSense™ Tracking Module T261 Front View .....	11
Figure 3-2.	Board to Board Receptacle Pin Map on T261 Module .....	13
Figure 3-3.	Intel® RealSense™ Tracking Camera T265.....	17
Figure 6-1.	System Block Diagram .....	21
Figure 6-2.	Passive Heat Spreader.....	23

*Description and Features*

Figure 6-3. T261 Module Screw Mount.....	24
Figure 6-4. Example for VR: HMD Positioning for Controller .....	25
Figure 6-5. Example for VR: Front Mounting Advantages .....	26
Figure 6-6. Example for VR: Top Mounting Challenges.....	26
Figure 6-7. Intel® RealSense™ Tracking Module T261 Center of Tracking Location .....	27
Figure 6-8. Intel® RealSense™ Tracking Camera T265 Center of Tracking Location .....	28
Figure 6-9. Tracking System Coordinate System.....	28
Figure 7-1. NRTL Certification .....	31

**Tables**

Table 2-1. Tracking Module Product SKU Descriptions .....	10
Table 3-1. Component Descriptions .....	11
Table 3-2. Tracking Module Properties.....	11
Table 3-3. Inertial Measurement Specifications.....	12
Table 3-4. Fisheye Image Sensor Properties .....	12
Table 3-5. Module Contact Plug Details.....	13
Table 3-6. Board to Board Connector Pin List.....	14
Table 3-7. Tracking Module Product Labeling .....	15
Table 3-8. Tracking Module Label Fields.....	15
Table 3-9. Intel® RealSense™ Tracking Module T261 Product Identifier Code and Product Material Code .....	15
Table 3-10. Intel® RealSense™ Tracking Module T261 Mechanical Dimensions.....	16
Table 3-11. Tracking Module Storage and Operating Conditions.....	16

*Description and Features*

Table 3-12. Tracking Module Power Requirements .....	17
Table 3-13. Intel® RealSense™ Tracking Camera T265 Mechanical Dimensions .....	17
Table 3-14. Max Skin Temperature.....	17
Table 3-15. Storage and Operating Conditions.....	18
Table 3-16. Product Identifier and Material Code .....	18
Table 4-1: Boot Device Vendor and Product IDs.....	19
Table 6-1: Intel® RealSense™ Tracking Module T261 – Component Power and TDP at Max Operating Mode.....	22
Table 6-2: Intel® RealSense™ Tracking Module T261 - Case Temperature Limits (Still Air) .....	22
Table 6-3: Optical Module Cover Material Parameters.....	23

*Description and Features*

## Revision History

Document Number	Revision Number	Description	Revision Date
572522	001	Initial Release	Jan 2019
	002	<ul style="list-style-type: none"><li>• Table 3-3. Inertial Measurement Specifications</li><li>• Figure 6-3. Center of Tracking Location</li><li>• Table 4-5. Boot Device Vendor and Product IDs</li><li>• 7.1 Manufacturer's Information</li><li>• 7.2 NRTL Statement</li></ul>	Mar 2019
	003	<ul style="list-style-type: none"><li>• Table 3-3. Inertial Measurement Unit</li></ul>	Apr 2019
	004	<ul style="list-style-type: none"><li>• Intel® RealSense™ Tracking Module T261</li><li>• 4.1 Boot Device Information</li><li>• Figure 6-3. T261 Module Screw Mount</li><li>• Figure 6-8. Intel® RealSense™ Tracking Camera T265 Center of Tracking Location</li></ul>	Sept 2019

§ §



*Description and Features*

## 1 *Description and Features*

<u>Description</u>	<u>Usages/Markets</u>
<p>Intel® RealSense™ Tracking Camera T265 and Intel® RealSense™ Tracking Module T261 are tracking capable devices based on visual and inertial sensor fusion. The assembly contains fisheye cameras, IMU module and a processing ASIC (Intel® Movidius™ Myriad™ 2 MA215x) with USB 3.0 interface to host processor SoC.</p>	<ul style="list-style-type: none"> <li>• Robots</li> <li>• Drones</li> <li>• Augmented Reality and Virtual Reality</li> </ul>
<u>Features</u>	<u>Minimum System Requirements</u>
<ul style="list-style-type: none"> <li>• Tracking feature using Fisheye Camera and Inertial Measurement Unit (IMU)</li> <li>• Intel® Movidius™ Myriad™ 2 ASIC</li> <li>• Middleware processed on Myriad 2 ASIC; Enabling higher CPU performance</li> <li>• 6DoF data streaming to host</li> <li>• Low latency</li> </ul>	<ul style="list-style-type: none"> <li>• Windows* 10/Linux*</li> <li>• USB 3.0 (without video streaming USB 2.0 is sufficient)</li> </ul> <p>Please check with your Intel representative for platform and OS combination supported and enablement timelines</p>

§ §

*Introduction*

## 2 *Introduction*

---

### 2.1 Purpose and Scope of this Document

This document captures the specifications for the Intel® RealSense™ Tracking Camera T265 and Intel® RealSense™ Tracking Module T261.

### 2.2 Terminology

Term	Description
6DoF	6 Degrees of Freedom refers to the freedom of movement in three dimensional space. Movement such as forward/backward, up/down, left/right, pitch, roll and yaw.
Fisheye camera	Also referred as wide angle camera
FOV	Field Of View describes the angular extent of a given scene that is imaged by a camera. A camera's FOV can be measured horizontally, vertically, or diagonally
Lens	This refers to the optical component of an imager. Its purpose is to focus the incoming light rays onto the CMOS chip in the imager.
System On Chip (SoC)	Integrated circuit (IC) that integrates all components of a computer or referred in this document as host processor SOC
Imaging or Optical module	This refers to a stiffened module containing at least two imagers. The distance between the imagers, which is referred to as the baseline or intraocular spacing, is typically in the range of 20 mm to 70 mm.
IMU	Inertial Measurement Unit
B2B	Board to Board connector
IR Cut Filter	Filter designed to prevent infrared (IR) light reaching the imagers.
TBD	To Be Determined. In the context of this document, information will be available in a later revision.

### 2.3 Tracking Camera Technology Overview

The tracking camera and module are a computer vision solution that outputs 6DoF data to the host system for immersive experience, navigation and mapping. T265/T261 uses inputs from dual fisheye cameras (OV9282) and an IMU (BMI055) along with processing capabilities from the Movidius MA215x ASIC in order to provide the host system 6DoF poses.



## 2.4 Tracking System

The T265 and T261 have one main board which includes all components on a single board. T265 comprises a T261 module, an interposer card with a USB connector and a peripheral envelope. The T261 module requires an external interposer card to allow USB connection, in case it is used outside the T265 peripheral.

## 2.5 Intel® RealSense™ Tracking Module T261

Table below describes main components that make up the module SKU

**Table 2-1. Tracking Module Product SKU Descriptions**

Component	T261
Movidius Myriad 2 ASIC	✓
Fisheye Imagers	✓
IMU	✓

## 2.6 Intel® RealSense™ Tracking Camera T265

T265 is a tracking camera peripheral including T261, interposer card and industrial design.

§ §

*Component Specification*

### 3 Component Specification

#### 3.1 Tracking Camera system Components

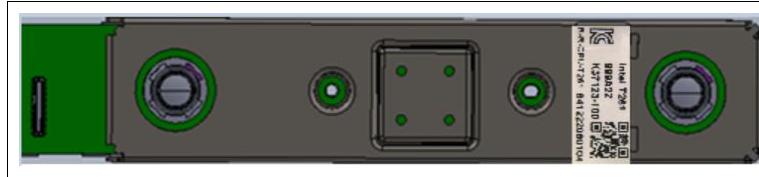
The device components are described in Table 3-1. The form factor module includes two fisheye image sensors, an Inertial Measurement Unit (IMU) and VPU processing ASIC.

**Table 3-1. Component Descriptions**

Component	Description
BMI055 IMU	Accelerometer and Gyroscope in a single package
OV9282 Fisheye Camera	Monochrome image sensor with wide field of view
Movidius MA215x	VPU Processing ASIC
Stiffener	Reinforcement housing to keep imagers aligned
Label	Manufacture and product identifier information
Other Components	IR Cut Filter, Voltage Regulators, etc.

#### 3.2 Intel® RealSense™ Tracking Module T261

**Figure 3-1. Intel® RealSense™ Tracking Module T261 Front View**



**Table 3-2. Tracking Module Properties**

Imaging Module	Intel® RealSense™ Tracking Module T261
Baseline (mm)	64±0.15
Left/Right Fisheye Imagers	OV9282
Shutter Type	Global
Fisheye FOV (degrees)	D:173
Module Dimensions (mm)	X=93.35 (+0.15 -0.25) Y=17.60±0.15 Z=7.13±0.30



*Component Specification*

H – Horizontal FOV, V – Vertical FOV, D – Diagonal FOV, X – Length, Y – Breadth, Z – Thickness

### 3.2.1 Inertial Measurement Unit

The IMU is a system-in-package for the detection of acceleration in 3 dimensions and rotations in 3 dimensions.

**Table 3-3. Inertial Measurement Specifications**

Parameter	Properties
Degrees of Freedom	6
Acceleration Range	$\pm 4g$
Accelerometer Sample Rate	62.5Hz
Gyroscope Range	$\pm 2000$ Deg/s
Gyroscope Sample Rate	200Hz

**NOTES:**

1. 6DoF pose data provided to host platform at a sample rate of 200Hz. The sample rate indicates average number of samples per second and might not imply a uniform distribution of the samples.

### 3.2.2 Fisheye Imagers

The fisheye imagers are used in the process of producing 6DoF data streamed to the host platform. The imagers provide monochrome images at 30FPS.

**Table 3-4. Fisheye Image Sensor Properties**

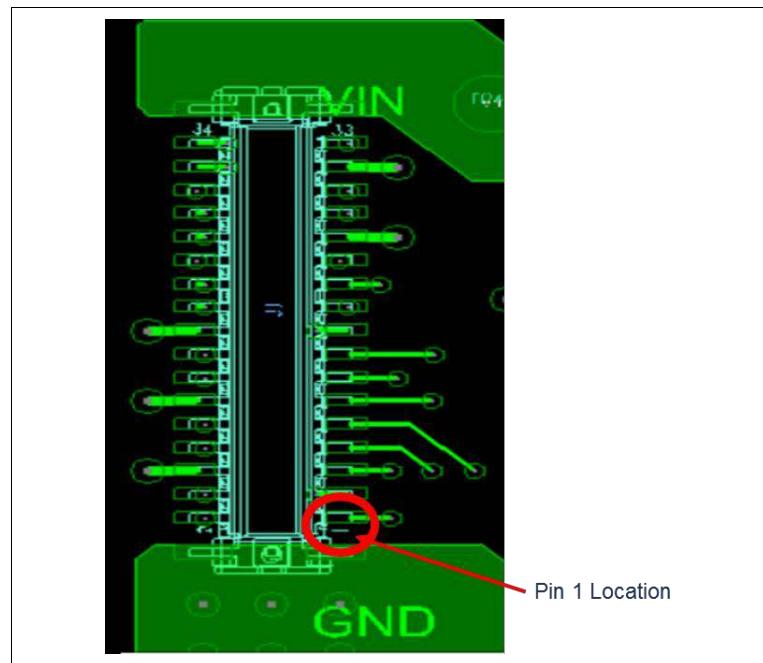
Parameter	Camera Sensor Properties
Active Pixels	848 X 800
Sensor Aspect Ratio	1.06
Format	8bit, 10-bit RAW
Filter Type	IR Cut Filter
Focus	Fixed
Shutter Type	Global Shutter
Signal Interface	MIPi CSI-2, 2 X Lanes

*Component Specification***3.2.3****Tracking Module Connector Plug**

The tracking module connector plug provides signal and power interface to the tracking module.

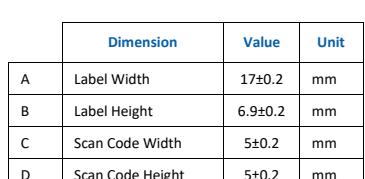
**Table 3-5. Module Contact Plug Details**

Parameter	Description	Diagram
Number of Contacts	38	
Product Name	NOVASTACK* 35-P Board-to-Board Connector	
Part Number	20708-034E-1	
Manufacturer Website	<a href="http://www.i-pex.com">www.i-pex.com</a>	

**Figure 3-2. Board to Board Receptacle Pin Map on T261 Module**

*Component Specification***Table 3-6. Board to Board Connector Pin List**

Pin Number	Description
1	Reserved
2	USB3_RX+
3	Reserved
4	USB3_RX-
5	Reserved
6	Ground
7	Reserved
8	USB3_TX+
9	Reserved
10	USB3_TX-
11	Reserved
12	Ground
13	Reserved
14	USB2+
15	Reserved
16	USB2-
17	Reserved
18	Ground
19	Reserved
20	Reserved
21	Reserved
22	Reserved
23	Reserved
24	Ground
25	Ground
26	Reserved
27	Reserved
28	Reserved
29	Reserved
30	Ground
31	Ground
32	Reserved

<i>Component Specification</i>																													
		 For illustration purpose only, subject to change																											
<b>Table 3-7. Tracking Module Product Labeling</b>																													
 <b>Scan Code Format</b> XXXXXXXXXXXXXOXXXXXXOXXXXXX-XXX																													
<b>Table 3-8. Tracking Module Label Fields</b>																													
<table border="1"> <thead> <tr> <th>Group</th><th>Field</th><th>Description</th><th>Type</th></tr> </thead> <tbody> <tr> <td>Company</td><td>Intel</td><td>Manufacturer</td><td>Static</td></tr> <tr> <td>Model Number</td><td>T261</td><td>Camera Model Number</td><td>Static</td></tr> <tr> <td rowspan="2">Product Assembly Number</td><td>XXXXXX</td><td>Product Identifier Code</td><td>Static</td></tr> <tr> <td>-XXX</td><td>Manufacture Configuration Code</td><td>Dynamic</td></tr> <tr> <td></td><td>OOOOOO</td><td>Product Material Code</td><td>Static</td></tr> <tr> <td>Serial Number</td><td>XXXXXXXXXXXX</td><td>Manufacture Unit Code</td><td>Dynamic</td></tr> </tbody> </table>			Group	Field	Description	Type	Company	Intel	Manufacturer	Static	Model Number	T261	Camera Model Number	Static	Product Assembly Number	XXXXXX	Product Identifier Code	Static	-XXX	Manufacture Configuration Code	Dynamic		OOOOOO	Product Material Code	Static	Serial Number	XXXXXXXXXXXX	Manufacture Unit Code	Dynamic
Group	Field	Description	Type																										
Company	Intel	Manufacturer	Static																										
Model Number	T261	Camera Model Number	Static																										
Product Assembly Number	XXXXXX	Product Identifier Code	Static																										
	-XXX	Manufacture Configuration Code	Dynamic																										
	OOOOOO	Product Material Code	Static																										
Serial Number	XXXXXXXXXXXX	Manufacture Unit Code	Dynamic																										
<b>Table 3-9. Intel® RealSense™ Tracking Module T261 Product Identifier Code and Product Material Code</b>																													
<table border="1"> <thead> <tr> <th>Production</th><th>Product Material Code</th></tr> </thead> <tbody> <tr> <td>Intel® RealSense™ Tracking Module T261</td><td>999AXH</td></tr> </tbody> </table>			Production	Product Material Code	Intel® RealSense™ Tracking Module T261	999AXH																							
Production	Product Material Code																												
Intel® RealSense™ Tracking Module T261	999AXH																												
572522-004		15																											



### 3.2.5 Stiffener

The stiffener maintains the precise alignment of the camera sensors and assists in subassembly rigidity. The stiffener consists of a bottom and a top plate. The stiffener is made of stainless steel grade AISI 304.

### 3.2.6 Mechanical Dimensions

**Table 3-10. Intel® RealSense™ Tracking Module T261 Mechanical Dimensions**

Dimension	Min	Nominal	Max	Unit
Width	93.1	93.35	93.5	mm
Height	17.45	17.60	17.75	mm
Depth	6.83	7.13	7.43	mm
Flatness Tolerance	-	0.25	-	mm
Weight	-	22	-	gr

### 3.2.7 Tracking Module Storage and Operating Conditions

**Table 3-11. Tracking Module Storage and Operating Conditions**

Condition	Description	Min	Max	Unit
Storage (Ambient), Not Operating	Temperature (Sustained, Controlled) <sup>(1)</sup>	0	40	°C
	Temperature (Short Exposure) <sup>(2)</sup>	-30	65	°C
	Humidity	90% RH, 30°C		
Case/Stiffener Temperature (Still Air) <sup>(3)(4)(5)</sup>	Temperature	0	55	°C

**NOTE:**

- (1) Controlled conditions should be used for long term storage of product.
- (2) Short exposure represents temporary max limits acceptable for transportation conditions.
- (3) Case temperature limits must be met for all operating temperatures.

(4) Case temperature is specified for the overall tracking module

(5) Case temperature 0° minimum and lower temperatures is non-condensing

### 3.2.8 Tracking Module Power Requirements

The tracking module is powered through USB VBUS power.

*Component Specification***Table 3-12. Tracking Module Power Requirements**

Parameter		Min	Nom	Max	Unit
VCC	Supply Voltage	4.5	5	5.25	V
ICC	Supply Current		300	300	mA

### 3.3 Intel® RealSense™ Tracking Camera T265 Device

**Figure 3-3. Intel® RealSense™ Tracking Camera T265**

#### 3.3.1 Intel® RealSense™ Tracking Camera T265 Mechanical Dimensions

**Table 3-13. Intel® RealSense™ Tracking Camera T265 Mechanical Dimensions**

Dimension	Min	Nominal	Max	Unit
Width	107.85	108.00	108.15	mm
Height	24.35	24.50	24.65	mm
Depth	12.35	12.50	12.65	mm
Flatness Tolerance	-	0.15	-	mm
Weight	57	60	63	gr

#### 3.3.2 Intel® RealSense™ Tracking Camera T265 Thermals

**Table 3-14. Max Skin Temperature**

Tracking Camera	Max Skin Temperature (25°C Ambient in Open Environment)
T265	40°C



*Component Specification*

### 3.3.3 Intel® RealSense™ Tracking Camera T265 Storage and Operating Conditions

**Table 3-15. Storage and Operating Conditions**

Condition	Description	Min	Max	Unit
Storage (Still Air), Not Operating	Temperature (Sustained, Controlled) <sup>(1)</sup>	0	40	°C
	Temperature (Short Exposure) <sup>(2)</sup>	-30	65	°C
	Humidity, Non-Condensing	90% RH, 30°C		
Operating <sup>(3)</sup> (Still Air)	Temperature	0	35	°C

**NOTES:**

1. Controlled conditions should be used for long term storage of product.
2. Short exposure represents temporary max limits acceptable for transportation conditions.
3. Component case temperature limits must be met for all operating temperatures.

### 3.3.4 Product Identifier and Material Code

**Table 3-16. Product Identifier and Material Code**

Production	Product Material Code
Intel® RealSense™ Tracking Camera T265	999AXJ

§ §

*Functional Specification*

## 4 Functional Specification

### 4.1 Boot Device Information

**Table 4-1: Boot Device Vendor and Product IDs**

Description	VID	PID
Movidius Device	03E7	2150
Intel® RealSense™ Tracking Module T261	8087	0B37
Intel® RealSense™ Tracking Camera T265	8087	0B37

1. If T261/T265 is being connected to host system via USB hub, keep in mind that T261/T265 will enter into USB enumeration protocol as soon as 5V has been provided on VBUS pin. USB protocol is handled by the Movidius MA215x device. Make sure USB hub is connected and enumerated to host system prior to power being supplied to T261/T265.
2. <1% drift observed in repeated testing in multiple use cases and environments. AR/VR use cases were tested with the T261/T265 mounted on the head in indoor living and office areas with typical indoor lighting including sunlight entering the room. Wheeled robot use cases tested with wheel odometer data integrated, in indoor office and home environments. Sufficient visibility of static tracked visual features is required, the device will not work in smoke, fog, or other conditions where the camera cannot observe visual reference points. Performance will vary across use cases and environments, the system will attempt to detect and report degraded performance but may fail to do so.

§ §

*Software*

## 5 *Software*

### 5.1 Intel® RealSense™ Software Development Kit 2.0

Intel® RealSense™ SDK 2.0 is a cross-platform library for working with Intel® RealSense™ Tracking Camera T265 and Tracking Module T261. It is open source and available on <https://github.com/IntelRealSense/librealsense>

The SDK at a minimum includes:

- **Intel® RealSense™ Viewer** - This application can be used view, record and playback depth streams, set camera configurations and other controls.
- **Debug Tools** - These command line tools gather data and generate logs to assist in debug of camera.
- **Code Examples** - Examples to demonstrate the use of SDK to include D400 Series camera code snippets into applications.
- **Wrappers** -Software wrappers supporting common programming languages and environments such as ROS, Python, Matlab, node.js, LabVIEW, OpenCV, PCL, .NET and more

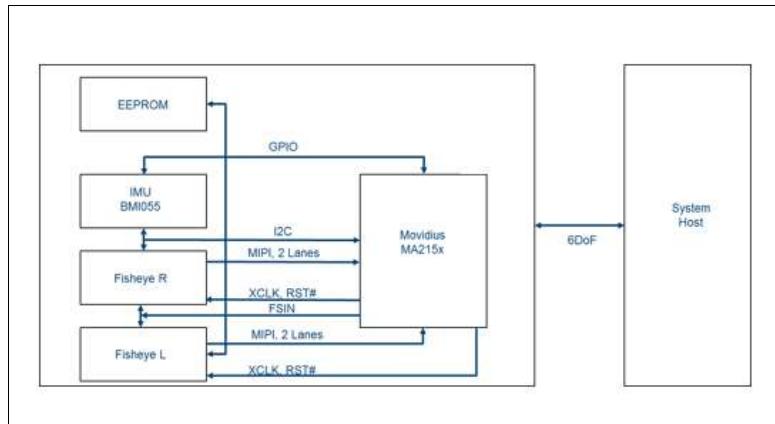
55

*System Integration*

## 6 System Integration

### 6.1 System Level Block Diagram

Figure 6-1. System Block Diagram



### 6.2 Intel® RealSense™ Tracking Module T261 Module Flex

It is critical that Intel® RealSense™ Tracking Module T261 does not experience flex during system integration or during use after integration. Micron level flexing of the module can render the calibration incorrect and will result in poor performance or nonfunctional data. It is important for system designers to isolate the module from any chassis flex the system may encounter. While the module has a reinforcement housing, the housing is not intended to counter loads from chassis flex.

### 6.3 Thermal

The system thermal design must ensure the component case temperature and system skin temperature limits are not exceeded.



*System Integration*

**Table 6-1: Intel® RealSense™ Tracking Module T261 – Component Power and TDP at Max Operating Mode**

Component	Power	TDP	Unit
ASIC	2000	2000	mW
Fisheye Camera (Left)	134	150	mW
Fisheye Camera (Right)	134	150	mW
IMU	17	17	mW
All Components	2285	2317	mW

**Table 6-2: Intel® RealSense™ Tracking Module T261 - Case Temperature Limits (Still Air)**

Component	Min	Max	Unit
ASIC	0	85	°C
Fisheye Camera (Left)	0	55	°C
Fisheye Camera (Right)	0	55	°C
IMU	0	85	°C



A thermal evaluation should be completed with the Intel® RealSense™ Tracking Module T261 to validate that the system thermal solution ensures temperature limits are not exceeded.

### 6.3.1 Passive Heat Spreader

To minimize the need or size for an internal passive heat spreader, it is recommended that a metal chassis material with a thermal conductance greater than 20 [W/mK] and an effective thermal resistance of less than 7.7 [K/W] be used. If a plastic chassis material is used, this will generally require a passive heat spreader solution. The recommended passive heat spreader solution for a plastic chassis is as follows: 100mm x 100mm x 0.2mm graphite material with plastic chassis thickness material of 1mm.

*System Integration*

**intel REALSENSE™ TECHNOLOGY**

**Figure 6-2. Passive Heat Spreader**

**6.4 Cover Design Guidance**

The T261 module components such as the fisheye camera lens must be covered to minimize dust, humidity and personal contact such as fingers to lens. All cover materials should be flat to prevent performance loss due to distortion. This is especially important for the fisheye camera sensor where distortion can significantly affect performance.

Cover materials placed over the fisheye camera sensor must be carefully selected to avoid impacting tracking performance. Distortion or reduced light sensitivity can make it difficult to track motion optically. The following recommendations should be met in cover design to support tracking performance. Other solutions can be acceptable but careful design and validation work should be done to verify a solution will perform adequately.

**Table 6-3: Optical Module Cover Material Parameters**

Specification	Recommendation	Notes
Hardness	6H (Design can be up to 9H with tempered glass)	Prevent Scratches
Flatness	0.05mm	Minimize Distortion
Minimum Gap Distance From T261 Lens to Cover Material	0.1mm (min) 0.6mm (max)	Lens height tolerance ±0.2mm should be considered
Fisheye FOV with Cover Window	170°	Keep FE FOV under all mechanical and optical design tolerances
Cover Window Coatings	Dual Side Anti-Reflect Coating	Avoid Reflections; Other coating material might have an effect on performance and should be evaluated.

572522-004

23

 <span style="float: right;"><i>System Integration</i></span>															
Total Window Transmission Wavelength Range	<table border="1"> <thead> <tr> <th colspan="2">400 – 900nm</th> </tr> <tr> <th>Incident Half Angle</th><th>Typical Trans</th></tr> </thead> <tbody> <tr> <td>0-50°</td><td>97%</td></tr> <tr> <td>60°</td><td>95%</td></tr> <tr> <td>70°</td><td>85%</td></tr> <tr> <td>80°</td><td>65%</td></tr> <tr> <td>85°</td><td>45%</td></tr> </tbody> </table>	400 – 900nm		Incident Half Angle	Typical Trans	0-50°	97%	60°	95%	70°	85%	80°	65%	85°	45%
400 – 900nm															
Incident Half Angle	Typical Trans														
0-50°	97%														
60°	95%														
70°	85%														
80°	65%														
85°	45%														
Cover Window Overall Tilt Tolerance by System Integrator	±0.5°														

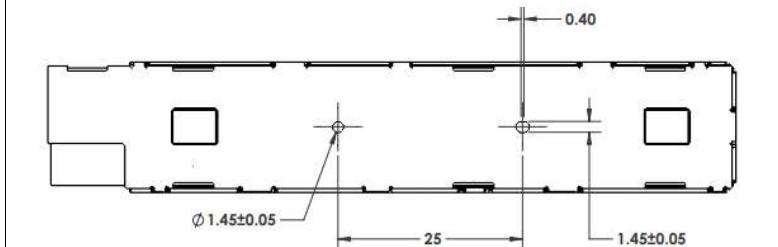
  

### 6.5 Mounting Guidance

#### 6.5.1 Screw Mount

The Intel® RealSense™ Tracking Module T261 module incorporates a screw hole for module mounting. The module should be mounted on a large heat sink or a heat dissipating structure element using M1.4 screw at the screw hole and thermal adhesive in the middle region (ex: 3M 8810). Thermal interface material should be used on backside region of ASIC and two fisheye imagers between camera module and heat sink or heat dissipating structure element for thermal transfer.

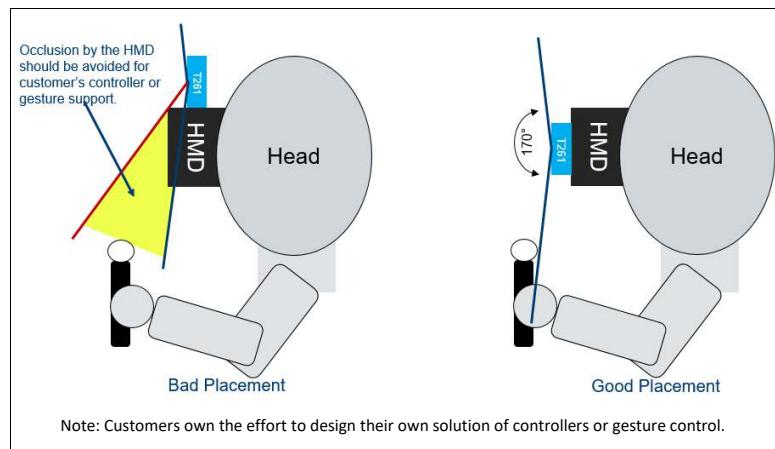
**Figure 6-3. T261 Module Screw Mount**



**Note:** Recommended torque for both M1.4 mounting points is 1.6kgf\*cm.

*System Integration***6.5.2****Occlusion Avoidance**

In order to improve quality of the 6DoF tracking capability of the Intel® RealSense™ Tracking Module T261 device, it is recommended that the mounting solution, whether integrated into HMD or used as a peripheral attachment, avoid covering, blocking or occluding the camera FOV. Below are images, depictions of possible mounting positions and mounting faults are examples for the VR segment. Other segments for T261 will have its own challenges that may not be depicted below.

**Figure 6-4. Example for VR: HMD Positioning for Controller**



System Integration

Figure 6-5. Example for VR: Front Mounting Advantages

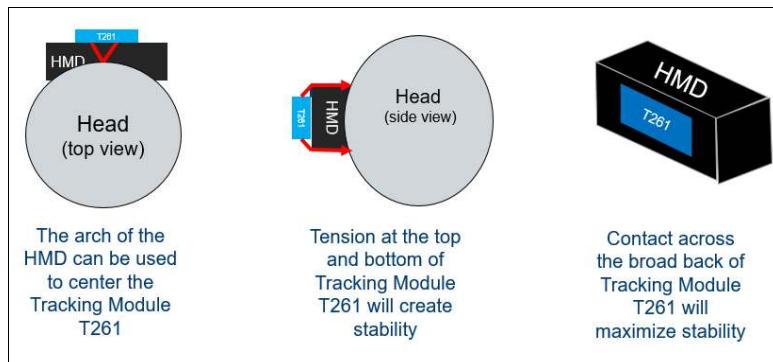
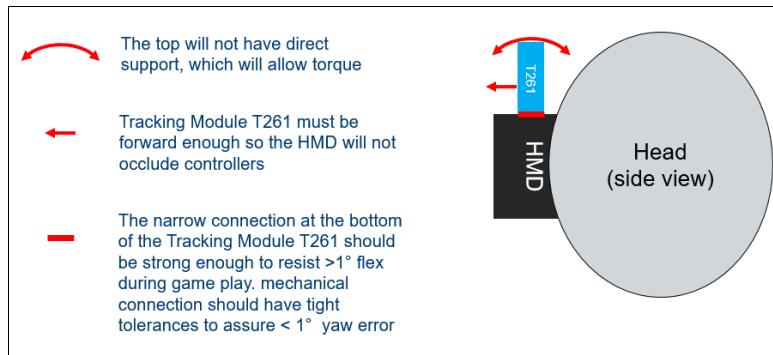


Figure 6-6. Example for VR: Top Mounting Challenges



## 6.6

### Gaskets

Gaskets are recommended for providing optical isolation and dust protection. However, gaskets can impede FOV and place unwanted stress on the module or the individual sensor lens holders.

Gasket static force can deform the cosmetic baffle/lens holder resulting in poor image quality and permanent damage to the camera. Gaskets placed on the module stiffener

*System Integration*

can transfer chassis flex into the camera module causing loss of data. Gasket thickness has a large effect on the static force applied to the module surface. The thinner the seal, the greater the static force applied. Once the gasket is compressed, the static force will increase exponentially.

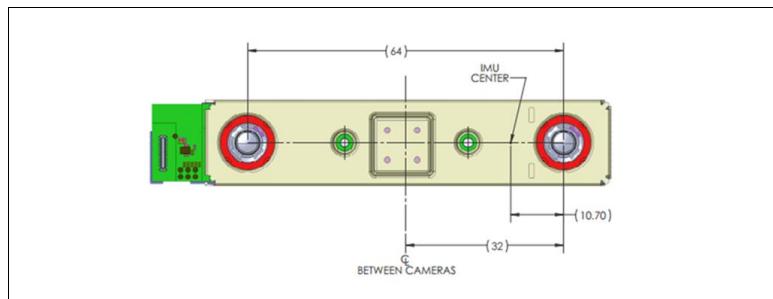
**6.6.1 Dust Protection**

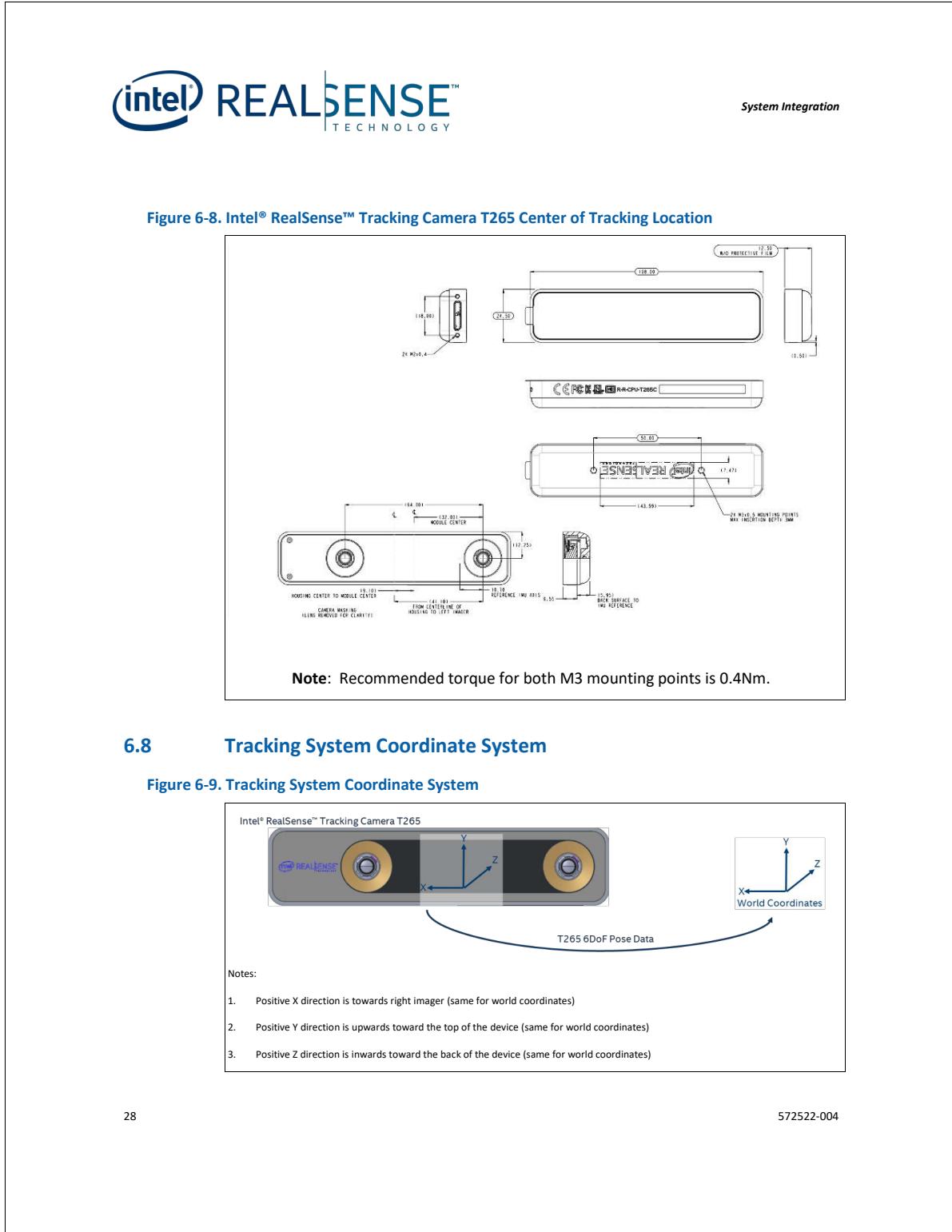
Dust particles can accumulate over the camera lenses which can be visually unappealing and degrade image quality.

**6.7 Center of Tracking Location**

The users of the tracking module and tracking camera must take into consideration the location of tracking as it pertains to the PCB inside chassis and the relationship this location has with respect to the overall system. The center of tracking corresponds to the center location between the right and left imagers on the PCB. The information in the figures below show the different mechanical specifics that help in understanding center of tracking. The center of tracking information also pertains to the coordinate system which will be discussed in future section in this document.

**Figure 6-7. Intel® RealSense™ Tracking Module T261 Center of Tracking Location**





*System Integration*



4. Coordinate system is the same for both the Intel® RealSense™ Tracking Module T261 and Intel® RealSense™ Tracking Camera T265

§ §



*Product Regulatory*

## 7 *Product Regulatory*



System integrators should refer to their respective regulatory and compliance owner to finalize regulatory requirements for a specific geography.



Do not power on the product if any external damage was observed.  
Do not try to update camera firmware that is not officially released for specific camera module SKU and revision.

### 7.1

#### **Manufacturer's Information**

Manufactured by Intel Corporation  
Attn: Corp. Quality  
2200 Mission College Blvd., Santa Clara, CA 95054 USA

EU Single Place of Contact:  
Attn: Corp Quality  
Intel Deutschland GmbH  
Am Campeon 10-12  
Neubiberg, 85579 - Germany

### 7.2

#### **NRTL Statement**

For the US and Canada market, this product has been tested and certified by Nemko, and found to be compliant with all applicable requirements of the specifications below.

UL 60950-1 2<sup>nd</sup> Edition, CAN/CSA C22.2 No. 60950-1-07, Information Technology Equipment – Safety – Part 1: General Requirements

*Product Regulatory*



Nemko is a Nationally Recognized Testing Laboratory (NRTL), recognized by US Occupational Safety and Health Administration (OSHA) as qualified to perform safety testing and certifications covered within its scope of recognition.

**Figure 7-1. NRTL Certification**

  
<https://www.nemko.com/certification/product-certification/certificates>  
Certificate #: NA201911024 (for T265 only)

---

**7.3 Ecology Compliance**

**7.3.1 RoHS Declaration**

**China RoHS Declaration**

---

产品中有毒有害物质的名称及含量  
Hazardous Substances Table

部件名称 Component Name	有毒有害物质或元素 Hazardous Substance					
	铅 Pb	汞 Hg	镉 Cd	六价铬 Cr (VI)	多溴联苯 PBB	多溴二苯醚 PBDE
相机 Camera	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
印刷电路板组件 Printed Board Assemblies	X	<input type="radio"/>				

572522-004

31



*Product Regulatory*

○：表示该有毒有害物质在该部件所有均质材料中的含量均在 GB/T 26572 标准规定的限量要求以下。

○：Indicates that this hazardous substance contained in all homogeneous materials of such component is within the limits specified in GB/T 26572.

×：表示该有毒有害物质至少在该部件的某一均质材料中的含量超出 GB/T 26572 标准规定的限量要求。

×： Indicates that the content of such hazardous substance in at least a homogeneous material of such component exceeds the limits specified in GB/T 26572.

对销售之日的所售产品，本表显示我公司供应链的电子信息产品可能包含这些物质。注意：在所售产品中可能会也可能不会含有所有所列的部件。

除非另外特别的标注，此标志为针对所涉及产品的环保使用期限标志。某些可更换的零部件可能会有一个不同的环保使用期限（例如，电池单元模块）。

此环保使用期限只适用于产品在产品手册中所规定的条件下工作。



The Environment-Friendly Use Period (EFUP) for all enclosed products and their parts are per the



In the European Union, this symbol indicates that this product including battery must not be disposed of with household waste. It is your responsibility to hand it over to a designated collection point for the recycling of waste electrical and electronic equipment. For more information, please contact your local waste collection center or the point of purchase of this product.

§ §

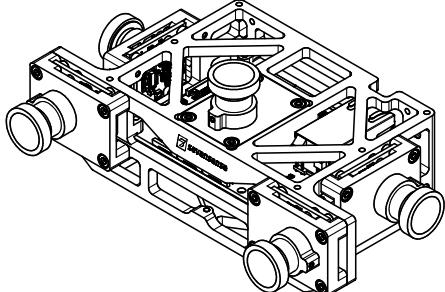


SevenSense Robotics AG  
 Weinbergstrasse 35  
 8006 Zurich, Switzerland  
[www.sevensense.ch](http://www.sevensense.ch)

# Alphasense Core Development Kit Datasheet

**SUMMARY**

Alphasense Core is a state-of-the-art visual-inertial sensor providing spatial awareness with its 360° view. It consists of up to 8 high-sensitivity cameras and a synchronized IMU. The cameras can be freely placed within a mobile robot or a sensing device. All of that to guarantee the highest quality of estimation, SLAM, local perception, semantic understanding or depth sensing.



**DEVELOPMENT KIT AT A GLANCE**

- ▶ A rigid frame with 5 global-shutter cameras equipped with state-of-the-art image sensors
- ▶ A high-performance IMU
- ▶ Precise timesync and timestamping of camera and IMU data streams
- ▶ PTP time synchronization with a host PC to seamlessly work with additional sensors
- ▶ Gigabit Ethernet interface
- ▶ Convenient test platform for the Production variant with up to 8 arbitrarily-placed cameras

## Specifications

**CAMERAS**

5 cameras, global shutter, high dynamic range, high sensitivity (Production variant: up to 8 cameras)	
Option A: Sony IMX-287 (monochrome)	Resolution: 0.4 MP Max frame rate: 75 fps (1 camera), 36 fps (8 cameras)
Option B: Sony IMX-273 (monochrome)	Resolution: 1.6 MP Max frame rate: 30 fps (1 camera), 9 fps (8 cameras)
Configurable automatic exposure control	
Camera cable length: up to 0.3 m using FFC (Production variant: up to 4.5 m using RJ50 with an adapter board)	

 www.sevensense.ch      1/3      01/04/2020

**LENS**

Opening angle (DxHxV): 165.4° x 126° x 92.4°

Focal length 2.4 mm

Infrared filter

**IMU**

Bosch BMI085 (6-axis MEMS)

Rate: 100, 200 or 400 Hz

**SIGNAL PROCESSING**

Mid-frame, exposure-compensated synchronization between IMU and camera images with an accuracy of < 100 us

PTP time synchronization with a host computer

**INTERFACE**

1000BASE-T (Gigabit) Ethernet (RJ45 connector)

IPv4 only

ROS-based device driver supporting Ubuntu 18.04 on Intel x86\_64 platforms

**POWER**

Input voltage: 6-15 V

Power consumption: 12 W with 8 cameras

Connector: Molex Nano-Fit 451300203

**MOUNTING**

Supports flexible placement of all cameras

Pre-mounted and pre-calibrated sensors with 5 cameras available

**DEVELOPMENT KIT 5-CAMERA FRAME**

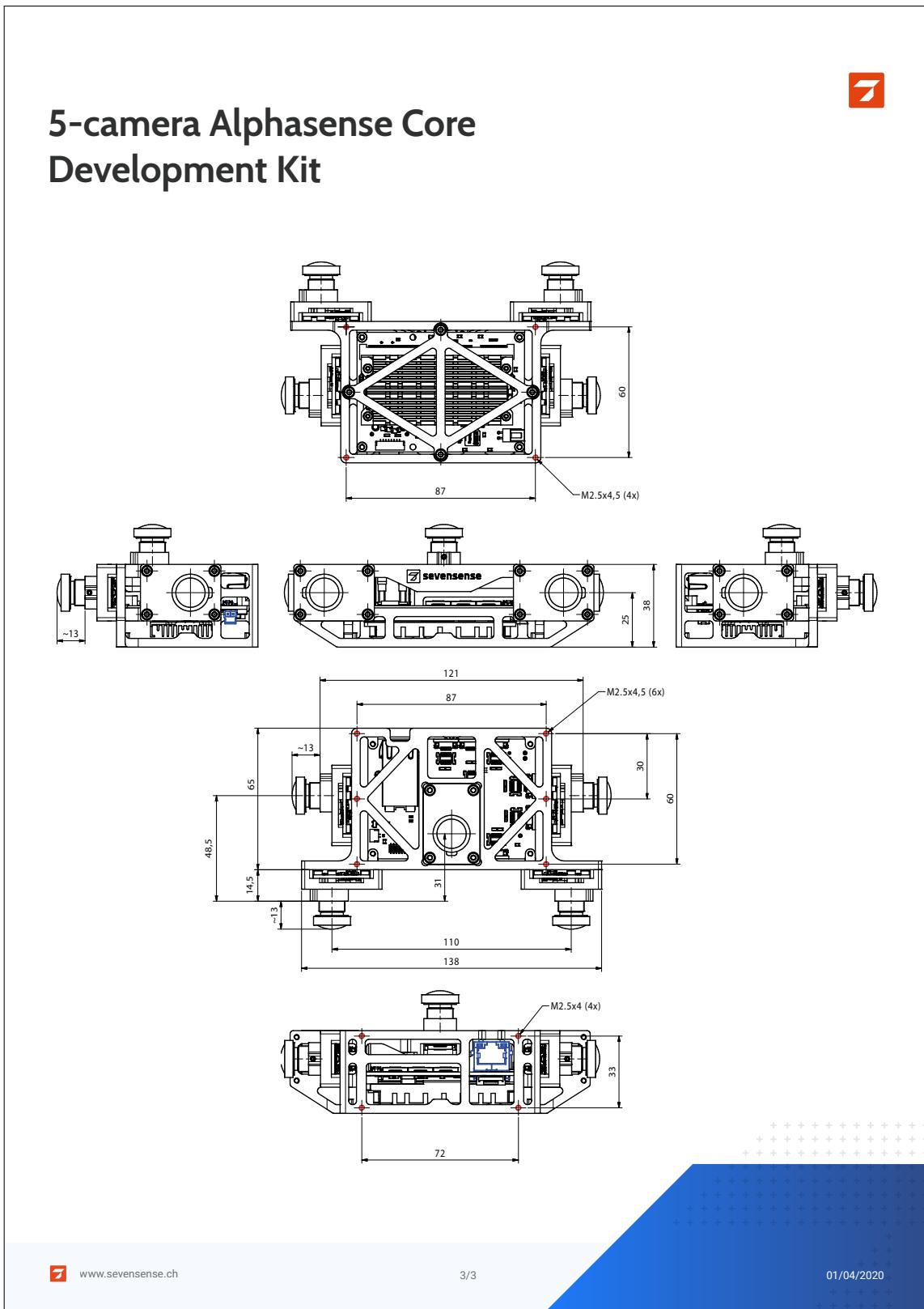
The sensor is assembled on an Aluminium or Nylon frame with 5 cameras (2 front stereo, 1 left, 1 right, 1 upwards-looking)

Mounting holes: M2.5 on top, back, bottom side

Material: Aluminium (G-Alu340) or Nylon (PA12)

Size (L x W x H): 13,8 cm x 7 cm x 4.4 cm

Weight (including sensor): 260 g (Aluminium), 175 g (PA12)



# GNSS Antenna GPS500

DUAL-BAND, HIGH PERFORMANCE MINI-SURVEY ANTENNA

## KEY FEATURES

- Supports GPS, GLONASS, Galileo, BeiDou, QZSS and SBAS signal reception
- Stable phase center guarantees the accuracy of positioning within millimeter-level
- Strong anti-interference ability to endure the harshest operating environments
- Low noise, 40 dB pre-amplifier
- 3.3 to 12 V supply voltage
- IP67 ruggedized protection

## HIGH PHASE CENTER STABILITY

GPS500 features a multi-point feeding design to achieve greater phase center stability. It effectively improves measurement accuracy and provides a better RTK solution. GPS500 has passed the international NGS certificate test.

## TRACKING IN CHALLENGING ENVIRONMENTS

The ability to receive low elevation signals with high gain and wide beam width makes GPS500 an excellent choice for tracking visible satellites under challenging conditions, providing the positioning solutions with precision and reliable data. It can be widely used in autonomous vehicles, construction, agricultural equipment, field base stations and GIS surveying where high-precision operations are needed.

## STRONG ANTI-INTERFERENCE PERFORMANCE

The antenna LNA features an excellent out-of-band rejection performance, which can suppress the electromagnetic interference, to provide stability and reliability of GNSS signals. It also effectively avoids the dangers of disconnection when receivers are operated under complex electromagnetic environments such as a communication tower and in busy urban areas.

## SMALL FORM FACTOR

This lightweight and small sized antenna can lighten the loads of small unmanned vehicles. A standard TNC female connector provides reliable and easy integration. Its IP67 ruggedized design protects it from dust and water.



## TECHNICAL SPECIFICATIONS

### Performance

Signals Received	
GPS	L1/L2
GLONASS	L1/L2
Galileo	E1/E5b[1192-1229.64 MHz]
BeiDou	B1/B2/ B3
QZSS	L1/L2
SBAS	L1
<b>Nominal Impedance</b>	50 ohm
<b>Polarization</b>	RHCP
<b>Axial Ratio</b>	≤ 3 dB
<b>Gain at Zenith</b>	
1205-1278 MHz	5.5 dBi [maximum]
1559-1615 MHz	5.5 dBi [maximum]
<b>LNA Gain</b>	40 dB [typical]
<b>Noise Figure</b>	≤ 2.0 dB
<b>Output/Input VSWR</b>	≤ 2.0
<b>Operation Voltage</b>	+3.3 to +12V DC
<b>Operation Current</b>	45 mA [maximum]
<b>Group Delay Ripple</b>	≤ 5 ns

### Mechanical

Dimension	φ 152*62.2 mm
Connector	TNC Female
Weight	440 g
Mounting	BSW5/8"-11 screw, 12-14 mm

### Environmental

<b>Temperature</b>	
Storage	-55 °C to +85 °C
Operating	-40 °C to +85 °C
<b>Humidity</b>	95% non-condensing
<b>Water/Dust Resistance</b>	IP67

Compliance FCC CE RoHS REACH

[www.swiftnav.com](http://www.swiftnav.com) | @SwiftNav | [sales@swiftnav.com](mailto:sales@swiftnav.com)

©2020 Swift Navigation, Inc. All rights reserved | 01.17.2020

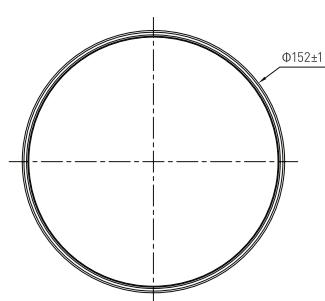


# GNSS Antenna GPS500

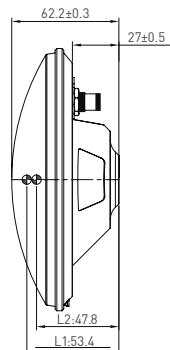
DUAL-BAND, HIGH PERFORMANCE MINI-SURVEY ANTENNA



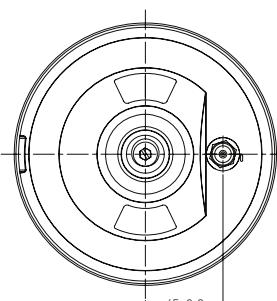
ANTENNA STRUCTURE AND PHASE CENTERS DRAWING  
(DIMENSIONS IN MM)



TOP VIEW



SIDE VIEW



BOTTOM VIEW

Undeclared tolerance: $\pm 0.2\text{mm}$

Specifications subject to change without notice.