

## TriSpector 1060 Evaluation

This document gives our evaluation of the TriSpector 1060. It is based on evaluation of the sensor performed by Stiftelsen Adopticum in a project financed by Kempestiftelserna. For more information about the sensor, please feel free to contact Adopticum.

---

### Introduction

This is a document summarizing properties of the TriSpector 1060 laser triangulation smart sensor. The manual for the sensor includes technical data and information (see Software below).

This document, however, focuses on information acquired from our testing with a sensor available and analysing properties of interest. The goal is to make it clearer what kind of measurement cases this sensor would be useful for.

### Equipment

The equipment used during testing was:

- TriSpector 1060 (to be evaluated), with power and connector cables from Sick AB.
- Adopticum's conveyor belt.
- Incremental encoder (attached to the conveyor belt). The conveyor belt moving distance and encoder ticks are related as: 0.015152 mm/ticks.
- Test objects for measurement accuracy testing:
  - o Checkerboard:
    - Square\_size: 23 mm x 23 mm
    - Number of square: 10\*7
  - o Lego shape (see Figure 1 below).
- Laptop with USB-LAN-port-cable.

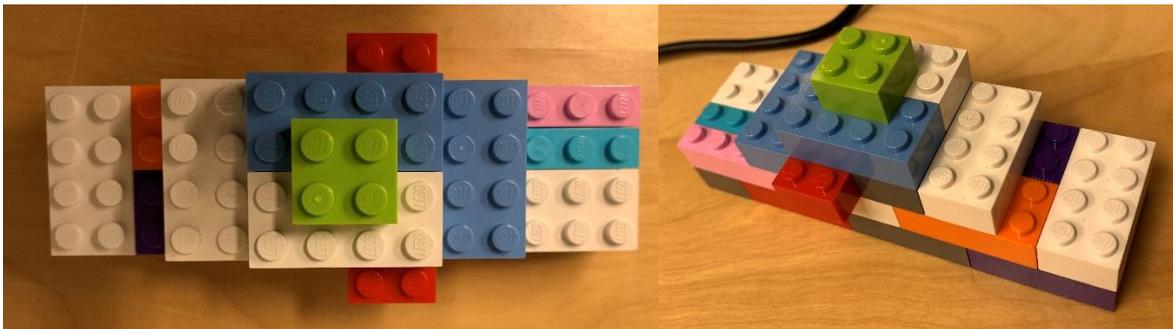


Figure 1: Lego shape used for creating a specific shape with building blocks with a set standardized size.

# ADOPTICUM

## Software

A web user interface (Chrome is recommended) was used for communication with the TriSpector. Communication is done through Ethernet.

Startup steps (Windows 10):

NOTE: You need a license to use AppStudio and develop applications for TriSpector.

Configure the network

- Go to Control Panel -> Network and Sharing Center
- Click on the desired network connection.
- Go to Properties.
- Choose Internet Protocol Version 4 (TCP/IPv4) and click on Properties.
- Choose the option *Use the following IP address*.
- Set the following:
  - o IP Address: 192.168.0.10 (or an address on the same network as the sensor, but not the same last number)
  - o Subnet Mask: 255.255.255.0
- Click OK.

Run the SICK AppManager and:

- Click the Search-button under the Device search tab.
- Click the camera with port 2122.
- The applications you want to run can be started by right-clicking on the correct app in the Device tab in the lower left corner and starting it/them.

NOTE: The apps can also be started by running them in SICK AppStudio (see separate manual for this).

Then start the web interface:

- Open a web browser.
- Enter the sensor address <http://192.168.0.10>

Start the sensor:

- **WARNING:** Make sure that the sensor is aimed so that no one can accidentally look into the laser light source. If laser class is above class 2, please contact your laser safety officer.
- Depending on the application used the sensor GUI will be different. *ImageAcquisition* is a standard example to get started where the camera parameters like exposure time, sensor resolution and plane above which the data is to be used.

## Sensor settings

Settings configured from the web GUI can be saved to file and loaded into the sensor at another time. This allows for a quick up-and-running setup for a new test session, a new measurement system installation or when a TriSpector needs to be replaced. Using the TriSpector API, images can be written to JSON-files to run them again, using an emulator for later analysis.

## Sensor properties of note

Data format:

Data can be written to JSON-files with the following format:

class = Image

data =

scalar structure containing the fields:

PixelSize =

scalar structure containing the fields:

x = 0.5034

y = 0.5333

z = 0.040000

Origin =

scalar structure containing the fields:

x = -188.18

y = -74.667

z = 0

HandleZeroPixels = MissingData

Data =

scalar structure containing the fields:

ImageType = uint16

Width = 800

Height = 703

Pixels =

scalar structure containing the fields:

numOfElems: 1x1 scalar

# ADOPTICUM

elemSz: 1x1 scalar

endian: 1x3 sq\_string

elemTypes: 1x1 cell

data: 1x1499736 sq\_string

Where Image.data.Data.Pixels.data contains the values, but needs the right endian and data type to be read properly.

Files have been written to read the image files:

- Read the JSON files and write files for Octave to read: **read\_json.py**
- Read and plot intensity and XYZ-data from files converted from **read\_json.py**:  
**Read\_Trispector\_json\_files.m**

## Accuracy

The accuracy and resolution of the sensor can be found in data sheets and have not been analysed extensively. However, measurements have been performed on test objects and the accuracy is deemed to allow for sub-mm-precision, for the depth (Z). X can also measure with sub-mm-precision, but it depends on the distance between the sensor and the objects. Y-precision depends on the relation between encoder ticks and distance travelled by the conveyor, as well as the settings of the camera that affect the profile rate (see Sensor speed below). 0.3 mm per profile (Y) has been achieved during tests.

## Sensor speed:

The number of profiles per second is set in a camera settings parameter. If an encoder is used the exposure time needs to be short enough that the encoder doesn't trig the camera for measurements too often. The X-resolution (horizontal resolution) can also affect the speed of the camera.

## Encoder:

Using an incremental encoder with A-B-ports and TTL-signals, then converted to a differential RS-485 signal, the TriSpector could be triggered using the movement of a conveyor belt. Using an encoder as a trigger is simple using the web interface. Simply choose how often, in mm, you want to read the profile measurements. The thing to keep in mind is that if the sensor is triggered too often for the sensor speed to keep up the profile data might include empty profiles.

The encoder ticks to mm-conversion also needs to be handled for accurate measurements. The mm/ticks can be set manually if this is known but it might require test measurements on a well-defined object of known size.

## Exposure to use:

# ADOPTICUM

The sensor can use the strongest, bottom, or top reflections for the laser line for cases where more than one reflection might return to the sensor.

## Prices

The price range for TriSpector-sensors for laser triangulation is about 86 000 SEK for a TriSpector 1060. With other components needed to use the sensor for conveyor belt measurements and an AppSpace license the cost is around 150 000 SEK (with no discount).

## User cases

- High speed measurement of objects on a limited surface/conveyor belt. Accurate measurements on a relatively small area but with the possibility of extending this area with aligning several TriSpectors or moving one sensor using mobile robot-parts, although this makes a system that is more expensive and possibly more complicated.
- Close to speed of 5000 profiles/second is possible using the correct settings. Using an encoder makes this harder to reach than using free-running mode.
- Using the TriSpector API several image processing tools are included for use when writing applications using SICK AppSpace. Some of these are: Thresholding images, find blobs with basic measurements (bounding box, object area, etc.), inverse images, etc. Also, 3D point cloud data objects can be treated.
- From the SICK AppSpace applications already develop can be used (bought) for specific tasks, but it is also possible to add plugins to applications that you can develop yourself using AppStudio.

## Summary

The TriSpector sensors combine laser and camera within one housing pre-calibrated, allowing for a simpler setup to get ready to use the sensor. It also has a lot of functionality that can be used with the sensor alone without an external PC.

However, to use the TriSpector 1060 sensor some programming is needed to be done to use the sensor for measuring. This means that more work needs to be done to get started but using the extensive API, applications can be written tailor-made for many different specific measurement tasks. Alternatively, already existing applications from SICK AppSpace can be used for specific tasks. However, if these AppSpace-applications need to be adjusted somehow, plugins might be needed which you need AppStudio for developing.

The sensors allow for laser triangulation measurements at high frequency, almost at 5000 profiles/second for the 1060 model. This depends on other settings for the sensor but it is able to deliver measurements at a high frequency.

Regarding the cost, the sensors' prices at 86 000 SEK or higher for ONE sensor is high if the built-in tools, one of the big advantages of a smart-sensor like this, will not be used and the acquiring of raw profiles is what it is to be used for. There are other sensors that can be used for this that are a bit cheaper. And if a TriSpector's laser emitter burns out and the whole sensor needs to be replaced, it means that the replacement costs will be higher than for other systems where the light source and the camera are separated.

The price for AppStudio is about 36 700 SEK, which is needed for developing your own applications or plugins for existing applications bought from SICK AppSpace. Which means an additional cost.

The TriSpector 1060-sensor can thus be a good and time-saving solution when an application is already available for the measurement need. Otherwise, it can be a very flexible solution, where the build-in toolset can be used as the user sees fit, but will require actual development of a specific application to solve the measurement problem. This means that coding skills will be required to work with the sensor, which must be found either in-house or by hiring qualified help. SICK can help with at least adjust an existing application of theirs to the customers that ask for it, but it is up to the customer to take responsibility for the application afterward.