

SmartRunner Explorer

Lichtschnittsensor zur hochgenauen Profilerfassung

Handbuch



Es gelten die Allgemeinen Lieferbedingungen für Erzeugnisse und Leistungen der Elektroindustrie, herausgegeben vom Zentralverband Elektroindustrie (ZVEI) e. V. in ihrer neuesten Fassung sowie die Ergänzungsklausel: "Erweiterter Eigentumsvorbehalt".

Weltweit

Pepperl+Fuchs-Gruppe

Lilienthalstr. 200

68307 Mannheim

Deutschland

Telefon: +49 621 776 - 0

E-Mail: info@de.pepperl-fuchs.com

<https://www.pepperl-fuchs.com>

1	Einleitung	5
1.1	Inhalt des Dokuments	5
1.2	Zielgruppe, Personal	5
1.3	Verwendete Symbole.....	6
2	Produktspezifikationen	7
2.1	Einsatz und Anwendung	7
2.2	Gefahren durch Laserstrahlung	8
2.3	Abmessungen	9
2.4	Anzeige- und Bedienelemente	10
2.5	Schnittstellen und Anschlüsse	12
2.6	Netzwerkschnittstelle.....	13
2.7	Softwareschnittstelle.....	14
2.8	Zubehör	15
2.8.1	Spannungsversorgung	15
2.8.2	Netzwerkkabel.....	16
3	Installation.....	17
3.1	Lagerung und Entsorgung.....	17
3.2	Vorbereitung.....	17
3.3	Erfassungsbereich	18
3.4	Montage des Sensors	19
3.5	Anschluss des Sensors	21
3.6	Windows Netzwerkkommunikation Sensor-PC/Laptop einrichten.....	23
4	Inbetriebnahme.....	26
5	Konfiguration	27
5.1	VsxProtocolDriver.....	27
5.2	Konfigurationsparameter.....	41
6	Software Vision Configurator	47
6.1	Verbinden mit dem Vision Configurator	48
6.2	Aufbau des Anwendungsfensters.....	50

6.3	Menüleiste	52
6.3.1	Menü File	52
6.3.2	Menü View	52
6.3.3	Menü Sensor	53
6.3.4	Menü Image	54
6.3.5	Menü Administration	54
6.3.6	Menü Help	55
6.4	Symbolleiste	56
6.5	Gerätedaten	57
6.6	Bild- und Linienanzeige	57
6.7	Parametrierbereich	63
6.7.1	Registerkarte Sensorinformation (Sensor information)	63
6.7.2	Registerkarte Common	64
6.7.3	Registerkarte Explorer	66
7	Wartung und Reparatur	67
7.1	Wartung	67
7.2	Reparatur	67
8	Störungsbeseitigung	68
8.1	Was tun im Fehlerfall	68

1 Einleitung

1.1 Inhalt des Dokuments

Dieses Dokument beinhaltet Informationen, die Sie für den Einsatz Ihres Produkts in den zutreffenden Phasen des Produktlebenszyklus benötigen. Dazu können zählen:

- Produktidentifizierung
- Lieferung, Transport und Lagerung
- Montage und Installation
- Inbetriebnahme und Betrieb
- Instandhaltung und Reparatur
- Störungsbeseitigung
- Demontage
- Entsorgung



Hinweis!

Entnehmen Sie die vollständigen Informationen zum Produkt der weiteren Dokumentation im Internet unter www.pepperl-fuchs.com.

Die Dokumentation besteht aus folgenden Teilen:

- vorliegendes Dokument
- Datenblatt

Zusätzlich kann die Dokumentation aus folgenden Teilen bestehen, falls zutreffend:

- EU-Baumusterprüfbescheinigung
- EU-Konformitätserklärung
- Konformitätsbescheinigung
- Zertifikate
- Control Drawings
- Betriebsanleitung
- weitere Dokumente

1.2 Zielgruppe, Personal

Die Verantwortung hinsichtlich Planung, Montage, Inbetriebnahme, Betrieb, Instandhaltung und Demontage liegt beim Anlagenbetreiber.

Nur Fachpersonal darf die Montage, Inbetriebnahme, Betrieb, Instandhaltung und Demontage des Produkts durchführen. Das Fachpersonal muss die Betriebsanleitung und die weitere Dokumentation gelesen und verstanden haben.

Machen Sie sich vor Verwendung mit dem Gerät vertraut. Lesen Sie das Dokument sorgfältig.

1.3 Verwendete Symbole

Dieses Dokument enthält Symbole zur Kennzeichnung von Warnhinweisen und von informativen Hinweisen.

Warnhinweise

Sie finden Warnhinweise immer dann, wenn von Ihren Handlungen Gefahren ausgehen können. Beachten Sie unbedingt diese Warnhinweise zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden.

Je nach Risikostufe werden die Warnhinweise in absteigender Reihenfolge wie folgt dargestellt:



Gefahr!

Dieses Symbol warnt Sie vor einer unmittelbar drohenden Gefahr.

Falls Sie diesen Warnhinweis nicht beachten, drohen Personenschäden bis hin zum Tod.



Warnung!

Dieses Symbol warnt Sie vor einer möglichen Störung oder Gefahr.

Falls Sie diesen Warnhinweis nicht beachten, können Personenschäden oder schwerste Sachschäden drohen.



Vorsicht!

Dieses Symbol warnt Sie vor einer möglichen Störung.

Falls Sie diesen Warnhinweis nicht beachten, können das Produkt oder daran angeschlossene Systeme und Anlagen gestört werden oder vollständig ausfallen.

Informative Hinweise



Hinweis!

Dieses Symbol macht auf eine wichtige Information aufmerksam.



Handlungsanweisung

Dieses Symbol markiert eine Handlungsanweisung. Sie werden zu einer Handlung oder Handlungsfolge aufgefordert.

2 Produktspezifikationen

2.1 Einsatz und Anwendung

Dieses Handbuch ist gültig für die Lichtschnittsensoren SmartRunner Explorer (im Nachfolgenden Sensor genannt). Der Sensor basiert auf der SmartRunner-Technologie und vereint das Lichtschnittverfahren zur Erfassung von Höhenprofilen mit einem 2-D-Vision-Sensor. Basierend auf dieser einmaligen Technologie kann der Sensor sowohl Höhenprofile in Weltkoordinaten als auch 2-D-Flächenbilder ausgeben.

Beim Lichtschnittverfahren wird mit einer Sendeoptik eine Laserlinie auf ein Objekt projiziert. Diese wird in einem bestimmten Winkel von einer Kamera erfasst. Anschließend wird über das Triangulationsprinzip ein Höhenprofil ermittelt. Dieses kann dann über die Ethernet-TCP/IP-Schnittstelle ausgegeben und mittels der mitgelieferten Programmierschnittstelle komfortabel in das jeweilige Programm eingebunden werden.

Der SmartRunner Explorer ermöglicht damit die flexible Umsetzung vieler Applikationen.

Aufbau des Sensors

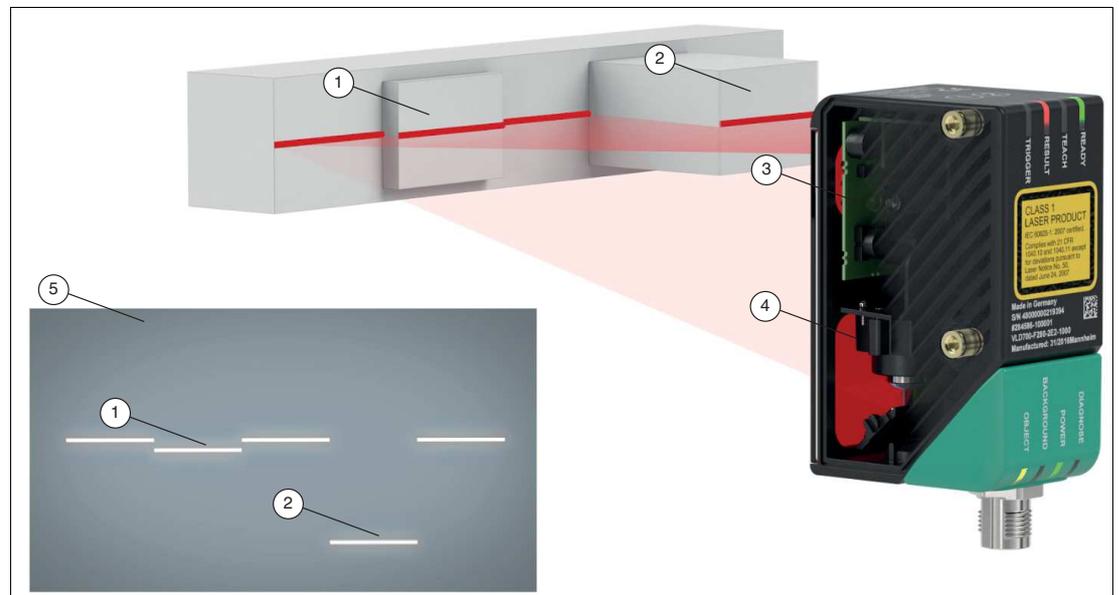


Abbildung 2.1 Übersicht Komponenten und Messergebnis

- 1 Flaches Profil
- 2 Erhöhtes Profil
- 3 Sendeinrichtung (Laser für Höhenprofil und LED's für die Flächenbeleuchtung)
- 4 Kamera
- 5 Höhenprofil auf Bildaufnehmer (Messergebnis)

2.2 Gefahren durch Laserstrahlung

In diesem Abschnitt werden der Inhalt und die Befestigungsposition des Warnaufklebers erläutert.

Der eingesetzte Sensor entspricht der Sicherheitsnorm IEC 60825-1:2007 für ein Produkt der Laserklasse 1. Des Weiteren wird die US-Regulierung 21 CFR 1040.10 und 1040.11 mit den Abweichungen der **Laser Notice No. 50** vom 24. Juni 2007 eingehalten.



Warnung!

Laserstrahlung der Klasse 1

Die Bestrahlung kann zu Irritationen gerade bei dunkler Umgebung führen. Nicht auf Menschen richten!

Niemals in die Laseraustrittsöffnung blicken, wenn der Sensor in Betrieb ist.

Wartung und Reparaturen nur von autorisiertem Servicepersonal durchführen lassen!

Das Gerät ist so anzubringen, dass der Warnhinweise deutlich sichtbar und lesbar ist.

Schutzabdeckung des Sensors nicht entfernen.

Der Warnaufkleber ist auf der Rückseite des Gehäuses gemäß folgender Abbildung angebracht.

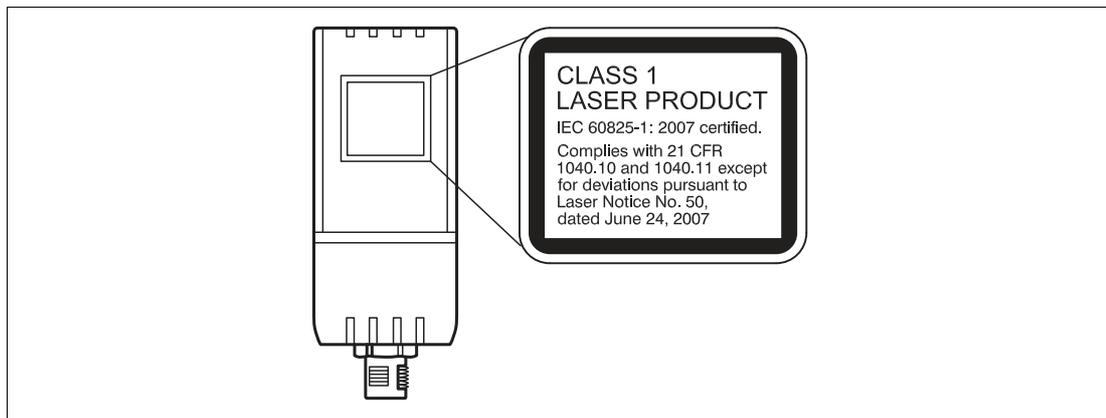


Abbildung 2.2 Warnhinweis Laserstrahlung

2.3 Abmessungen

Die Geräte der SmartRunner-Serie haben folgende identische Gehäusemaße.

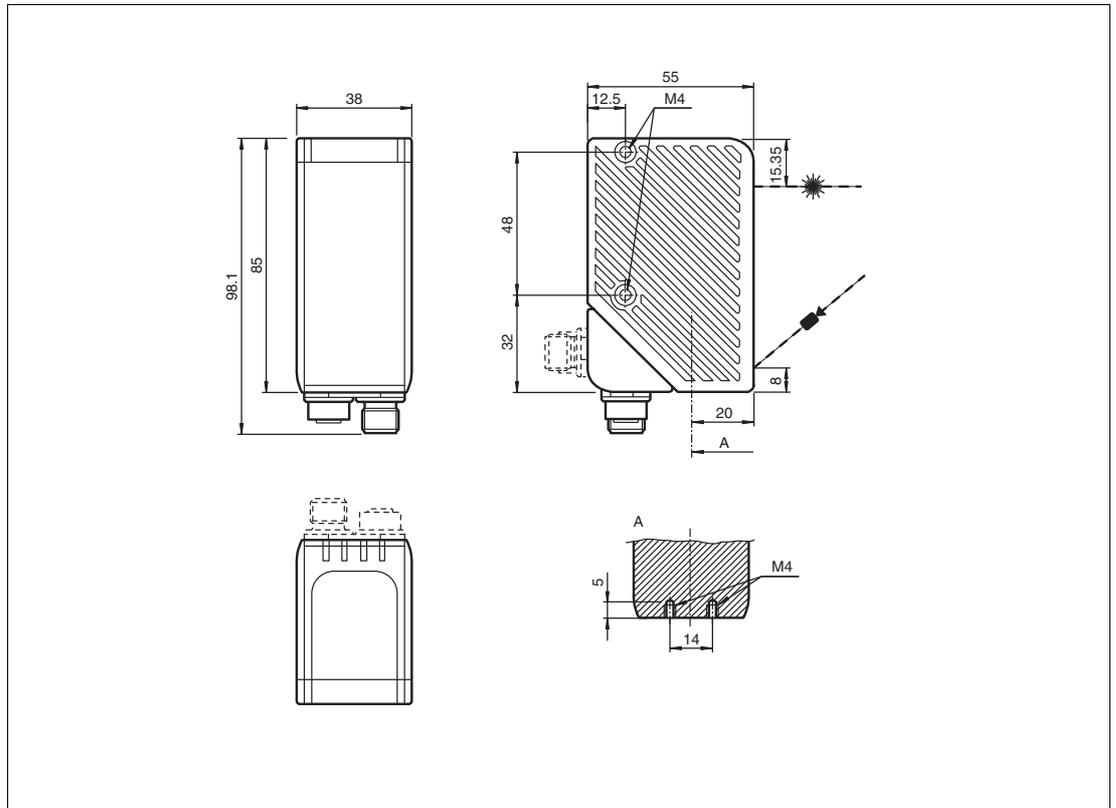


Abbildung 2.3 Abmessungen der SmartRunner-Serie

2.4 Anzeige- und Bedienelemente

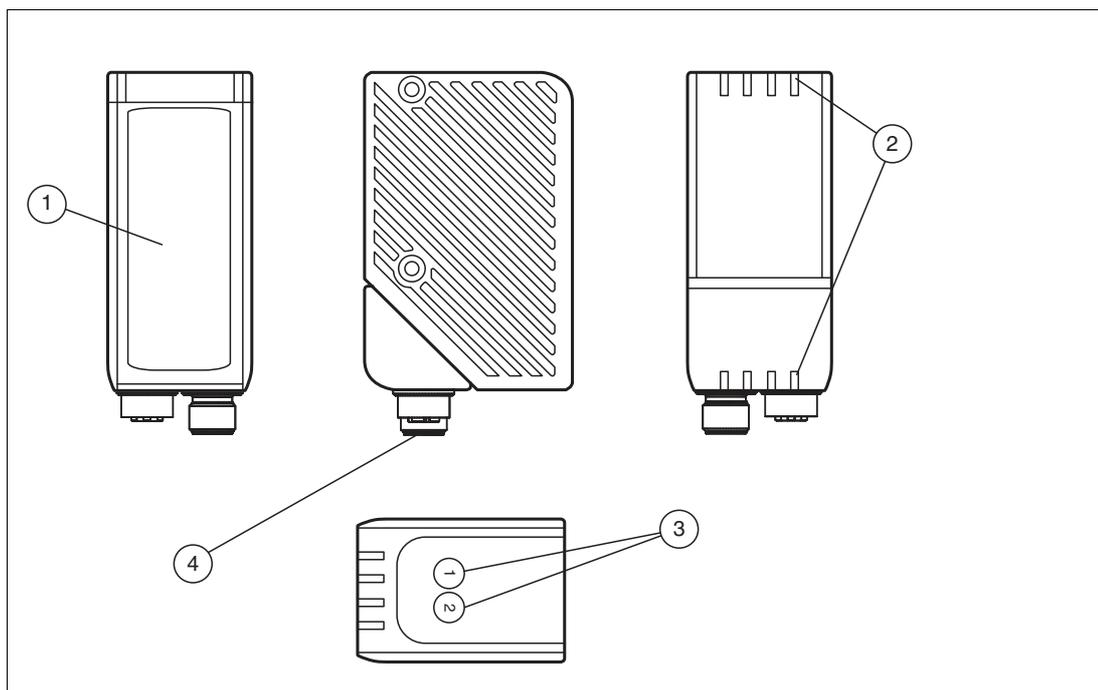


Abbildung 2.4 Übersicht Anzeige- und Bedienelemente

Position	Bezeichnung	Funktion
1	Schutzabdeckung Sendeoptik	Dient zum Schutz vor Beschädigung und Verschmutzung
2	LEDs	Die Funktionsbeschreibung der LEDs entnehmen Sie der nachfolgenden Tabelle.
3	Funktionstasten im Präsentationsmodus	<ul style="list-style-type: none"> Funktionstaste 1: Auslösen einer Auswertung Funktionstaste 2: Kürzer 2 s gedrückt aktiviert den Einlernvorgang Teach In. Länger als 2 s gedrückt aktiviert den Codekartenmodus
	Funktionstasten im Runtime-Modus	<ul style="list-style-type: none"> Funktionstaste 1: keine Funktion Funktionstaste 2: Länger als 2 s gedrückt aktiviert den Codekartenmodus
4	Anschlüsse	24 V DC + IO (Spannungsversorgung, Eingänge und Ausgänge) LAN (Netzwerk)

Beschreibung der LEDs

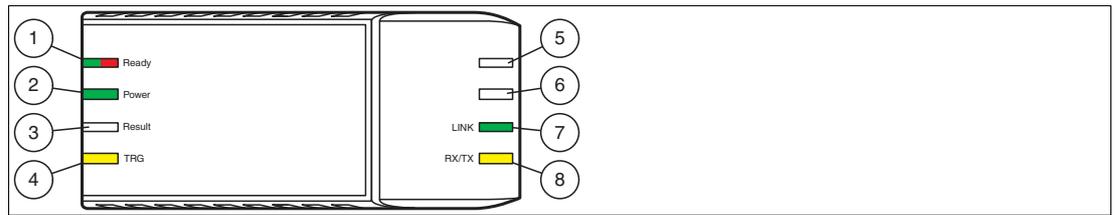


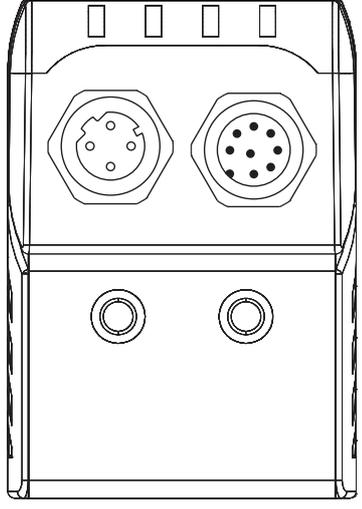
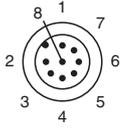
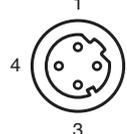
Abbildung 2.5 Übersicht LEDs

LED	Bezeichnung	Farbe	Zustand	Bedeutung
1	READY	(rot)	Ein	Leuchtet rot, wenn eine Sensorstörung vorliegt und nach einem Update. Blinkt rot, während des Updatevorgangs. HINWEIS: Unterbrechen Sie nicht die Stromversorgung während des Updates!
		(grün)	Ein	Leuchtet grün, wenn der Sensor betriebsbereit ist.
		(grün)	Blinkt	Blinkt grün, wenn der Sensor im Konfigurationsmodus ist.
2	POWER	(grün)	Ein	Leuchtet auf, sobald Spannung anliegt.
3	RESULT	-	-	Reserviert
4	TRIGGER	(gelb)	Ein	Leuchtet gelb, wenn das Hardware-Triggersignal angesteuert wird.
5	-	-	-	-
6	-	-	-	-
7	LINK	(grün)	Ein	Leuchtet grün, der Sensor hat eine Verbindung zum Ethernet.
		(gelb)	Ein	Leuchtet gelb, der Sensor hat keine Verbindung zum Ethernet.
8	RX/TX	(gelb)	Flackert	Flackert gelb, wenn Datenaustausch aktiv ist.

2.5 Schnittstellen und Anschlüsse

Folgende Anschlüsse befinden sich am Sensor.

Anschlussbelegung

	24 V DC + IO (Spannungsversorgung, Eingänge und Ausgänge)	
	M12-Stecker, A-kodiert, 8-polig	
		<ul style="list-style-type: none"> 1. IN Trigger 2. + UB 3. nc 4. nc 5. nc 6. nc 7. GND
	LAN (Netzwerk)	
M12-Buchse, D-kodiert, 4-polig		
	<ul style="list-style-type: none"> 1. TX+ Ethernet 2. RX+ Ethernet 3. TX- Ethernet 4. TX- Ethernet 	



Tipp

Die Gehäuseecke mit den Anschlüssen ist drehbar. Um eine einfache Verkabelung zu gewährleisten, können Sie je nach Montageposition den Gerätestecker in eine andere Richtung drehen.

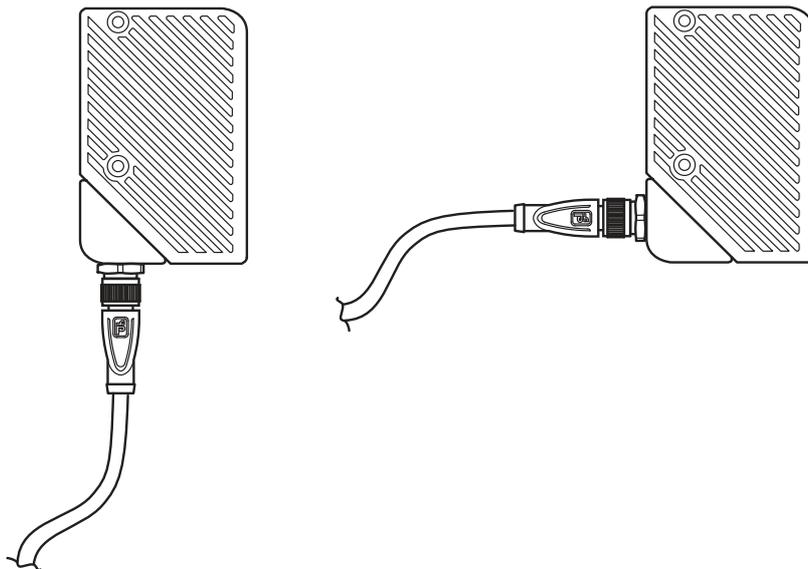


Abbildung 2.6 M12-Gerätestecker

2.6 Netzwerkschnittstelle

Die Netzwerkschnittstelle überträgt folgende Daten.

- Steuerbefehle (Start, Stopp, Betriebsart)
- Firmware-Updates (Datenübertragung vom Host zum Sensor)
- Ergebnisdaten wie z. B. Scans und Bilder (Datenübertragung vom Sensor zum Host)

Die Übertragung von Parametern und Statusabfragen erfolgt in Form von Strings, die XML eingebettet sind. Die Übertragung von Daten (Scans, Bilder) und Firmware erfolgt in Form von binären Daten.

Die Kommunikation erfolgt mittels TCP/IP über Port 50005.

Die werksseitige Voreinstellung der IP-Adresse lautet 192.168.2.3.

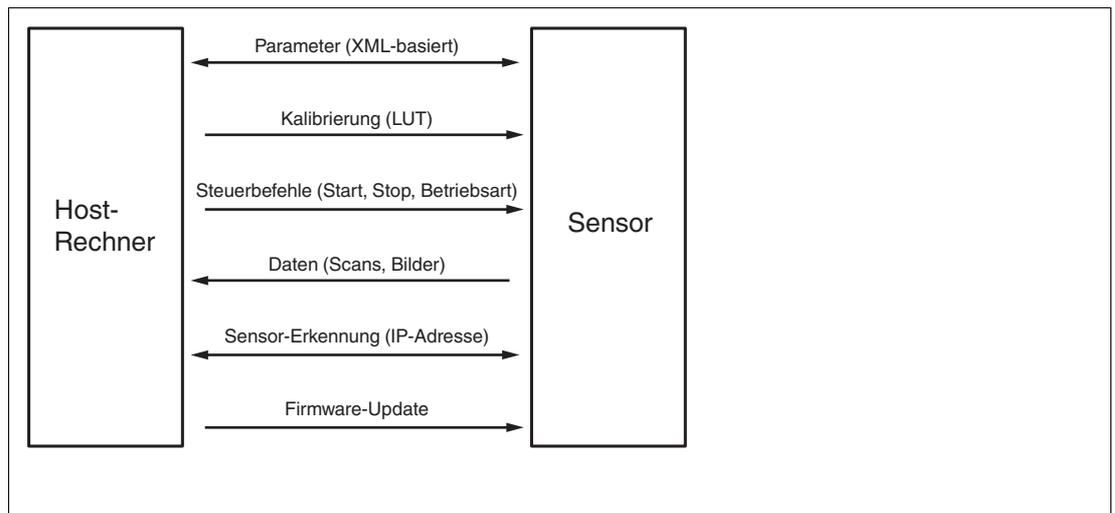


Abbildung 2.7 Netzwerkschnittstelle

2.7 Softwareschnittstelle

Der SmartRunner Explorer verfügt über eine VSX-Programmierschnittstelle. Der Treiber Vsx-ProtocolDriver bietet den vollen Zugriff auf die Ein- und Ausgangsdaten des Sensors und ermöglicht die Einbindung in eine C#-basierte Programmierumgebung. Hierzu stellt der Treiber eine Verbindung mit dem Sensor her und übernimmt die Kommunikation entsprechend dem Kommunikationsprotokoll. Dem Benutzer werden Funktionen zur Verfügung gestellt, mit denen Parameter auf dem Sensor eingestellt, Parameterwerte vom Sensor abgefragt und ganze Parametersätze sowohl lokal als auch auf dem Sensor gespeichert und geladen werden können. Weiterhin können Sensorbilder empfangen werden. Jede Funktion enthält darüber hinaus ein Error-Objekt, welchem im Falle eines Fehlers der Funktion Informationen entnommen werden können.

Zur einfachen Einbindung in eine PC-Software gibt es eine **.NET 5.0 basierende** Softwareschnittstelle. Diese wird in Form von DLLs zur Verfügung gestellt und übernimmt die Kommunikation zum Sensor. Binden Sie die DLLs in die Programmierumgebung ein und führen die beschriebenen Programmierzeilen aus.

Die Beispiele beziehen sich auf die Programmierumgebung Visual Studio 2019 und auf die Programmiersprache C#.



Hinweis!

Einbinden des NuGet

Um die DLL verwenden zu können, muss das NuGet eingebunden werden. In Visual Studio kann dies z.B. durch den NuGet-Pakete-Manager durchgeführt werden. Die DLL findet Sie auf der Produktseite des SmartRunner Explorer von Pepperl und Fuchs im Softwareordner. In der dort abgelegten ZIP-Datei befindet sich das NuGet unter dem Projektordner ext.



Hinweis!

Weitere Informationen zur Konfiguration siehe Kapitel 5.1.

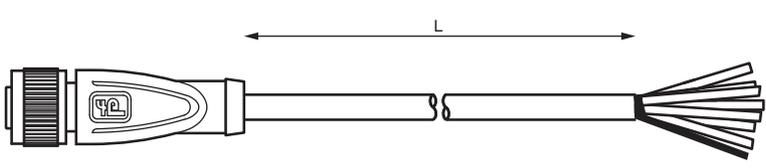
2.8 Zubehör

Es steht Ihnen verschiedenes Zubehör zur Verfügung.

2.8.1 Spannungsversorgung

Zum Anschluss der Spannungsversorgung, Eingänge und Ausgänge an den Sensor verwenden Sie folgende Verbindungskabel.

Konfektionierbare Kabeldose

Bestellbezeichnung			
V19-G-2M-PUR-ABG	M12-Buchse, gerade, 8-polig	L = 2 m	offenes Kabelende mit mehrdrähtigen Leitern
V19-G-5M-PUR-ABG	M12-Buchse, gerade, 8-polig	L = 5 m	offenes Kabelende mit mehrdrähtigen Leitern
V19-G-10M-PUR-ABG	M12-Buchse, gerade, 8-polig	L = 10 m	offenes Kabelende mit mehrdrähtigen Leitern

Weitere und abweichende Längen auf Anfrage.

2.8.2 Netzwerkkabel

Der Sensor wird über einen M12-Stecker mit dem Netzwerk verbunden.

Bezeichnung	Beschreibung
V45-G	RJ45-Netzwerkstecker, konfektionierbar
V1S-G	M12-Stecker, 4-polig, konfektionierbar
V1SD-G-2M-PUR-ABG-V45X-G	Verbindungskabel, RJ45-Netzwerkstecker mit M12-Stecker, gekreuzt, 4-polig
V1SD-G-2M-PUR-ABG-V45-G	Verbindungskabel, RJ45-Netzwerkstecker mit M12-Stecker, 4-polig

3 Installation

3.1 Lagerung und Entsorgung

Bewahren Sie die Originalverpackung auf. Lagern oder transportieren Sie das Gerät immer in der Originalverpackung.

Lagern Sie das Gerät immer in trockener und sauberer Umgebung. Beachten Sie die zulässigen Umgebungsbedingungen, siehe Datenblatt.

Das Gerät, die eingebauten Komponenten, die Verpackung sowie eventuell enthaltene Batterien müssen entsprechend den einschlägigen Gesetzen und Vorschriften im jeweiligen Land entsorgt werden.

3.2 Vorbereitung



Gerät auspacken

1. Prüfen Sie Verpackung und Inhalt auf Beschädigung.
↳ Benachrichtigen Sie bei Beschädigung den Spediteur und verständigen Sie den Lieferanten.
2. Prüfen Sie den Lieferumfang anhand Ihrer Bestellung und der Lieferpapiere auf Vollständigkeit und Richtigkeit.
↳ Bei auftretenden Fragen wenden Sie sich an Pepperl+Fuchs.
3. Bewahren Sie die Originalverpackung für den Fall auf, dass das Gerät zu einem späteren Zeitpunkt eingelagert oder verschickt werden soll.

3.3 Erfassungsbereich

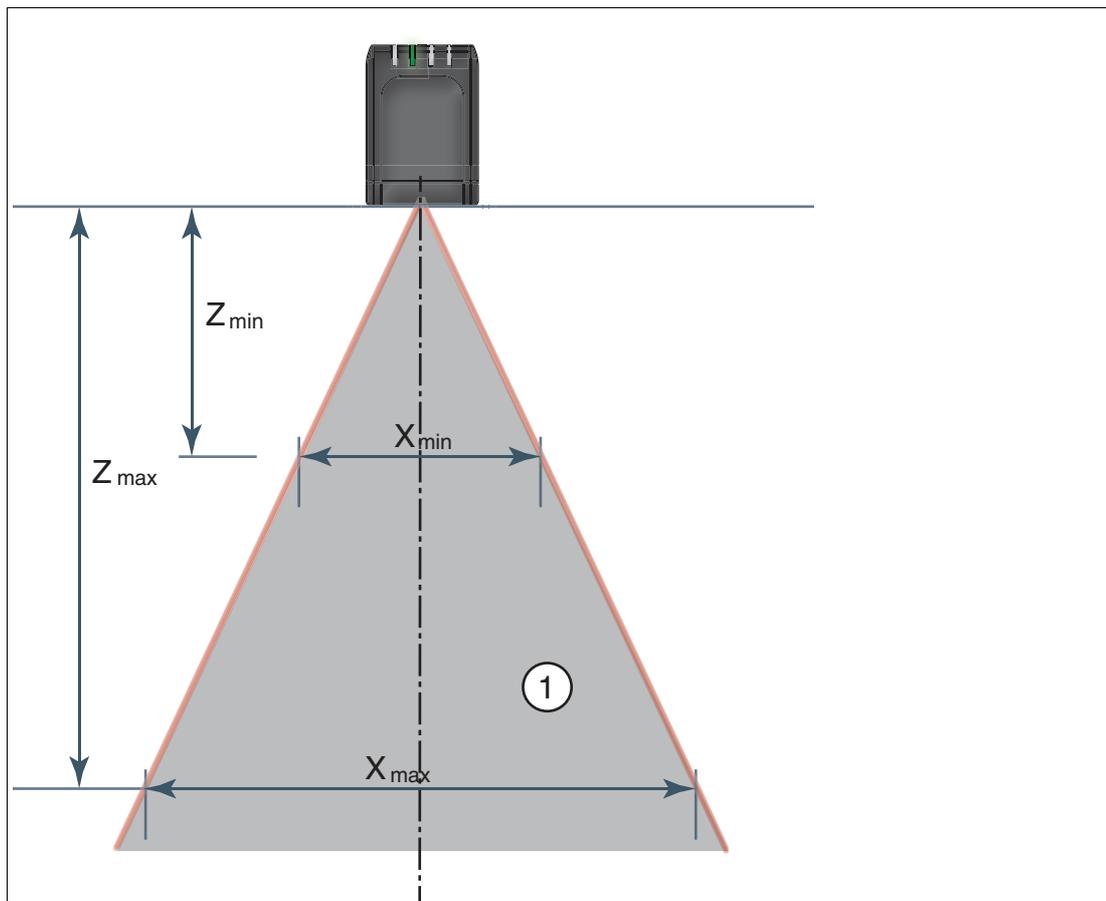


Abbildung 3.1 Erfassungsbereich

1 Sichtbereich

Beachten Sie bei der Planung Ihrer Anlage den Erfassungsbereich des SmartRunner Matchers. Detaillierte Informationen zum Erfassungsbereich finden Sie im jeweiligen Datenblatt des Sensors.



Hinweis!

Die kleinstmögliche Auflösung in X- und Z-Richtung steigt linear mit dem Abstand Z zum Sensor an.

3.4 Montage des Sensors



Hinweis!

Montage eines optischen Gerätes

- Zielen Sie mit dem Sensor nicht in die Sonne.
- Schützen Sie den Sensor vor direkter und dauerhafter Sonneneinwirkung.
- Beugen Sie die Bildung von Kondensation vor, indem Sie den Sensor keinen großen Temperaturschwankungen aussetzen.
- Setzen Sie den Sensor keinen Einflüssen von aggressiven Chemikalien aus.
- Halten Sie die Scheiben des Gerätes sauber. Verwenden Sie dazu weiche Tücher und gegebenenfalls handelsübliche Glasreiniger.

Wir empfehlen in regelmäßigen Abständen die Optikfläche zu reinigen und Verschraubungen, sowie die elektrischen Verbindungen zu überprüfen.

Der Arbeitsabstand ist je nach Sensor unterschiedlich. Entnehmen Sie den passenden Arbeitsabstand aus dem Datenblatt des zu montierenden Sensors.

Die folgenden beiden Abbildung zeigen die Orientierung des Sensors bei Fremdlicht:

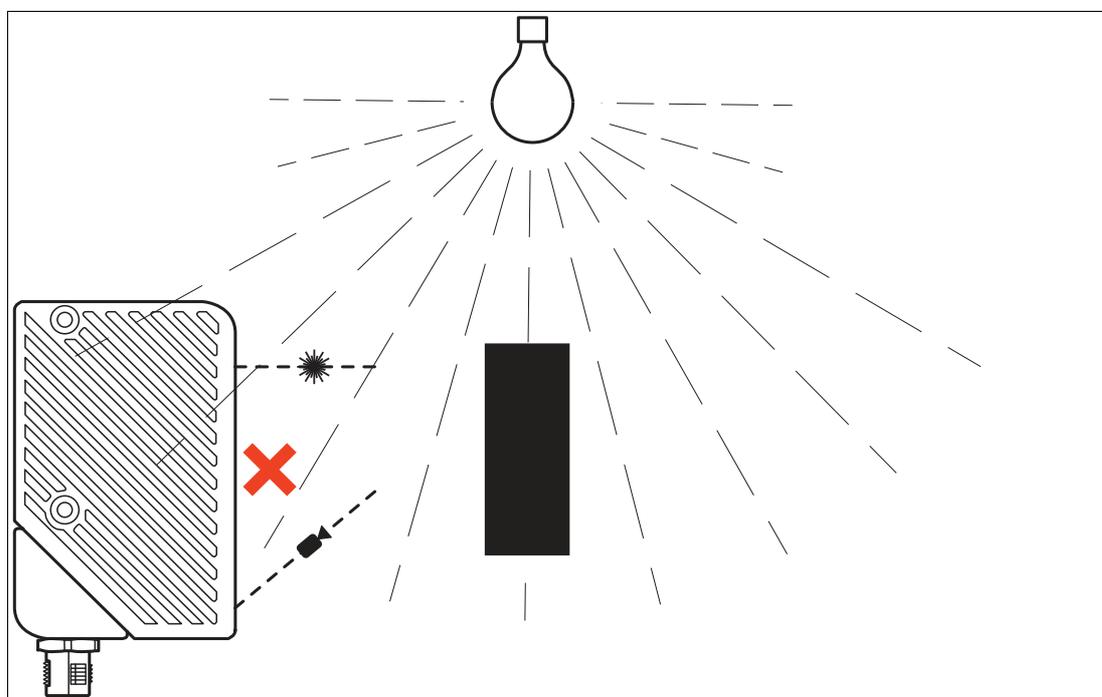


Abbildung 3.2 Ausrichtung falsch

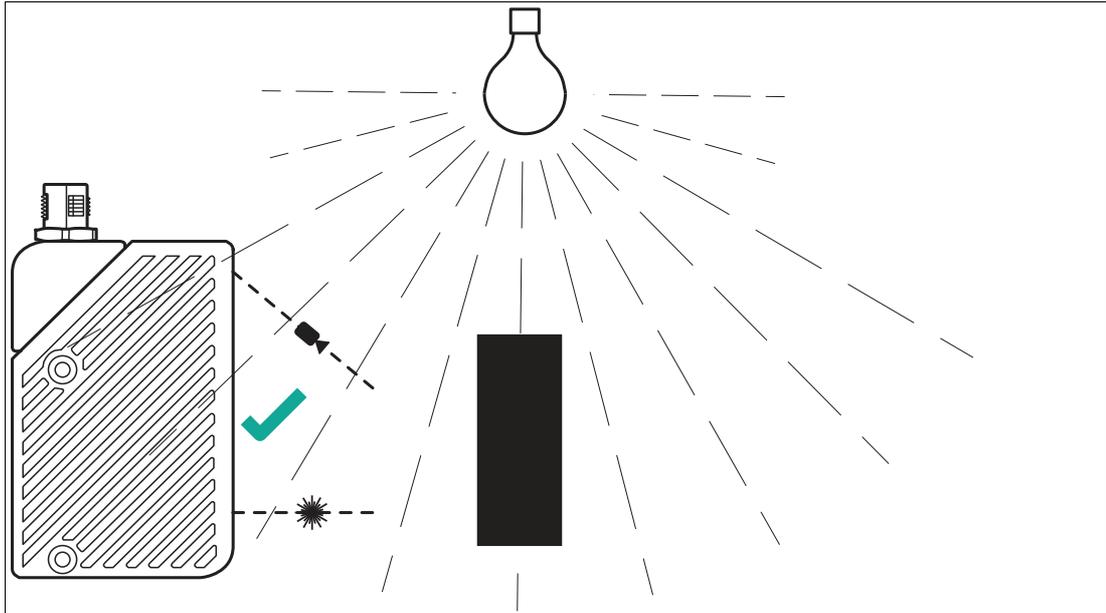


Abbildung 3.3 Ausrichtung richtig

Die Untergrundfläche muss plan sein, um Gehäuseverzug beim Festziehen zu vermeiden. Wir empfehlen, die Schrauben mit Federscheiben zu sichern, um einer Dejustierung des Sensors vorzubeugen. Achten Sie darauf, dass nach der Montage des Sensors noch ausreichend Platz vorhanden ist, um die Anschlusskabel an den Sensor anzuschliessen



Vorsicht!

Geräteschäden durch unsachgemäße Montage!

Wenn die zulässige Einschraubtiefe und der maximal zulässige Anziehdrehmoment überschritten werden, können Gerätekomponenten beschädigt werden.

Beachten Sie, dass die Gewinde am Gehäuseboden keine Durchgangsbohrungen sind.

Beachten Sie die zulässige maximale Einschraubtiefe, um Geräteschäden oder eine nicht korrekte Befestigung zu vermeiden.

Überschreiten Sie niemals die maximal zulässige Anziehdrehzahl der Befestigungsschrauben. Das maximale Anziehdrehmoment der Befestigungsschrauben darf 2 Nm nicht übersteigen.

Befestigung des Gehäuses

Zur einfachen Montage des Sensors in Ihrer Anlage verfügt das Gerät an den beiden Gehäuseseiten und am Gehäuseboden über jeweils 2 M4-Gewinde. Daraus ergeben sich 3 verschiedene Möglichkeiten den Sensor in Ihrer Anlage zu montieren.

- Einseitige seitliche Befestigung mit M4-Schrauben: Über die 2 M4-Gewindehülsen, können Sie das Gehäuse mit seiner rechten oder linken Seite montieren. Die maximale Einschraubtiefe der M4-Schrauben beträgt 8 mm.
- Durchgehende seitliche Befestigung mit M3-Schrauben: Die M4-Gewindehülsen sind so angebracht, dass M3-Schrauben ganz durch das Gehäuse durchgehen. Verwenden Sie 2 ausreichend lange M3-Schrauben mit 2 Kontermuttern, um das Gerät in der Anlage zu montieren
- Befestigung an Geräteunterseite mit M4-Schrauben: Über die 2 Gewindehülsen können Sie das Gehäuse an der Geräteunterseite montieren. Die maximale Einschraubtiefe der M4-Schrauben beträgt 5 mm.

Positionierung des Sensors

Achten Sie bei der Positionierung des Sensors, dass der Sichtbereich der Kamera nicht durch Objekte verdeckt wird, die gescannt werden sollen.

3.5 Anschluss des Sensors



Vorsicht!

Beschädigung des Geräts

Anschließen von Wechselspannung oder zu hoher Versorgungsspannung kann das Gerät beschädigen oder die Gerätefunktion stören.

Falscher elektrischer Anschluss durch Verpolung kann das Gerät beschädigen oder die Gerätefunktion stören.

Gerät an Gleichspannung (DC) anschließen. Stellen Sie sicher, dass die Höhe der Versorgungsspannung im spezifizierten Bereich des Geräts liegt. Stellen Sie sicher, dass die Anschlussdrähte der verwendeten Kabeldose richtig angeschlossen sind.



Versorgungsspannung anlegen

Um den Sensor mit Spannung zu versorgen, gehen Sie wie folgt vor:

1. Stecken Sie die Buchse M12, 8-polig am Versorgungskabel in den dafür vorgesehenen Stecker an der Gehäuseseite.
2. Drehen Sie die Überwurfmutter über den Steckverbinder, bis zum Endanschlag.

↳ Damit ist das Versorgungskabel gegen versehentliches Herausziehen gesichert.



Hinweis!

Abschirmung von Leitungen

Das Abschirmen ist eine Maßnahme zur Dämpfung elektromagnetischer Störungen. Damit diese Störströme nicht selbst zur Störquelle werden, ist eine niederohmige bzw. impedanzarme Verbindung zum Schutzleiter bzw. Potenzialausgleich besonders wichtig. Verwenden Sie nur Anschlussleitungen mit Schirmgeflecht. Vermeiden Sie Anschlussleitungen mit Folienschirm, weil dies die Leitungskapazitäten erhöhen würde. Die Abschirmung wird beidseitig aufgelegt, d. h. im Schaltschrank **und** am Sensor. Die als Zubehör erhältliche Erdungsklemme ermöglicht das einfache Einbeziehen in den Potenzialausgleich.

In Ausnahmefällen kann eine einseitige Anbindung günstiger sein, wenn

- keine Potenzialausgleichsleitung verlegt ist bzw. keine Potenzialausgleichsleitung verlegt werden kann.
- ein Folienschirm verwendet wird.

Bei der Abschirmung müssen ferner folgende Punkte beachtet werden:

- Verwenden Sie Kabelschellen aus Metall, die die Abschirmung großflächig umschließen.
- Legen Sie den Kabelschirm direkt nach Eintritt in den Schaltschrank auf die Potenzialausgleichsschiene.
- Führen Sie Schutzerdungsanschlüsse sternförmig zu einem gemeinsamen Punkt.
- Verwenden Sie für die Erdung möglichst große Leitungsquerschnitte.



Netzwerkverbindung herstellen

Um eine Netzwerkverbindung herzustellen, gehen Sie wie folgt vor:

1. Verwenden Sie ein Netzkabel, das auf einer Seite einen RJ45 Netzwerkstecker und auf der anderen Seite einen M12 Stecker 4-polig hat. Stecken Sie den M12 Stecker, 4-polig in die Buchse auf der Seite des Sensors.
2. Im Auslieferungszustand besitzt der Sensor eine feste IP-Adresse. Um eine Kommunikation im Netzwerk zu ermöglichen, konfigurieren Sie Ihr Netzwerk. Entnehmen Sie die Konfigurationsdaten der Netzwerk-Konfigurationsübersicht.



Hinweis!

Netzwerkconfiguration dokumentieren

Der Sensor kommuniziert mit der angeschlossenen Maschinensteuerung über das TCP/IP-Protokoll. Um eine korrekte Kommunikation zu gewährleisten, notieren Sie sich unbedingt alle Änderungen, die Sie an der Netzwerkconfiguration vornehmen.

3.6 Windows Netzwerkkommunikation Sensor-PC/Laptop einrichten

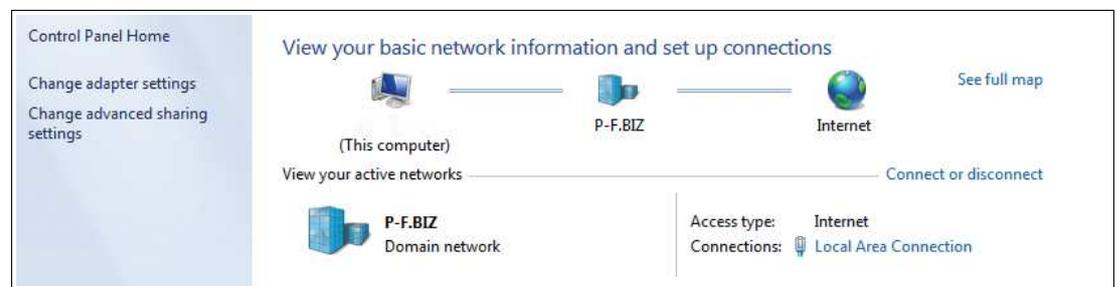
Im Auslieferungszustand besitzt der Sensor eine feste IP-Adresse **192.168.2.3**. Um eine Kommunikation im Netzwerk zu ermöglichen, ist es notwendig die Netzwerkeinstellungen Ihres PCs/Laptops mit dem Sensor abzugleichen und ggf. einzustellen. Gehen Sie dazu wie folgt vor:



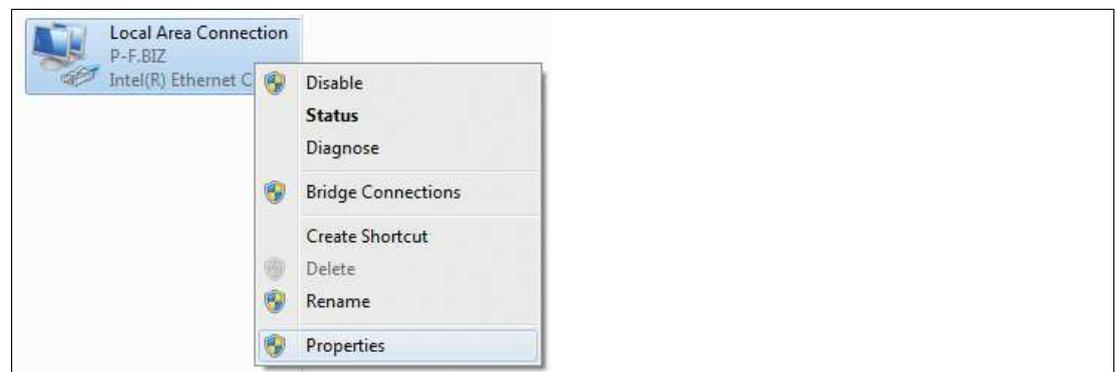
IP-Adresse einrichten

Nachfolgenden wird beschrieben, wie Sie die Netzwerk-Verbindungseinstellungen Ihres Windows PCs prüfen und entsprechend anpassen. Die Abbildungen in dieser Beschreibung wurden mit Windows 7 erstellt. Die nachfolgende Beschreibung gilt auch für höhere Windows-Versionen.

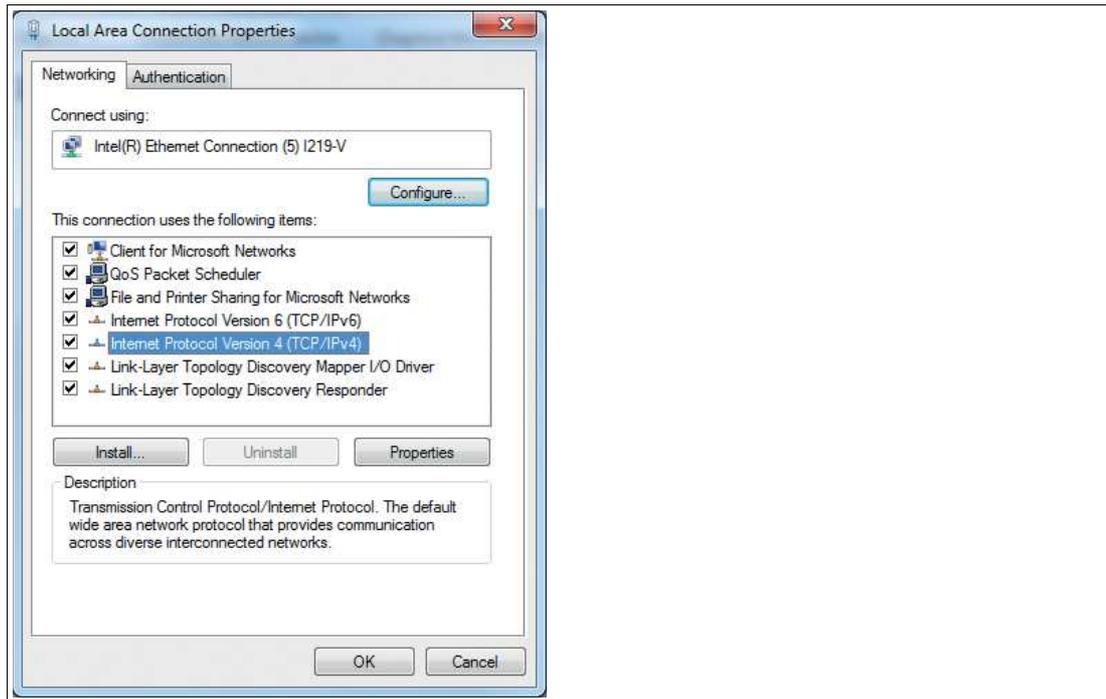
1. Klicken Sie auf die Windows-Schalfläche "**Start**".
2. Wählen Sie "**Control Panal > Network and Sharing Center**".
3. Jetzt klicken Sie auf "**Change adapter settings**".



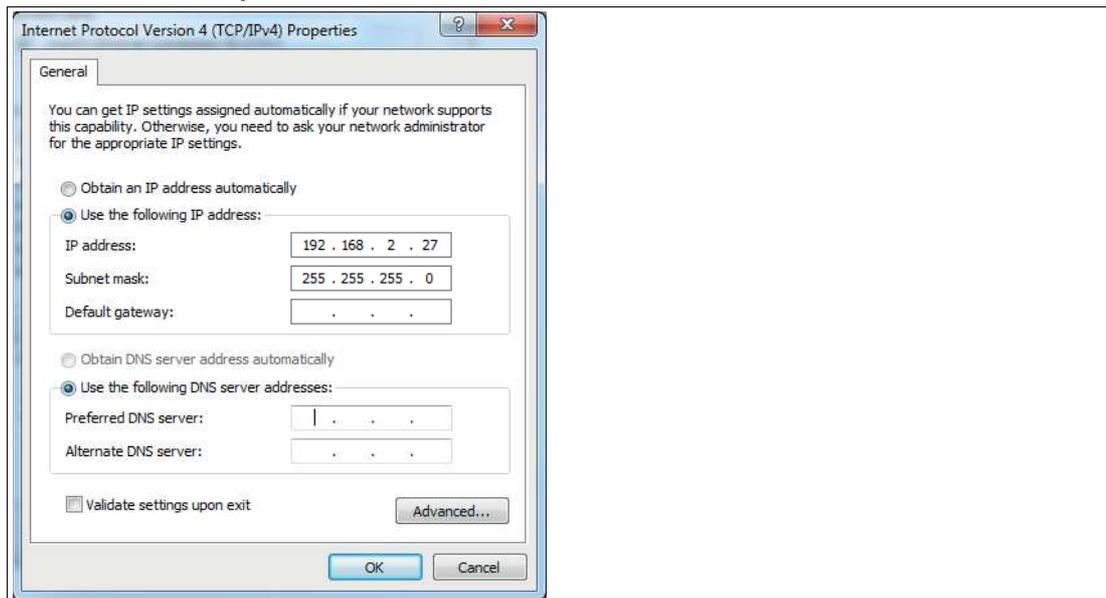
4. Wählen Sie die gewünschte Verbindung und führen Sie einen Rechtsklick auf Ihre Auswahl. Wählen Sie im Auswahlfenster die Funktion **Properties**.



5. Doppelklicken Sie auf "**Internet Protocol Version 4 (TCP/IPv4)**".



↳ Das Fenster **Properties** des TCP/IP-Protokolls erscheint.



6. Wählen Sie den Reiter "**General**".
7. Aktivieren Sie die Eingabefunktion "**Use the following IP address**".
8. Tragen Sie die IP-Adresse des Sensors ein, jedoch nur die erste drei Segmente der IP-Adresse. Das letzte Segment muss sich von der IP-Adresse des Sensors unterscheiden.
9. Im vorliegenden Beispiel geben Sie die folgende IP-Adresse und Subnetzmaske ein:
 - **IP-address: 192.168.2.27**
 - **Subnet mask: 255.255.255.0**
10. Klicken Sie auf **OK** und im nächsten Dialog auf **Cancel**.

↳ Die Netzwerkkonfiguration ist damit abgeschlossen und das Gerät kann verwendet werden.



Hinweis!

Änderungen in den Netzwerkeinstellungen des PCs/Laptops erfordern erweiterte Benutzerrechte. Wenden Sie sich gegebenenfalls bei Ihren Administrator.

4 Inbetriebnahme

Übersicht über die Schritte für die Inbetriebnahme:

1. Arbeitsabstand einstellen, siehe Kapitel 3.3
2. Sensor montieren, siehe Kapitel 3.4
3. Sensor anschließen, siehe Kapitel 3.5
4. Windows Netzwerkkommunikation Sensor-PC/Laptop einrichten, siehe Kapitel 3.6
5. Sensor mit der Konfigurationssoftware "Vision Configurator" konfigurieren, siehe Kapitel 6

5 Konfiguration

5.1 VsxProtocolDriver

Allgemein

Der SmartRunner Explorer verfügt über eine VSX-Programmierschnittstelle. Der Treiber Vsx-ProtocolDriver bietet den vollen Zugriff auf die Ein- und Ausgangsdaten des Sensors und ermöglicht die Einbindung in eine C#-basierte Programmierumgebung. Hierzu stellt der Treiber eine Verbindung mit dem Sensor her und übernimmt die Kommunikation entsprechend dem Kommunikationsprotokoll. Dem Benutzer werden Funktionen zur Verfügung gestellt, mit denen Parameter auf dem Sensor eingestellt, Parameterwerte vom Sensor abgefragt und ganze Parametersätze sowohl lokal als auch auf dem Sensor gespeichert und geladen werden können. Weiterhin können Sensorbilder empfangen werden. Jede Funktion enthält darüber hinaus ein Error-Objekt, welchem im Falle eines Fehlers der Funktion Informationen entnommen werden können.

Der Treiber ist in C# implementiert und benötigt als Voraussetzung .NET Standard 5.0 oder höher.

Die Funktionen des Treibers können **synchron** oder **asynchron** verwendet werden. Hierfür muss jeweils die gewünschte Instanz unter Verwendung der Init-Funktion erstellt werden.

In diesem Handbuch werden aus Gründen der Übersichtlichkeit nur die wichtigsten Funktionen und Variablen beschrieben. Die DLL enthält zudem weitere Funktionen, die für andere Pepperl+Fuchs-Vision-Sensoren verwendet werden. Diese Funktionen werden im zugehörigen Produkthandbuch beschrieben. Bei einigen Funktionen gibt es mehrere Deklarationsmöglichkeiten. Im Nachfolgenden werden die präferierten Funktionen fett markiert.



Hinweis!

Einbinden des NuGet

Um die DLL verwenden zu können, muss das NuGet eingebunden werden. In Visual Studio kann dies z.B. durch den NuGet-Pakete-Managers durchgeführt werden. Die DLL findet Sie auf der Produktseite des SmartRunner Explorer von Pepperl und Fuchs im Softwareordner. In der dort abgelegten ZIP-Datei befindet sich das NuGet unter dem Projektordner ext.

Synchrone und asynchrone Funktionen

Die in den Parametern verwendeten Hilfsklassen werden im Anschluss beschrieben.

Initialisierung

UDP-Broadcast

Die statische Funktion gibt eine Liste der Vsx-Geräte zurück, welche über einen UDP-Broadcast im Netzwerk gefunden werden.

Asynchrone Funktion

```
public static async Task<(bool Succ, List<Device> DeviceList, Error ErrorDesc)> UdpDeviceList()
```

Synchrone Funktion

```
public static (bool Succ, List<Device> DeviceList, Error ErrorDesc) UdpDeviceList()
```

Mögliche Error-IDs: VSX_DRIVER_CONNECTION_ERROR

Neue Treiberinstanz TCP/IP

Initialisiert eine neue Treiberinstanz, mittels derer über TCP/IP mit dem Gerät kommuniziert werden kann. Während die IP-Adresse angegeben werden muss, kann der Standard VSX-PORT = 50005 verwendet werden.

Bei dieser Sensorversion gibt es keinen Plugin-Namen. Der Port und der Plugin-Name sind nur optional verwendbar.

Asynchrone Funktion

```
public static VsxProtocolDriver Init(string ipAddress, int port = VSXPORT, string pluginName = "")
```

Synchrone Funktion

```
public static VsxProtocolDriverSync Init(string ipAddress, int port = VSXPORT, string pluginName = "")
```

Mögliche Error-IDs: Keine

Verbindung herstellen

Stellt eine Verbindung mit dem Gerät unter Verwendung der mittels Init gesetzten Parameter her. Als Timeout für das Öffnen der Verbindung kann der CONNECTION_TIMEOUT_MS = 1000 verwendet werden. Für die Verwendung aller nicht statischer Funktionen muss eine Verbindung zum Gerät hergestellt sein.

Asynchrone Funktion

```
public async Task<(bool Succ, Error ErrorDesc)> Connect(int timeout = CONNECTION_TIMEOUT_MS)
```

Synchrone Funktion

```
public (bool Succ, Error ErrorDesc) Connect(int timeout = VsxProtocolDriver.CONNECTION_TIMEOUT_MS)
```

Mögliche Error-IDs: VSX_DRIVER_CONNECTION_ERROR

Verbindung trennen

Asynchrone Funktion

```
public async Task<(bool Succ, Error ErrorDesc)> Disconnect()
```

Synchrone Funktion

```
public (bool Succ, Error ErrorDesc) Disconnect()
```

Mögliche Error-IDs: Keine

IVsxMessage

Eine IVsxMessage ist ein allgemeines Interface für Daten, die zwischen PC und Sensor ausgetauscht werden.

Asynchrone Funktion

```
public static (bool Succ, Error ErrorDesc) SaveData(string filename, IVsxMessage message)
```

Mögliche Error-IDs: VSX_DRIVER_DATA_ERROR, VSX_DRIVER_INVALID_DATA_ERROR, VSX_DRIVER_SAVE_FILE_ERROR

Geräteinformationen

Liefert Informationen über das Gerät zurück (wie u.a. MAC-Adresse usw.).

Asynchrone Funktion

```
public async Task<(bool Succ, Device CurrentDevice, Error ErrorDesc)> GetCurrentDeviceInformation()
```

Synchrone Funktion

```
public (bool Succ, Device CurrentDevice, Error ErrorDesc) GetCurrentDeviceInformation()
```

Mögliche Error-IDs: VSX_DRIVER_CONNECTION_ERROR

Wert eines einzelnen Geräteparameters liefern

Liefert den Wert eines einzelnen Parameters des Geräts zurück.

Asynchrone Funktion

```
public async Task<(bool Succ, object parameterValue, Error ErrorDesc)> GetSingleParameterValue(Parameter parameter)
```

```
public async Task<(bool Succ, object parameterValue, Error ErrorDesc)> GetSingleParameterValue(string parameterId)
```

```
public async Task<(bool Succ, object parameterValue, Error ErrorDesc)> GetSingleParameterValue(ushort settingsVersion, ushort configVersion, string configId, string parameterId)
```

Synchrone Funktion

```
public (bool Succ, object parameterValue, Error ErrorDesc) GetSingleParameterValue(Parameter parameter)
```

```
public (bool Succ, object parameterValue, Error ErrorDesc) GetSingleParameterValue(string parameterId)
```

```
public (bool Succ, object parameterValue, Error ErrorDesc)
GetSingleParameterValue(ushort settingsVersion, ushort
configVersion, string configId, string parameterId)
```

Mögliche Error-Ids: VSX_DRIVER_CONNECTION_ERROR, VSX_DRIVER_DATA_ERROR

Wert eines einzelnen Geräteparameters setzen

Setzt den Wert eines einzelnen Parameters auf dem Gerät. Der Parameter wird durch die Übergabe der Funktionsparameter analog zur Funktion `GetSingleParameterValue` definiert (siehe Beschreibung hierzu). Zusätzlich wird der gewünschte Wert übergeben.

Asynchrone Funktion

```
public async Task<(bool Succ, Error ErrorDesc)>
SetSingleParameterValue(Parameter parameter, object value)
```

```
public async Task<(bool Succ, Error ErrorDesc)>
SetSingleParameterValue(string parameterId, object value)
```

```
public async Task<(bool Succ, Error ErrorDesc)>
SetSingleParameterValue(ushort settingsVersion, ushort
configVersion, string configId, string parameterId, object value)
```

Synchrone Funktion

```
public (bool Succ, Error ErrorDesc) SetSingleParameterValue(Parameter
parameter, object value)
```

```
public (bool Succ, Error ErrorDesc) SetSingleParameterValue(string
parameterId, object value)
```

```
public (bool Succ, Error ErrorDesc) SetSingleParameterValue(ushort
settingsVersion, ushort configVersion, string configId, string
parameterId, object value)
```

Mögliche Error-IDs: VSX_DRIVER_CONNECTION_ERROR, VSX_DRIVER_DATA_ERROR

Netzwerkeinstellungen ändern

Ändert die Netzwerkeinstellungen auf dem Gerät. Anschließend ist die Verbindung zum Gerät getrennt und muss mittels der "Connect"-Funktion neu aufgebaut werden.

Asynchrone Funktion

```
public async Task<(bool Succ, Error ErrorDesc)>  
SetNetworkSettings(string ipAddress, string networkMask, string  
gateway)
```

Synchrone Funktion

```
public (bool Succ, Error ErrorDesc) SetNetworkSettings(string  
ipAddress, string networkMask, string gateway)
```

Mögliche Error-IDs: VSX_DRIVER_CONNECTION_ERROR

Firmwaredatei senden

Sendet die unter dem angegebenen Pfad und Dateinamen liegende Firmwaredatei zum Gerät. Während des laufenden Updates kann über den FirmwareStateChannelReader der aktuelle Status ausgelesen werden.

Asynchrone Funktion

```
public async Task<(bool Succ, Error ErrorDesc)> SendFirmware(string  
fileName)
```

Synchrone Funktion

```
public (bool Succ, Error ErrorDesc) SendFirmware(string fileName)
```

Mögliche Error-IDs: VSX_DRIVER_CONNECTION_ERROR,
VSX_DRIVER_DEVICE_ERROR

Informationen zu Firmwareupdates

Kann asynchron ausgelesen werden und beinhaltet während eines Firmwareupdates Informationen über den aktuellen Status.

```
public ChannelReader<FirmwareState> FirmwareStateChannelReader
```

Parametersatz auslesen

Liest den aktuellen Parametersatz vom Gerät und speichert ihn unter dem angegebenen Pfad und Dateinamen ab.

Asynchrone Funktion

```
public async Task<(bool Succ, Error ErrorDesc)>  
DownloadParameterSet(string destinationFileName)
```

Synchrone Funktion

```
public (bool Succ, Error ErrorDesc) DownloadParameterSet(string  
destinationFileName)
```

Mögliche Error-IDs: VSX_DRIVER_SAVE_FILE_ERROR

Parametersatz speichern

Speichert die aktuellen Parametereinstellungen auf dem Gerät ab. Die eingestellten Werte werden dann bei jedem Gerätestart gesetzt.

Alle Parameterdaten werden zunächst flüchtig auf dem Gerät gespeichert und können durch einen Neustart oder durch LoadParameterSetOnDevice zurückgesetzt werden. Erst durch SaveParameterSetOnDevice werden die Parameter dauerhaft gespeichert.

Asynchrone Funktion

```
public async Task<(bool Succ, Error ErrorDesc)>  
SaveParameterSetOnDevice()
```

Synchrone Funktion

```
public (bool Succ, Error ErrorDesc) SaveParameterSetOnDevice()
```

Mögliche Error-IDs: VSX_DRIVER_CONNECTION_ERROR

Parametereinstellungen laden

Lädt die mit SaveParameterSetOnDevice() gespeicherten Parametereinstellungen auf das Gerät. Die Parameter haben dann die zuvor gespeicherten Werte. Eine aktuelle Parameterliste wird zurückgegeben.

Asynchrone Funktion

```
public async Task<(bool Succ, List<Parameter> ParameterList, Error  
ErrorDesc)> LoadParameterSetOnDevice()
```

Synchrone Funktion

```
public (bool Succ, List<Parameter> ParameterList, Error ErrorDesc)  
LoadParameterSetOnDevice()
```

Mögliche Error-IDs: VSX_DRIVER_GENERAL_ERROR, VSX_DRIVER_DATA_ERROR,
VSX_DRIVER_CONNECTION_ERROR

Werkseinstellungen laden

Lädt die Werkseinstellungen aller Parameter auf dem Gerät. Eine aktuelle Parameterliste wird zurückgegeben.

Asynchrone Funktion

```
public async Task<(bool Succ, List<Parameter> ParameterList, Error ErrorDesc)> LoadDefaultParameterSetFromDevice()
```

Synchrone Funktion

```
public (bool Succ, List<Parameter> ParameterList, Error ErrorDesc) LoadDefaultParameterSetFromDevice()
```

Mögliche Error-IDs: VSX_DRIVER_GENERAL_ERROR, VSX_DRIVER_DATA_ERROR, VSX_DRIVER_CONNECTION_ERROR

Auslesen von Sensordaten

Startet das Auslesen von Sensordaten. Die Sensordaten werden pro Trigger in VsxDynamic-Container gepackt. Diese können mit der Funktion GetDynamicContainer abgerufen werden. Für mehr Details siehe Containerbeschreibung. Die bufferSize gibt an, wie viele Container vom Treiber zwischengespeichert werden können, die startCondition ab welchem Container zwischengespeichert werden soll und die strategy was passieren soll, wenn der Zwischenspeicher voll ist. Bei DROP_OLDEST wird dann der älteste gespeicherte Container verworfen, bei DROP_WRITE der aktuell angekommene Container.

Asynchrone Funktion

```
public void ResetDynamicContainerGrabber(int bufferSize, int startCondition = -1, DataFunction.Strategy = DataFunction.Strategy.DROP_OLDEST)
```

Synchrone Funktion

```
public void ResetDynamicContainerGrabber(int numberOfItems, int startCondition = -1, Strategy strategy = Strategy.DROP_OLDEST)
```

Mögliche Error-IDs: Keine

Ausgabe des ältesten zwischengespeicherten DynamicContainer

Gibt den ältesten zwischengespeicherten DynamicContainer (siehe "DynamicContainer" auf Seite 38) zurück (siehe ResetDynamicContainerGrabber). timeoutMs gibt an, wie lange in ms versucht wird, einen Container aus dem Zwischenspeicher zu lesen. Ist in dieser Zeit kein Container vorhanden, kommt die Funktion mit einem Fehler zurück.

Asynchrone Funktion

```
public async Task<(bool Succ, IVsxDynamicContainer Container, int  
NumberOfDiscardedItems, Error ErrorDesc)> GetDynamicContainer(int  
timeoutMs = System.Threading.Timeout.Infinite)
```

Synchrone Funktion

```
public (bool Succ, IVsxDynamicContainer Container, int  
NumberOfDiscardedItems, Error ErrorDesc) GetDynamicContainer(int  
timeoutMs = Timeout.Infinite)
```

Mögliche Error-IDs: VSX_DRIVER_INIT_ERROR, VSX_DRIVER_TIMEOUT_ERROR

Auslesen von Logdaten

Startet das Auslesen von Logdaten neu. Die Logdaten können mit der Funktion GetLogMessage abgerufen werden. Die bufferSize gibt an, wie viele Logdatenmessages vom Treiber zwischengespeichert werden können. Die typeMask gibt an, welche Logdatentypen vom Gerät übertragen werden sollen und die "strategy" gibt an was passieren soll, wenn der Zwischenspeicher voll ist. Bei _DROP_OLDEST wird die älteste gespeicherte Logmessage verworfen und bei DROP_WRITE die aktuell angekommene Logmessage.

Asynchrone Funktion

```
public void ResetLogMessageGrabber(int bufferSize, int typeMask,  
DataFunction.Strategy = DataFunction.Strategy.DROP_OLDEST)
```

Mögliche Error-IDs: Keine

Ausgabe des ältesten zwischengespeicherten LogMessage

Gibt die älteste zwischengespeicherten LogMessage zurück (siehe ResetLogMessageGrabber). timeoutMs gibt an, wie lange in ms versucht wird, eine Logmessage aus dem Zwischenspeicher zu lesen. Ist in dieser Zeit keine Logmessage vorhanden, kommt die Funktion mit einem Fehler zurück.

Asynchrone Funktion

```
public async Task<(bool Succ, VsxLogMessage LogMessage, int  
NumberOfDiscardedItems, Error ErrorDesc)> GetLogMessage(int  
timeoutMs = System.Threading.Timeout.Infinite)
```

Synchrone Funktion

```
public (bool Succ, VsxLogMessage LogMessage, int  
NumberOfDiscardedItems, Error ErrorDesc) GetLogMessage(int timeoutMs  
= Timeout.Infinite)
```

Mögliche Error-IDs: VSX_DRIVER_INIT_ERROR, VSX_DRIVER_TIMEOUT_ERROR

Properties

Verbindungsstatus

Gibt den Verbindungsstatus an.

```
public bool Connected { get; }
```

Timeout

Timeout in ms der angibt, wie lange bei einer Anfrage an das Gerät auf dessen Antwort gewartet wird. Der Standardwert ist "DEFAULT_ETHERNET_TIMEOUT_MS".

```
public int WaitTimeout { get; set; }
```

Anzahl wie viele DynamicContainer bzw. LogMessages verworfen wurden

Gibt an, wie viele DynamicContainer bzw. LogMessages seit dem letzten Reset Grabber verworfen wurden, nachdem kein Platz mehr im Zwischenspeicher war.

```
public int MissingContainerFramesCounter { get; }
```

```
public int MissingLogMessagesCounter { get; }
```

Anzahl wie viele DynamicContainer bzw. LogMessages zwischengespeichert wurden

Gibt an, wie viele Dynamic Container bzw. Log Messages vom Treiber zwischengespeichert werden können.

```
public int DynamicContainerQueueSize { get; }
```

```
public int LogMessageQueueSize { get; }
```

Gibt an, welche Container bzw. Log Messages verworfen werden sollen

Gibt an, welche Container bzw. Log Messages verworfen werden sollen, sofern der Zwischenspeicher voll ist. DROP_OLDEST verwirft die älteste gespeicherte, DROP_WRITE verwirft die aktuell angekommene.

```
public Strategy DynamicContainerGrabberStrategy { get; }
```

```
public Strategy LogMessageGrabberStrategy { get; }
```

Events

Verbindungsverlust

Wird ausgelöst, sobald der Treiber einen Verbindungsverlust zum Gerät feststellt. Parameter sind die IP des zuvor verbundenen Geräts und eine Meldung, warum die Verbindung verloren gegangen ist. Dieses Event wird nur ausgelöst, wenn eine TCP/IP-Verbindung verwendet wird.

```
public event Action<string, string> OnDisconnect
```

Hilfsklassen

Device

Beinhaltet Informationen über das aktuell verbundene Gerät.

```
public string PhysicalAddress;
```

```
public int PhysicalPort;
```

```
public string IpAddress;
```

```
public string NetworkMask;
```

```
public string Gateway;
```

```
public string MacAddress;
```

```
public string Identifier;
```

```
public string FirmwareVersion;
```

```
public string SensorType;
```

Error

Beinhaltet Informationen über einen aufgetretenen Fehler.

```
public ErrorId Id;
```

```
public string Tag;
```

```
public string Message;
```

Folgende Error-IDs sind möglich

VSX_DRIVER_NO_ERROR

VSX_DRIVER_INIT_ERROR

VSX_DRIVER_TIMEOUT_ERROR

VSX_DRIVER_SAVE_FILE_ERRO

VSX_DRIVER_DATA_ERROR

VSX_DRIVER_CONNECTION_ERROR

VSX_DRIVER_INVALID_DATA_ERROR

VSX_DRIVER_DEVICE_ERROR

VSX_DRIVER_GENERAL_ERROR

FirmwareState

Beinhaltet Informationen über den aktuellen Status eines laufenden Firmwareupdates.

```
public int Id;
```

```
public string Tag;
```

```
public string Message;
```

DynamicContainer

Ein vom Sensor empfangener DynamicContainer kann die folgenden Messages enthalten:

TagName	Typ	Status	Beschreibung
"Image"	ImageData[Mono8]	optional	Bilddaten
"Line"	LineMessage	optional	Liniendaten

Aufbau des Typs LineMessage

Haupteigenschaft

Line

Gibt eine Punktliste zurück. Jeder Punkt besteht aus Bild-, Weltkoordinaten und Qualitätsinformationen. Im Detail sieht die Punktinformation wie folgt aus:

Die LineMessage hat folgendes Interface:

- ContentValue: Reserviert für zukünftige Verwendung
- ImageCoordinate: Bildkoordinaten in Subpixeln
- Intensity: Wert der mit dem Punktgrauwert korreliert
- LineId: Immer 0 in Einzelliniensystemen
- Quality: Punktqualität in Prozent
- SegmentId: Wenn unterstützt Liniensegmentnummer ansonsten 0
- Valid: Wahr wenn Punkt gültig ist
- WorldCoordinate: Weltkoordinaten in mm

Nebeneigenschaften (Metainformationen)

Metainformationen geben zusätzliche Informationen zu den Daten.

MinX

Minimal möglicher X-Wert der Gerätefamilie, üblicherweise negativ in μm .

MaxX

Maximaler möglicher X-Wert der Gerätefamilie, üblicherweise positiv in μm .

MinZ

Minimal möglicher Z-Wert der Gerätefamilie, positiv in μm .

MaxZ

Maximaler möglicher Z-Wert der Gerätefamilie, positiv in μm .

FrameCounter

Aktueller Bildzähler, zusammengehörige Daten haben den gleichen Zähler.

ScaleXYZ, ScaleC

Skalierungsfaktoren für die interne Berechnung der Liniendaten.

Width

Maximale Anzahl der möglichen Linienpunkte.

Format

Reserviert für zukünftige Verwendung.

Status

Reserviert für zukünftige Verwendung.

Lines

Gibt eine Liste von Linien zurück. In der Regel enthält die Liste nur eine Linie. Jede Linie besteht aus einer Liste von Punkten, die Punkte der ersten Linie können direkt über Line abgerufen werden.

VsxDynamicContainerMessage : IVsxMessage

Beinhaltet eine Liste von IVsxMessages, welche wiederum gesendete Daten vom Gerät beinhalten. Die enthaltenen Messages werden in der Liste mittels eines String identifiziert. Die möglichen Messages sind gerätespezifisch. Beispiel zum VsxDynamicContainerMessage, siehe "2. Ausgabe von Liniendaten" auf Seite 45.

```
public bool ContainsMessage(string tag)
```

```
public IVsxMessage GetMessage(string tag)
```

VsxImageData2Message : IVsxMessage

Beinhaltet Bilddaten eines speziellen Bildes. Die einzelnen Bildwerte werden als Bytes gespeichert.

```
public ImageData2Format;  
  
public int Width;  
  
public int Height;  
  
public int LinePitch;  
  
public long FrameCounter;  
  
public double CoordinateScale;  
  
public double CoordinateOffset;  
  
public double AxisMin;  
  
public double AxisMax;  
  
public double InvalidDataValue;  
  
public byte[] ImageData;  
  
public float[] ImageDataFloats;
```

5.2 Konfigurationsparameter

Bevor der Sensor in der Anlage verwendet werden kann, muss er parametrisiert werden. Nachfolgend sind die für den Sensor verwendeten Parameter aufgeführt.

In der Spalte "read (r)/ write (w)" wird angegeben, ob das Feature lesbar ("r"), schreibbar ("w") oder beides ("r/w") ist.

Config ID	Parameter ID	Parameter-name	Werte	read (r)/ write (w)	Beschreibung
Toolbar	0x640001	Trigger Laser	0,1	-/w	Manuellen Trigger durchführen.
	0x640002	Trigger-LED	0,1	-/w	LED-Trigger durchführen. HINWEIS: Bei aktiviertem Autotrigger wird ein Linienbild über die "Trigger-LED" ausgegeben.
	GetImage	Get Image	0,1	-/w	Letztes Bild erneut senden.
	0x6D0001	Teach	0,1	-/w	Belichtungszeit mit dem nächsten Trigger automatisch anpassen.

Config ID	Parameter ID	Parameter-name	Werte	read (r)/ write (w)	Beschreibung
General	0xCC0001	Custom Name	Char[32]	r/w	Benutzerdefinierter Name
	0x510001	AutoTrigger	0,1	r/w	De-/Aktivierung der zyklischen Bildaufnahme.
	0x680001	Exposure time	1 ... 10000	r/w	Einstellen der manuellen Belichtungszeit. Um die Belichtungszeit manuell einzustellen, muss die Funktion "Manuelle Belichtungszeit verwenden" (siehe "Menüpunkt Illumination" auf Seite 64) aktiviert werden. Durch Erhöhen des Wertes steigt die Belichtungszeit und damit die Bildhelligkeit. Werte unter 1000 μ s sind in den meisten Fällen geeignet.
	0xBE0001	Use manual exposure time	0,1	r/w	Bei Aktivierung wird die manuell eingestellte Belichtungszeit verwendet. Ist der Haken nicht gesetzt, wird die Belichtungszeit beim "Teach" automatisch geregelt
	0x100001	Flash time	1 ... 10000	r/w	Belichtungszeit des verwendeten Blitzlichts beim Triggern von LEDs in μ s (siehe 0x640002)
	0x9F0001	Object contrast	1 ... 100	r/w	Kontrastschwelle zur Erkennung der Laserlinie auf dem Objekt in % (voreingestellter Wert: 45)
	0xC90001	ROI min X	-3200 ... 3200	r/w	Der kleinste Wert auf der x-Achse des ROI (in mm).
	0xC90002	ROI max X	-3200 ... 3200	r/w	Der größte Wert auf der x-Achse des ROI (in mm).
	0xC90003	ROI min Z	1 ... 3200	r/w	Der kleinste Wert auf der z-Achse des ROI (in mm).
	0xC90004	ROI max Z	1 ... 3200	r/w	Der größte Wert auf der z-Achse des ROI (in mm).
	ImageTransferActive	Transfer images	0,1	r/w	Bilder automatisch übertragen. HINWEIS: kann die Auswertzeit erhöhen.

Beispiele für die Anwendung der Konfigurationsparameter

In diesen Beispielen werden folgende Deklarationen verwendet:

```
public static class Sr2DParameterIdentifier
{
    public static string Name_0x010001 = "0x010001";
    public static string Homepage_0x020001 = "0x020001";
    public static string Productname_0x030001 = "0x030001";
    public static string Version_0x070001 = "0x070001";
    public static string Firmware_0xC30001 = "0xC30001";
    public static string Autotrigger_0x510001 = "0x510001";
    public static string Exposuretime_0x680001 = "0x680001";
    public static string Usemanualexposuretime_0xBE0001 = "0xBE0001";
    public static string Flashtime_0x100001 = "0x100001";
    public static string Presentationmode_0xFD0001 = "0xFD0001";
    public static string Objectcontrast_0x9F0001 = "0x9F0001";
    public static string HighPositionResolution_0x280001 = "0x280001";
    public static string ROIminX_0xC90001 = "0xC90001";
    public static string ROImaxX_0xC90002 = "0xC90002";
    public static string ROIminZ_0xC90003 = "0xC90003";
    public static string ROImaxZ_0xC90004 = "0xC90004";
    public static string Triggerlaser_0x640001 = "0x640001";
    public static string TriggerLED_0x640002 = "0x640002";
    public static string Getimage_GetImage = "GetImage";
    public static string Teach_0x6D0001 = "0x6D0001";
    public static string CustomName_0xCC0001 = "0xCC0001";
    public static string ImageTransferActive = "ImageTransferActive";
}
```

Setzen des Autotrigger:

```
var setParamResult = await _vsxProtocolDriver.SetSingleParameterValue(1, 1, "General",
Sr2DParameterIdentifier.Autotrigger_0x510001, "1");
```



Hinweis!

In der derzeitigen Version werden die beiden ersten Parameter (SettingsVersion und ConfigVersion) immer auf 1 gesetzt.

Durchführung einer Flächenbildaufnahme durch Trigger-LED

```
var setParamResult = await _vsxProtocolDriver.SetSingleParameterValue(1, 1, "Toolbar",
Sr2DParameterIdentifier.TriggerLED_0x640002, "1");
```

Bildübertragung einschalten

```
var ret3 = await _vsxProtocolDriver.SetSingleParameterValue(1, 1, "General",
Sr2DParameterIdentifier.ImageTransferActive, "1");
```

Beispielprogramm

Im Package "VSX-Interface libraries..." befindet sich das Beispielprogramm "ExampleSr2D.cs". Es dient dazu die ersten Schritte zur Einbindung des SmartRunner Explorer zu vereinfachen. In diesem Beispielprogramm werden folgende Schritte durchgeführt:

- TCP/IP-Verbindungsaufbau
- Befehl zum Start des Dauer-Trigger
- Mit jedem Trigger werden die Liniendaten in einem Container gespeichert
- Eine bestimmte Anzahl (Anzahl muss ermittelt werden) von Container kann in der API zwischengespeichert werden
- Hole Liniendaten aus Container (Polling)
- Beispielzugriff auf die Liniendaten

Beispiele zur Verwendung der Programmierschnittstelle

1. Aufbau der TCP/IP-Verbindung

Hierbei werden zuerst über UDP alle verfügbaren Geräte aufgelistet und versucht sich mit dem ersten Gerät zu verbinden. Wenn über UDP kein Gerät gefunden wurde dann wird versucht eine Verbindung über eine feste IP-Adresse (in diesem Fall die Default-IP) aufzubauen.

```
//First discover devices via UDP and use the ip of the first device found
var sensors = await VsxProtocolDriver.UdpDeviceList();
if (sensors.Succ && sensors.DeviceList.Count > 0)
{
    // create a new VsxProtocolDriver instance
    _vsxProtocolDriver = VsxProtocolDriver.Init(sensors.DeviceList[0].IpAddress);
}
else //use fix ip address
{
    // create a new VsxProtocolDriver instance with fix ip address
    _vsxProtocolDriver = VsxProtocolDriver.Init("192.168.2.3");
}
// Connect with device
var connRes = await _vsxProtocolDriver.Connect();
```

2. Ausgabe von Liniendaten

Es wird gewartet bis der Container gefüllt ist. Dies kann zeitlich, wie in diesem Fall, auf 500 ms begrenzt werden. Diese Timeout-Zeit sollte je nach den zu empfangenen Daten eingestellt werden. Handelt es sich nur um eine Linie, dann sind ca. 40 ms ausreichend. Anschließend wird geschaut, ob der Container existiert und ob dieser Daten enthält. In diesem Fall, ob er Linien enthält. Die Daten der Linie werden anschließend in die Variable `singleLine` übergeben.

```
// try to get a data container within 500 ms
var dataltem = await _vsxProtocolDriver.GetDynamicContainer(500);
// dataltem.Container now contains an image/result/line package
if (dataltem.Succ && (dataltem.Container != null))
{
    // Check if there is a line message contained
    if (dataltem.Container.ContainsMessage("Line"))
    {
        var lineMessage = dataltem.Container.GetMessage("Line") as VsxLineMessage;
        // Get line from package
        var singleLine = lineMessage.Line;
    }
}
```



Hinweis!

Für den Aufbau der Daten der "Linie", siehe "Aufbau des Typs LineMessage" auf Seite 38.

3. Zugriff auf eine Linienkoordinate

Hierzu muss zuerst wie in Beispiel 2 der Container abgeholt werden. Anschließend kann über folgende Programmzeile auf eine beliebige Linienkoordinate zugegriffen werden. In diesem Fall ist es die erste Koordinate.

```
Coordinate coord = singleLine[0];
```

4. Ausgabe von Flächenbilddaten

Es wird auf den ersten Container gewartet. Dies kann zeitlich wie in diesem Fall auf 500 ms begrenzt werden. Diese Timeout-Zeit kann je nach den zu empfangenen Daten eingestellt werden. Anschließend wird geschaut, ob der Container existiert und ob dieser ein Bild enthält. Falls dieser ein Bild enthält, wird es in die Variable Bmp gespeichert und das Bild anschließend im png-Format auf dem Temp-Verzeichnis abgelegt.

```
// try to get a data container within 500 ms
var dataltem = await_vsxProtocolDriver.GetDynamicContainer(500);
// Check if dataltem.Container is now existing
if (dataltem.Succ && (dataltem.Container !=null))
{
// Check if there is an Image message contained
if (dataltem.Container.ContainsMessage("Image"))
{
var Img = dataltem.Container.GetMessage("Image") as
VsxImageDataRegionMessage;
var Bmp = Img.Image;
Bmp.Save("C:/Temp/Bild.png");
}
}
```

6 Software Vision Configurator

Die Inbetriebnahme und die Bedienung des Sensors erfolgen mit der Software Vision Configurator.

Vision Configurator ermöglicht die komfortable Bedienung des Sensors mithilfe einer übersichtlichen Bedienoberfläche. Zu den Standardfunktionen gehören z. B. die Herstellung einer Verbindung zum Sensor, die Parametrierung von Betriebsparameter, die Speicherung von Datensätzen, die Visualisierung von Daten und die Fehlerdiagnose.



Hinweis!

Im Vision Configurator sind bereits folgende Benutzerrollen mit unterschiedlicher Berechtigung vordefiniert.

Benutzerrechte und Passwort

Benutzerrechte	Beschreibung	Passwort
Default	Anzeige aller Informationen Konfiguration des Sensors Anlegen von Benutzern gleicher oder niedriger Stufe	Es wird kein Passwort benötigt
User	Anzeige aller Informationen Konfiguration des Sensors Anlegen von Benutzern gleicher oder niedriger Stufe	User
Admin	Anzeige aller Informationen Konfiguration des Sensors	Erfragen Sie das Admin-Passwort bei Pepperl+Fuchs

Tabelle 6.1 Abhängig von der jeweiligen Benutzerrolle haben die Benutzer unterschiedliche Zugriffs- und Verwaltungsrechte.



Netzwerkverbindung herstellen

Um eine Netzwerkverbindung mit dem Sensor herzustellen, gehen Sie wie folgt vor:

1. Versorgen Sie den Sensor mit Spannung.
2. Starten Sie die Software Vision Configurator.
3. Geben Sie den Benutzernamen und das Passwort ein.



Hinweis!

Weitere Schritte zur benutzerdefinierten Installation und der Installation weiterer Komponenten werden im Handbuch Vision Configurator beschrieben. Das Handbuch Vision Configurator finden Sie im Internet unter www.pepperl-fuchs.com.

6.1 Verbinden mit dem Vision Configurator



Sensor verbinden

Mit der Funktion "Auto detect" (1) kann nach angeschlossenen Sensoren gesucht werden. Gehen Sie dazu wie folgt vor:

1. Stellen Sie sicher, dass Sensor und PC/Laptop betriebsbereit sind und eine Ethernet-Verbindung besteht.
2. Klicken Sie auf die Schaltfläche "Auto detect (TCP/IP only)" (1), um nach angeschlossenen Sensoren zu suchen.

↳ Wird ein Sensor (2) erkannt, erscheint dieser im Übersichtsfenster.

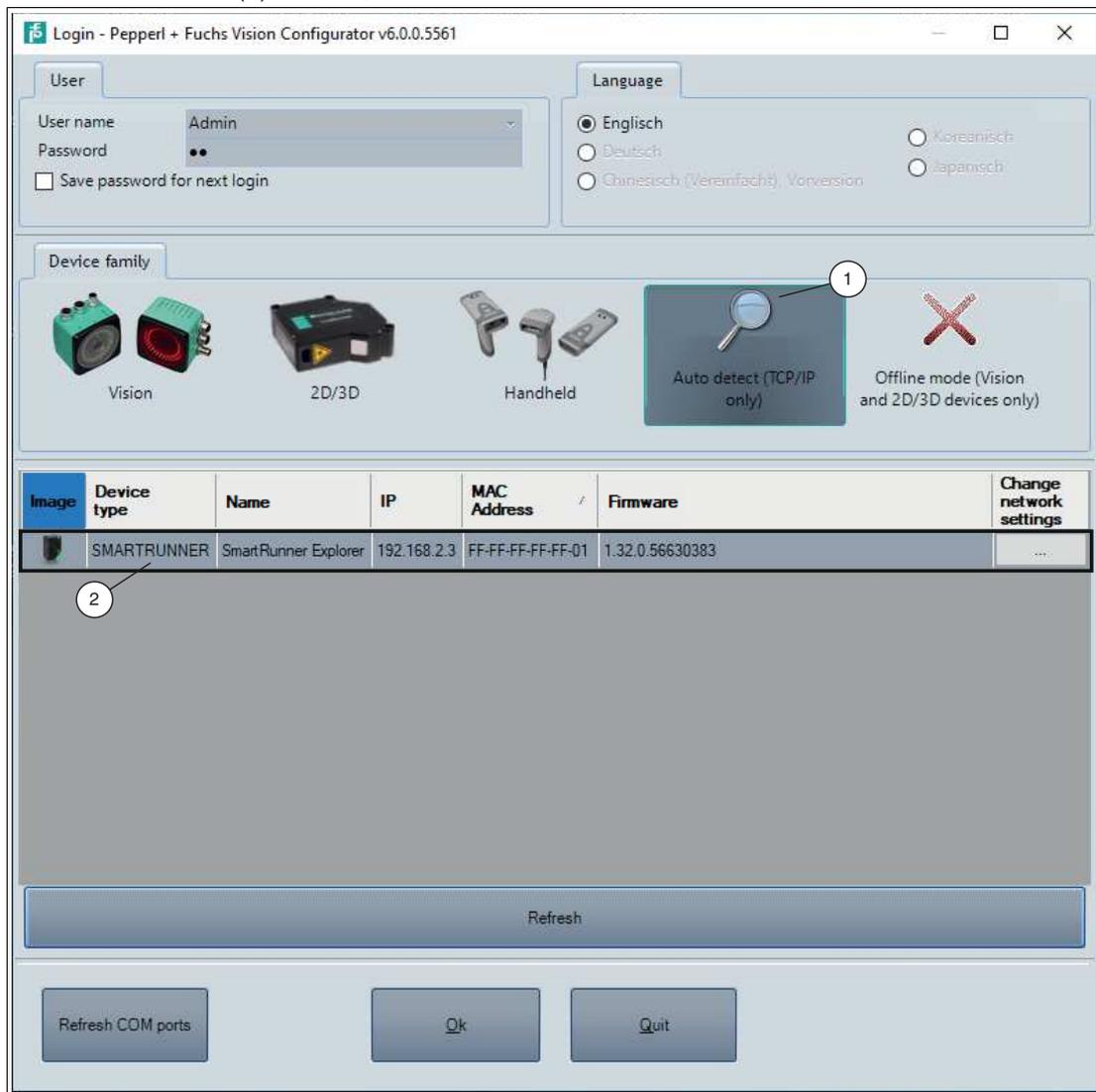


Abbildung 6.1 "Auto Detect" im Vision Configurator

3. Klicken Sie auf die Schaltfläche "Ok".
↳ Der Sensor (2) erscheint auf dem Startbildschirm aller gefundenen Sensoren.
4. Wählen Sie auf dem Startbildschirm unter der Registerkarte "Device family" die Schaltfläche "2D/3D" (1) an.

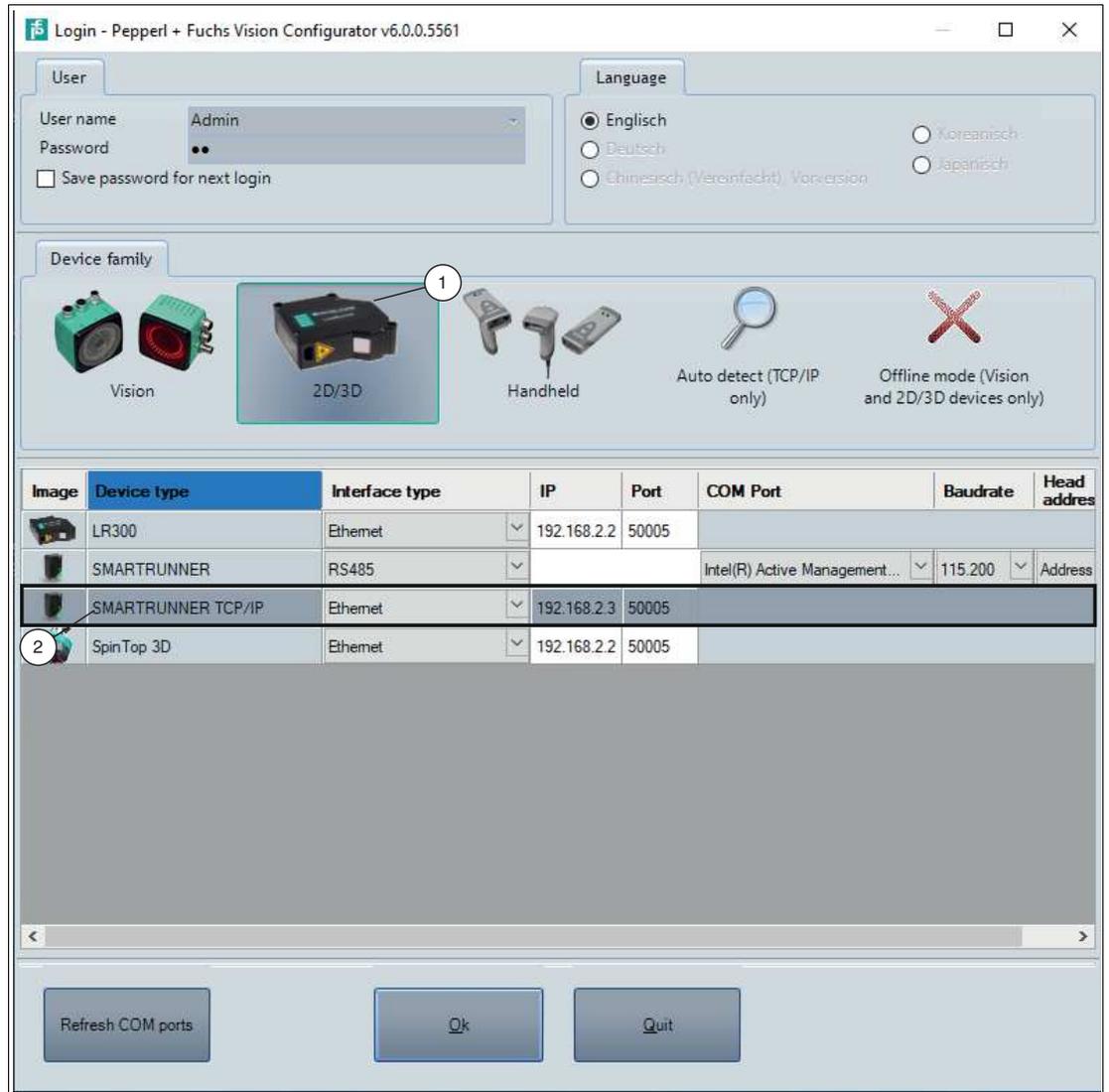


Abbildung 6.2 Startbildschirm

5. Wählen Sie den Sensor "SMARTRUNNER TCP/IP" (2) aus dem Übersichtsfenster.
6. Klicken Sie auf die Schaltfläche "Ok".

↳ Der Sensor ist verbunden und erscheint im Anwendungsfenster.

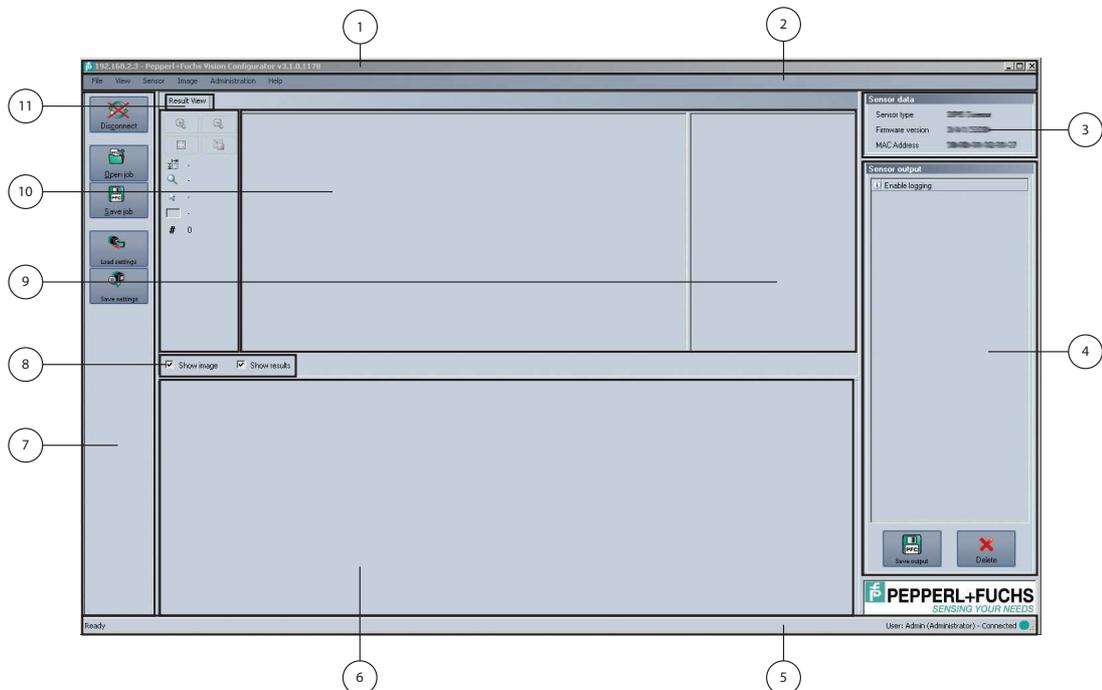
6.2 Aufbau des Anwendungsfensters

Nach erfolgreichem Login öffnet sich der Anwendungsbildschirm.



Hinweis!

Die einzelnen Funktionen sind abhängig vom angeschlossenen Sensortyp und aktueller Berechtigungsstufe und sind somit nicht immer alle sichtbar.



Die Software ist analog zu den meisten Windows-Applikationen aufgebaut.

1	Titelleiste	<ul style="list-style-type: none"> • Zeigt die IP-Adresse, die Softwarebezeichnung und die Versionsnummer an • enthält die Schaltflächen Minimieren / Maximieren / Schließen
2	Menüleiste	<ul style="list-style-type: none"> • zeigt alle Menüs des Programms an • dient als Übersicht und Navigation
3	Maske Sensor data	<ul style="list-style-type: none"> • zeigt die Sensordaten des angeschlossenen Sensors an
4	Maske Sensor output	<ul style="list-style-type: none"> • zeigt die Loganzeige an
5	Statusleiste	<ul style="list-style-type: none"> • zeigt die Statusinformationen zur Anwendung
6	Parametrierbereich	<ul style="list-style-type: none"> • enthält die sensorspezifischen Parameter, die Sie einstellen können
7	Symbolleiste	<ul style="list-style-type: none"> • enthält symbolische Schaltflächen als Erweiterung zum Menü
8	Kontrollkästchen	<ul style="list-style-type: none"> • Show images: Aktiviert oder deaktiviert die Bildanzeige • Show results: Aktiviert oder deaktiviert den Ergebnisbereich
9	Ergebnisbereich	<ul style="list-style-type: none"> • zeigt Ergebnisinformationen des Sensors • Es können abhängig vom angeschlossenen Sensor unterschiedlich viele Registerkarten angezeigt werden. • Dieses Feld kann mit dem Punkt Show results aktiviert oder deaktiviert werden

10	Bildanzeige	<ul style="list-style-type: none">• zeigt die aufgenommenen oder im Fehlerspeicher liegenden Bilder an• Dieses Feld kann mit dem Punkt Show images aktiviert oder deaktiviert werden
11	Registerkarte	Zeigt Informationen über aktuelles Bild und des sich unter der Maus befindlichen Pixels an. So werden folgende Punkte angezeigt: <ul style="list-style-type: none">• Bildgröße• Zoomstufe• Mausposition in Bildkoordinaten• aktueller Grauwert• Bildnummer

6.3 Menüleiste

In der Menüleiste werden verschiedene Menüfunktionen aufgeführt. Der Funktionsumfang ist abhängig vom angeschlossenen Sensortyp und von den Berechtigungen des angemeldeten Benutzers.



Abbildung 6.3 Menüleiste

6.3.1 Menü File

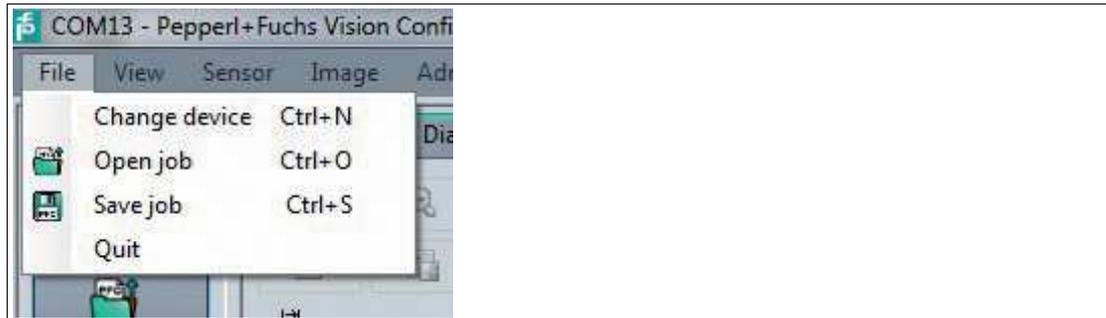


Abbildung 6.4 Menü File

Change device	Trennt die Verbindung zum Gerät und wechselt zurück zum Login-Dialog.
Open job	Lädt eine auf dem PC abgespeicherte Sensorkonfiguration.
Save job	Speichert die aktuelle Sensorkonfiguration auf dem PC.
Quit	Beendet das Programm.

Tabelle 6.2 Menü File

6.3.2 Menü View



Abbildung 6.5 Menü View

Show standard buttons	Schaltet die Anzeige der Buttons in der linken Leiste ein und aus.
Show sensor data	Schaltet die Anzeige der Sensordaten rechts oben aus.
Displayed message types...	Öffnet ein Auswahlfenster, in dem folgende Anzeigefenster aktiviert bzw. deaktiviert werden können: Info, Result OK, Result not OK, Warning, Error Critical, Assert

Tabelle 6.3 Menü View

6.3.3 Menü *Sensor*



Abbildung 6.6 Menü **Sensor**

Load settings	Lädt die gespeicherten Einstellungen aus dem Sensor
Save settings	Speichert die Einstellungen in den Sensor
Change network settings	Netzwerkeinstellungen ändern. In dem Einstellungsfenster können Sie die IP-Adresse, Subnetzmaske, Gatewayadresse und DHCP einstellen.
Make firmware update	Führt Firmwareupdates durch. Dieser Befehl sollte nur durch erfahrene Anwender benutzt werden.
Show device version	Zeigt die Geräteversion an
Sync with sensor	Synchronisation mit dem Sensor
Adjust sensor calibration	Kalibrierung des Sensors anpassen

Tabelle 6.4 Menü **Sensor**



Hinweis!

Firmwareupdate

Nachdem Sie die Firmware aktualisiert haben und **Update complete** angezeigt wird, starten Sie den Sensor neu.

6.3.4 Menü *Image*



Abbildung 6.7 Menü **Image**

Load imagefile	Bilddatei laden
Open image folder	Öffnet den Ordner, in dem aktuell Bilder abgespeichert werden.
Save image	Speichert das aktuell angezeigte Bild auf dem PC ab.
Copy image to clipboard	Lädt eine Bilddatei in den Zwischenspeicher.
Upload image to device	Bild auf das Gerät hochladen
Show graphic	Schaltet vom Sensor gesendete Anzeigedaten im Bild ein und aus.

Tabelle 6.5 Menü **Image**

6.3.5 Menü *Administration*



Abbildung 6.8 Menü **Administration**

User administration	Öffnet ein Fenster, in dem alle aktuell angelegten Benutzer gleicher oder niedrigerer Berechtigungsstufe angezeigt werden. Weiterhin können hier neue Benutzer gleicher oder niedrigerer Berechtigungsstufe angelegt oder gelöscht werden. Zusätzlich kann hier das Passwort eines Benutzers auf das Standardpasswort der jeweiligen Benutzerstufe zurückgesetzt werden.
Change password	Ändert das Passwort des aktuellen Benutzers.
Change user	Der Anmeldebildschirm wird geöffnet und es kann ein anderer Benutzer und / oder Sensor ausgewählt werden.
Send XML file...	Speichert die XML-Daten auf einem Computer.
Load XML file...	Lädt XML-Daten von einem Computer.
Create reader programming code	Erstellen einer Leser-Programmiercode

Tabelle 6.6 Menü **Administration**

6.3.6 Menü *Help*

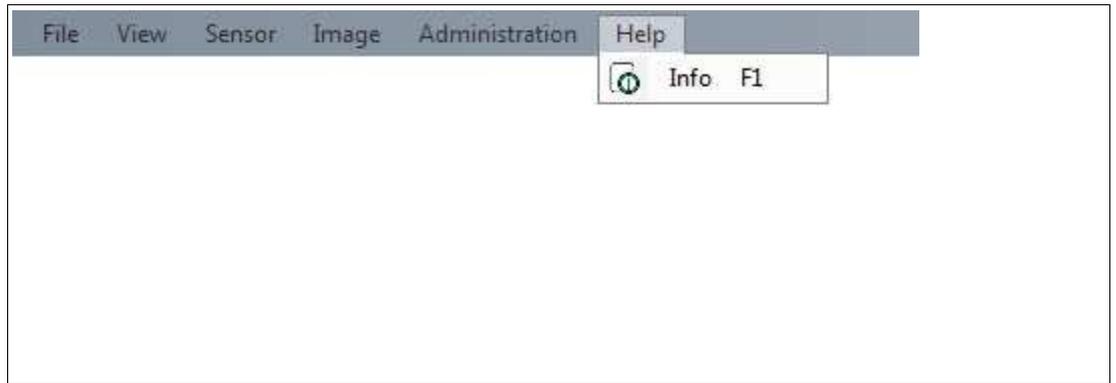


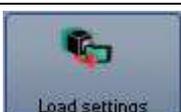
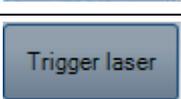
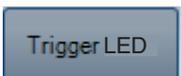
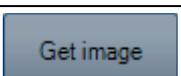
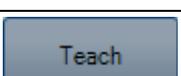
Abbildung 6.9 Menü **Help**

Info	Zeigt Informationen über den Vision Configurator an.
-------------	--

Tabelle 6.7 Menü **Help**

6.4 Symbolleiste

Über die Symbolleiste können verschiedene Funktionen angewählt werden.

	Beim Anwählen der Schaltfläche Connect, wird eine Verbindung zwischen PC und Sensor aufgebaut.
	Die Verbindung zwischen PC und Sensor wird getrennt.
	Öffnen einer abgespeicherten Einstellung.
	Speichert die vorgenommenen Einstellungen.
	Einstellungen werden aus dem Sensor ausgelesen.
	Alle vorgenommenen Einstellungen werden auf dem Sensor gespeichert.
	Zurücksetzen auf Standardeinstellungen.
	Laser-Trigger ausführen, um ein Linienbild aufzunehmen.
	LED-Trigger ausführen, um ein Flächenbild aufzunehmen. HINWEIS: Bei aktivierten Autotrigger wird mit dem "Trigger LED" ein Linienbild und kein Flächenbild ausgegeben.
	Aktuelles Sensorbild wird geladen.
	Belichtungszeit wird ausgeregelt.

6.5 Gerätedaten

Der Bereich Gerätedaten (Device data) zeigt den angeschlossenen Gerätetyp die Firmwareversion und die MAC-Adresse an.



Abbildung 6.10 Gerätedaten

6.6 Bild- und Linienanzeige

In der Bild- und Linienanzeige können Sie Aufnahmen auswerten. Durch die Analyse der aufgenommenen Profilform ist eine qualitative Aussage über das Messergebnis möglich. So können zum Beispiel störende Reflexionen erkannt und eliminiert werden. Des Weiteren gilt der Zusammenhang zwischen Belichtungszeit und Bildschärfe. Eine korrekte Belichtung ist abhängig von der Helligkeit des Profils und der einfallenden Lichtmenge. Eine zu kurze Belichtungszeit führt zu unterbelichteten (zu dunklen) Bildern, eine zu lange zu überbelichteten Bildern.

Um Fehler bei der Aufnahme zu vermeiden, stehen Ihnen unterschiedliche Möglichkeiten die aufgenommenen Daten anzuzeigen und anschließend zu korrigieren.

Result View - Linienbild

Unter der Registerkarte **Result View** können Sie das aktuell aufgenommene Bild inklusive des Höhenprofils öffnen. Dazu müssen Sie in der Symbolleiste die Schaltflächen **Trigger laser** anklicken.



Abbildung 6.11 Linienbild

- 1 In roter Farbe wird die gefundene Linie dargestellt.
- 2 In weißer Farbe wird die Laserbeleuchtung dargestellt.

Durch Betätigen der rechten Maustaste über das aufgenommene Bild erscheint folgendes Kontextmenü:



Abbildung 6.12 Bildanzeige Result View Kontextmenü

Bezeichnung	Funktion
Load image file...	Lädt ein Sensorbild. Das Sensorbild kann ausgewählt werden.
Open image folder	Öffnet den Speicherort
Copy image to clipboard	Bild in die Zwischenablage kopieren
Save image	Speichert das angezeigte Sensorbild

Result View - Flächenbild

Unter der Registerkarte **Result View** können Sie das aktuell aufgenommene Bild inklusive Höhenprofil öffnen. Dazu müssen Sie in der Symbolleiste die Schaltflächen **Trigger LED** anklicken.

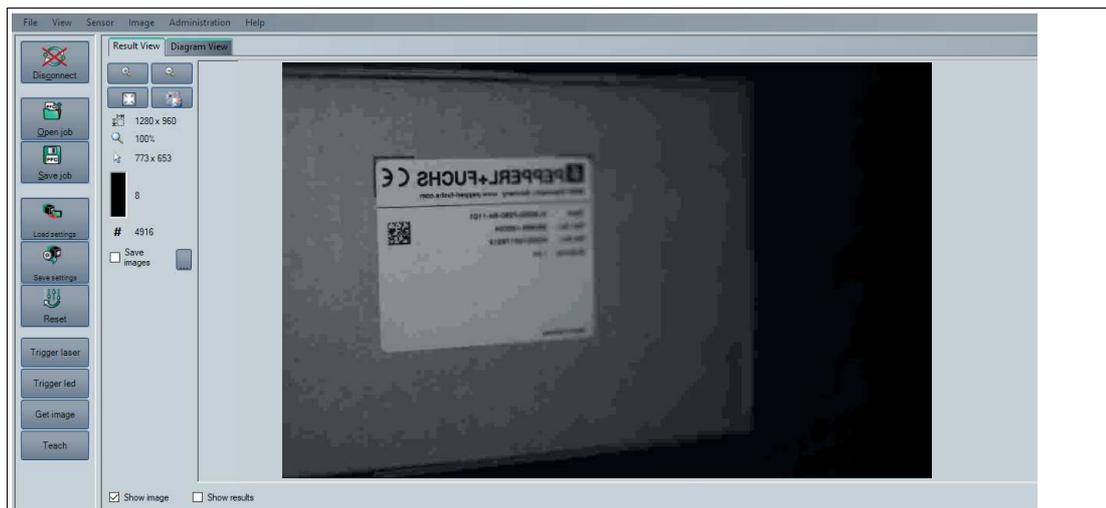


Abbildung 6.13 Flächenbild



Hinweis!

Beachten Sie, dass die Funktion "Image Transfer active" (siehe "Menüpunkt Image" auf Seite 64) aktiviert ist, damit das Flächenbild angezeigt wird.

Durch Betätigen der rechten Maustaste über das aufgenommene Bild erscheint folgendes Kontextmenü:



Abbildung 6.14 Bildanzeige Result View Kontextmenü

Bezeichnung	Funktion
Load image file...	Lädt ein Sensorbild. Das Sensorbild kann ausgewählt werden.
Open image folder	Öffnet den Speicherort
Copy image to clipboard	Bild in die Zwischenablage kopieren
Save image	Speichert das angezeigte Sensorbild

Result View - Werkzeugleiste

Die Werkzeugleiste befindet sich auf der linken Seite unter der Registerkarte **Result View**. In der Werkzeugleiste sind einige nützliche Funktionen, die zur weiteren Bearbeitung aufgenommener Bilder verwendet werden. Folgende Funktionen stehen Ihnen zur Verfügung.

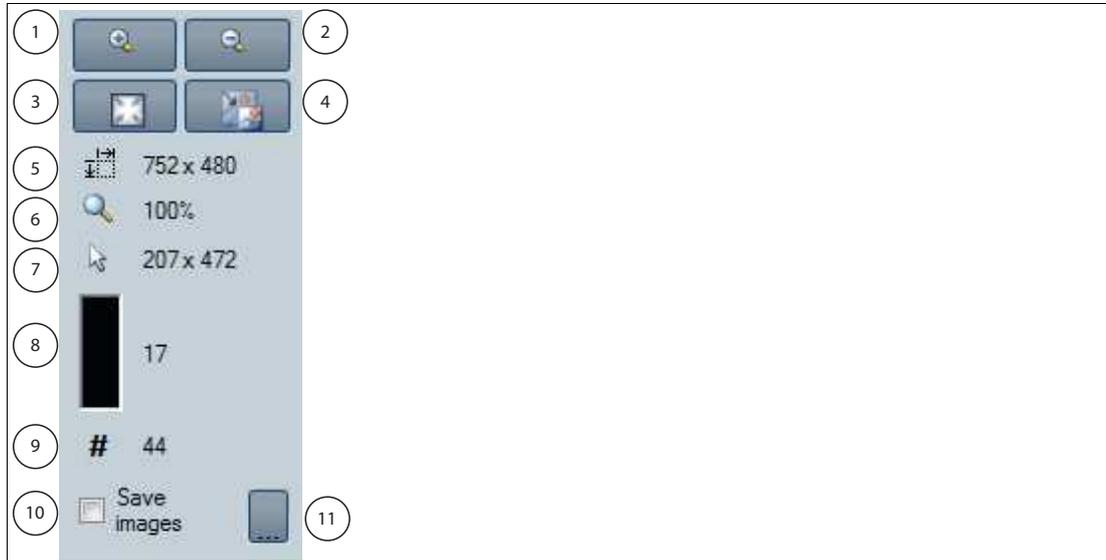


Abbildung 6.15 Werkzeugleiste

Position	Bezeichnung	Funktion
1	Lupe +	Bild vergrößern
2	Lupe -	Bild verkleinern
3	Fenster anpassen	Bildgröße im Fenster anpassen
4	Originalgröße	Originalbildgröße einstellen
5	Größenangabe	Informationsfeld Bildgröße
6	Zoomfaktor	Informationsfeld Zoomfaktor, Zoomfaktor 100% ist Originalbildgröße
7	Positionsangabe	Zeigt die Position des Mauszeigers
8	Grauwertangabe	Grauwertangabe des Pixels, auf dem der Mauszeiger steht
9	Bildzähler	Zeigt die aktuelle Bildnummer an
10	Bild speichern	Bild nach der Übertragung speichern
11	Pfad auswählen	Pfad auf dem Speichermedium auswählen

Diagram View

Unter der Registerkarte **Diagram View** können Sie die grafische Darstellung der Ergebnisdaten (Höhenprofil) öffnen. Dazu müssen Sie in der Symbolleiste die Schaltflächen **Trigger laser** anklicken. Anschließend kann über die Registerkarte **Diagram View** die grafische Darstellung abgerufen werden.

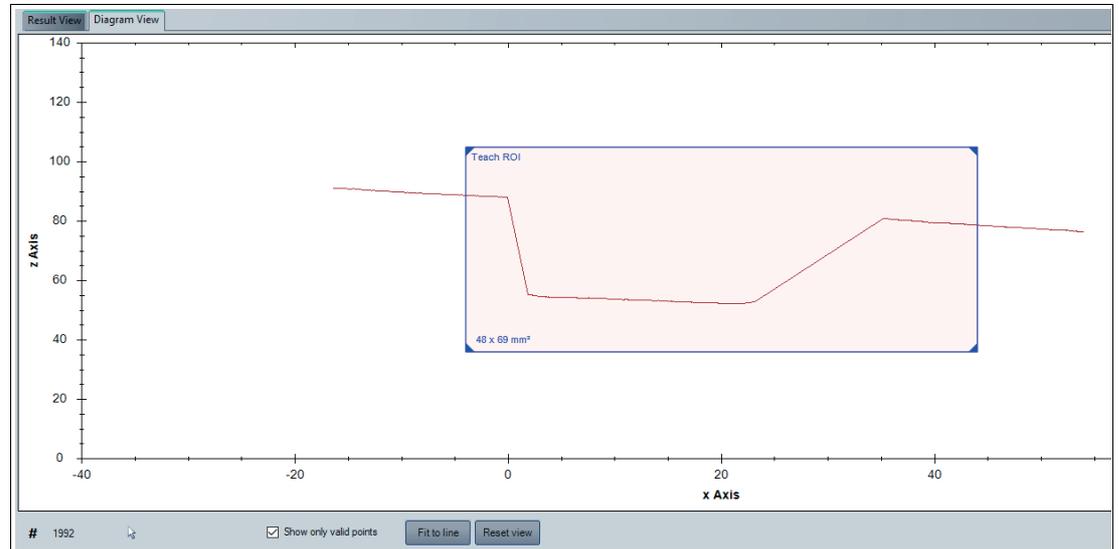


Abbildung 6.16 Diagram View

Durch Betätigen der rechten Maustaste über die grafische Darstellung erscheint folgendes Kontextmenü:



Abbildung 6.17 Bildanzeige Diagram View Kontextmenü

Bezeichnung	Funktion
Copy	Diagramm in Arbeitsspeicher kopieren
Save Image As...	Diagramm auf Festplatte speichern
Page Setup...	Seiteneinrichtung für die Druckfunktion
Print...	Diagramm drucken
Show Point Values	Zeigt Werte der diskreten Linienpunkte in Weltkoordinaten [mm] als Tooltip an.
Un-Zoom	Letzten Zoomvorgang rückgängig machen
Undo All Zoom/Pan	Alle Zoom- und Schwenkvorgänge rückgängig machen
Set Scale to Default	Skaliert den Maßstab anhand der Liniendaten

Diagram View - Werkzeugleiste

Die Werkzeugleiste befindet sich unterhalb der Diagrammanzeige. In der Werkzeugleiste sind einige nützliche Funktionen, die zur weiteren Bearbeitung der Diagramme verwendet werden. Folgende Funktionen stehen Ihnen zur Verfügung.

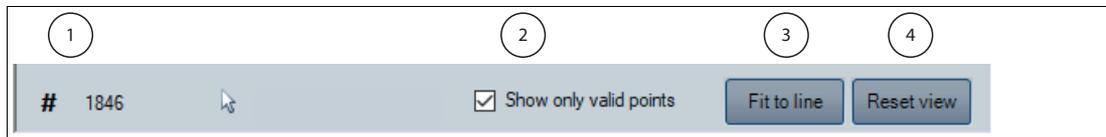


Abbildung 6.18 Diagram View - Werkzeugleiste

Position	Bezeichnung	Funktion
1	Grauwert	Grauwertangabe des Pixels
2	Show only valid points	Zeigt alle vorhandenen Punkte an
3	Fit to line	Gesamte sichtbare Linie im Blickfeld
4	Reset view	Setzt auf die Ursprungsansicht zurück

6.7 Parametrierbereich

In dem Parametrierbereich werden verschiedene Parameterangaben aufgeführt. Die einzelnen Parameter sind abhängig von der aktuellen Berechtigungsstufe und sind somit nicht immer alle sichtbar. Einige Features sind nur in verschiedenen Varianten verfügbar. Abhängig von den eingestellten Parametern werden einige Felder ausgegraut dargestellt.

6.7.1 Registerkarte Sensorinformation (Sensor information)

Unter der Registerkarte **Sensor information** steht Ihnen der Menüpunkte **Sensor information** zur Verfügung. Unter dem Menüpunkt **Sensor information** können Sie nähere Informationen zum Sensor einsehen. .

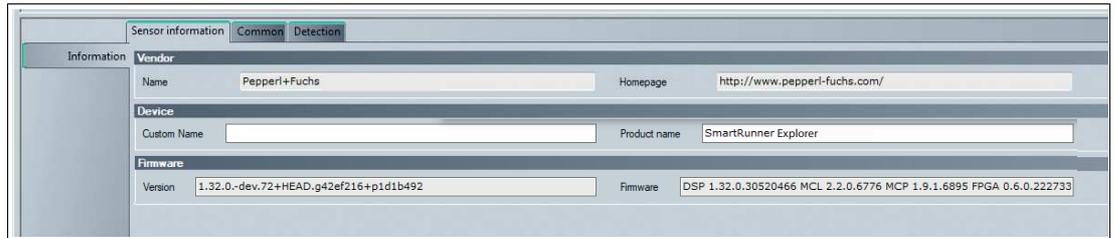


Abbildung 6.19 Registerkarte Sensorinformation (Sensor information)

Vendor	Name	Hersteller
	Homepage	Hersteller-Homepage
Device	Custom Name	Benutzerdefinierter Name
	Product Name	Produktbezeichnung
Firmware	Version	Firmwareversion
	Firmware	Firmwarebezeichnung

6.7.2 Registerkarte Common

Unter der Registerkarte **Common** stehen Ihnen 5 Menüpunkte zur Verfügung. In diesem Abschnitt werden die Menüpunkte im Detail erläutert.

Menüpunkt Illumination

Unter dem Menüpunkt **Illumination** können Sie die Belichtung des Sensors einstellen.

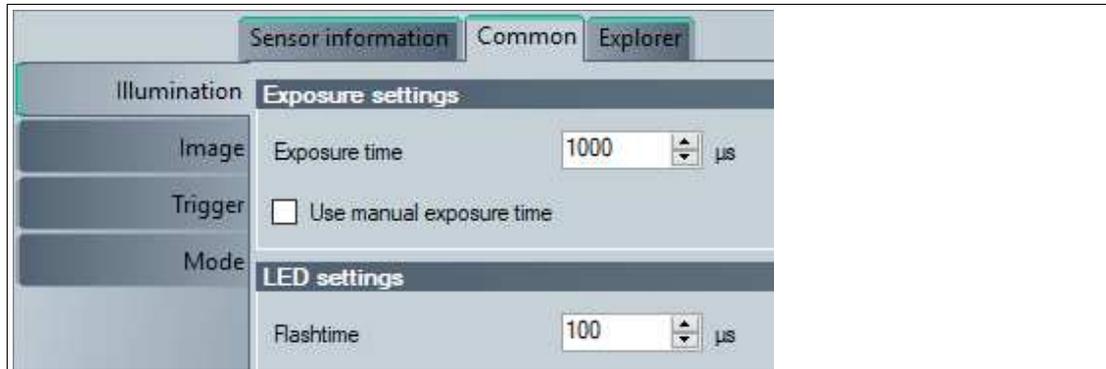


Abbildung 6.20 Menüpunkt Illumination

Bezeichnung	Funktion
Exposure time	Einstellung der manuellen Belichtungszeit. Um die Belichtungszeit manuell einzustellen, muss die Funktion "Use manual exposure time" aktiviert sein. Durch Erhöhung des Wertes nimmt die Belichtungszeit und damit die Bildhelligkeit zu. In den meisten Fällen sind Werte unter 1000 µs geeignet
Use manual exposure time	Bei Aktivierung wird die manuell eingestellte Belichtungszeit verwendet. Ist der Haken nicht gesetzt, wird die Belichtungszeit beim "Teach" automatisch geregelt
Flashtime	Einstellung der Belichtungszeit der LED-Beleuchtung in µs.

Menüpunkt Image

Unter dem Menüpunkt **Image** können Sie die Bildaufnahme einstellen.

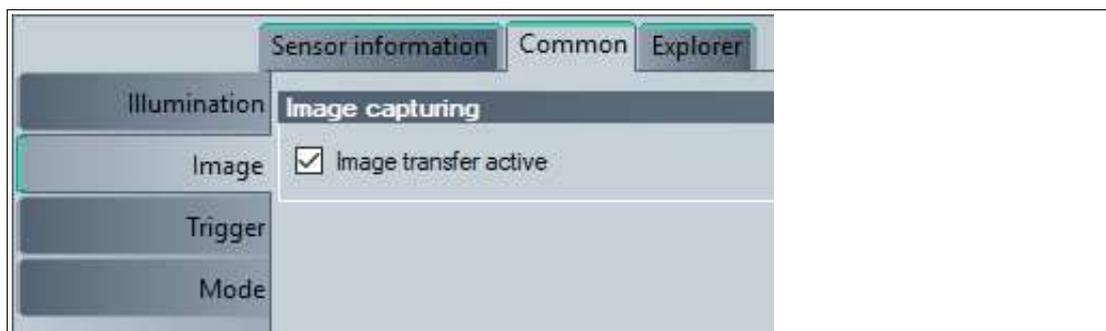


Abbildung 6.21 Menüpunkt Image

Bezeichnung	Funktion
Image transfer active	Mit dieser Funktion werden bei jeder Bildaufnahme automatisch Bilder übertragen. Wenn die Funktion deaktiviert wird, werden keine Bilder im "Result View" ausgegeben.

Menüpunkt Trigger

Unter dem Menüpunkt **Trigger** können Sie den Autotrigger aktivieren bzw. deaktivieren.



Abbildung 6.22 Menüpunkt Trigger

Bezeichnung	Funktion
Autotrigger	Mit gesetztem Häkchen wird ein zyklisch wiederkehrender Trigger aktiviert. Im "Präsentationsmodus" muss der Autotrigger aktiviert sein. HINWEIS: Bei aktivierten Autotrigger wird mit dem "Trigger LED" ein Linienbild und kein Flächenbild ausgegeben.

Menüpunkt Mode

Unter dem Menüpunkt **Mode** können Sie den "Presentation mode" und die "Funktionstasten 1 und 2" aktivieren bzw. deaktivieren. Mit gesetztem Häkchen sind der "Presentation mode" und die "Funktionstasten 1 und 2" aktiviert, ohne Häkchen deaktiviert.

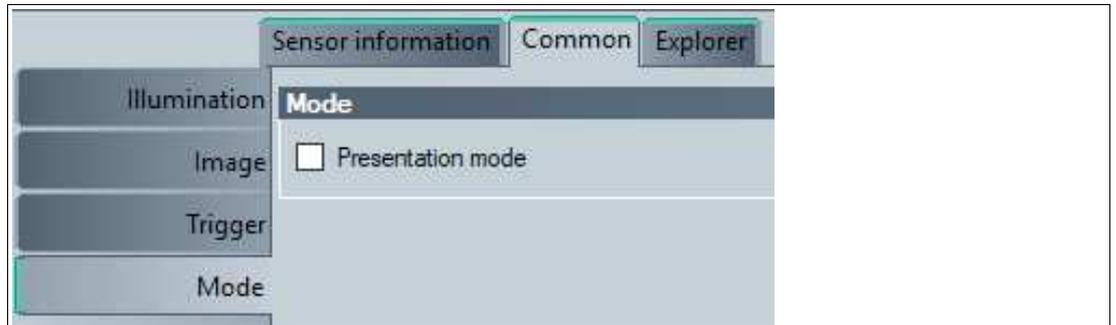


Abbildung 6.23 Menüpunkt Mode

Bezeichnung	Funktion
Presentation mode	Betriebsmodus für die Präsentation bzs. Test ohne Zuhilfenahme eines PC

6.7.3 Registerkarte Explorer

Unter der Registerkarte **Matcher** stehen Ihnen 2 Menüpunkte zur Verfügung. In diesem Abschnitt werden die Menüpunkte im Detail erläutert.

Menüpunkt Line

Unter dem Menüpunkt **Line** können Sie den x- und z-Toleranzbereich des Sensors einstellen. Der Toleranzbereich gibt an, um welchen Wert sich das zu erkennende Höhenprofil innerhalb des Auswertebereichs verschieben darf.

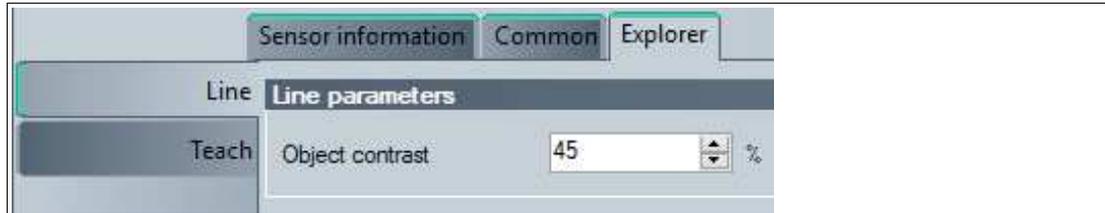


Abbildung 6.24 Menüpunkt Match

Bezeichnung	Funktion
Object contrast	Kontrastschwelle, die für die Erkennung der Laserlinie auf dem Objekt benutzt wird.

Menüpunkt Teach

Der Menüpunkt **Teach** wird zur Einstellung der Belichtungszeit genutzt. Der benötigte Einlernbereich "Teach ROI" wird anhand des Linienprofils unter der Registerkarte "Diagram View" eingestellt. Die Koordinaten der x- und z-Achse werden im Anzeigefeld unterhalb der Grafik dargestellt.

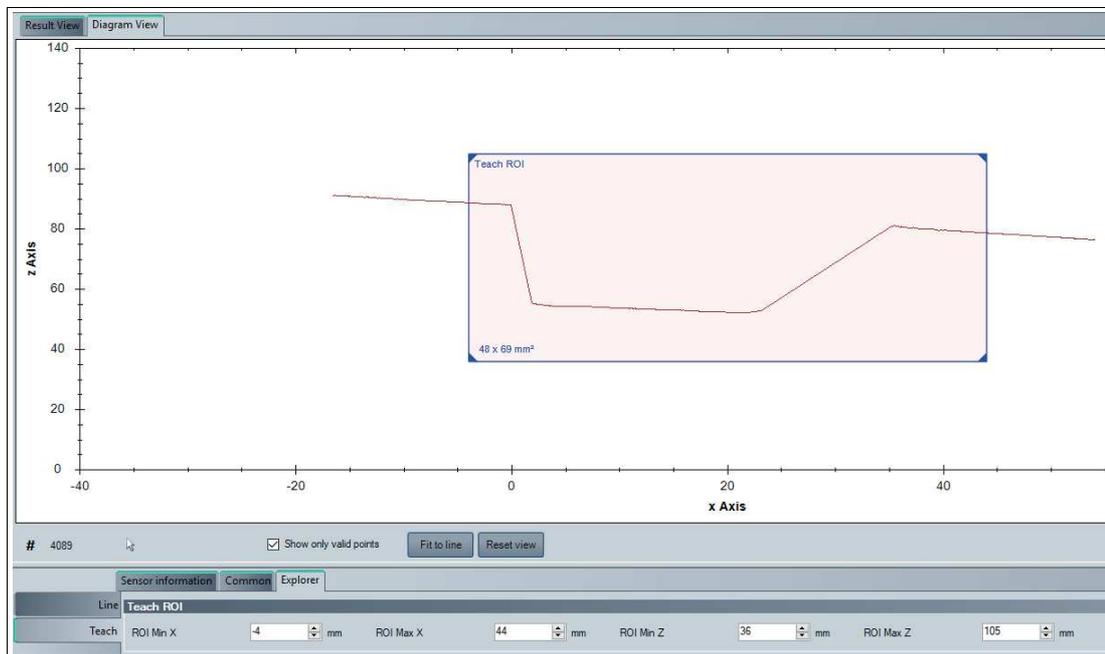


Abbildung 6.25 Menüpunkt Teach

Bezeichnung	Funktion
ROI Min X	der kleinste Wert auf der x-Achse
ROI Max X	der größte Wert auf der x-Achse
ROI Min Z	der kleinste Wert auf der z-Achse
ROI Max Z	der größte Wert auf der z-Achse

2022-01

7 **Wartung und Reparatur**

7.1 **Wartung**



Gefahr!

Lebensgefahr durch elektrischen Strom!

Bei Berührung mit spannungsführenden Teilen besteht unmittelbare Lebensgefahr.

- Arbeiten an der elektrischen Anlage nur von Elektrofachkräften ausführen lassen.
 - Vor Wartungs-, Reinigungs- und Reparaturarbeiten Stromversorgung abschalten und gegen Wiedereinschalten sichern.
 - Feuchtigkeit von spannungsführenden Teilen fernhalten.
-

Das Gerät ist wartungsfrei. Um die bestmögliche Geräteleistung zu erzielen, halten Sie die Optikeinheit des Gerätes sauber und reinigen Sie diese bei Bedarf.

Beachten Sie bei der Reinigung folgende Hinweise:

- Berühren Sie die Optikeinheit nicht mit den Fingern.
- Tauchen Sie das Gerät nicht in Wasser ein. Besprühen Sie es nicht mit Wasser oder anderen Flüssigkeiten.
- Verwenden Sie zum Reinigen der Geräteoberfläche keine Scheuermittel.
- Benutzen Sie ein Baumwoll- oder Zellstofftuch, das mit Wasser oder Isopropylalkohol angefeuchtet (nicht getränkt) ist.
- Entfernen Sie Alkoholrückstände mit einem Baumwoll- oder Zellstofftuch, das mit destilliertem Wasser angefeuchtet (nicht getränkt) ist.
- Wischen Sie die Geräteoberflächen mit einem fusselfreien Tuch trocken.

7.2 **Reparatur**

Das Gerät darf nicht repariert, verändert oder manipuliert werden. Ersetzen Sie das Gerät im Fall eines Ausfalls immer durch ein Originalgerät.

8 Störungsbeseitigung

8.1 Was tun im Fehlerfall

Bevor Sie das Gerät reparieren lassen, führen Sie folgende Maßnahmen durch:

Checkliste

Fehler	Ursache	Behebung
LED "Power" leuchtet nicht	Die Stromversorgung ist abgeschaltet	Ermitteln Sie, ob es einen Grund für die Abschaltung gibt (Installationsarbeiten, Instandhaltungen, usw.). Schalten Sie ggf. die Stromversorgung ein.
	Verdrahtungsfehler im Verteiler oder Schaltschrank, Kabelbruch	Überprüfen Sie sorgfältig die Verdrahtung und beheben Sie ggf. vorhandene Verdrahtungsfehler. Prüfen Sie die Kabel auf Funktion.
Steuerung empfängt keine Messdaten	Anschlusskabel nicht verbunden	Verbinden Sie das Anschlusskabel.
	Falsches Anschlusskabel verwendet	Verwenden Sie ausschließlich das passende Anschlusskabel.
Messobjekt nicht erkannt	Reflexionen	Reflexionen vermeiden
	Fremdbelichtung	Fremdbelichtung vermeiden
	Belichtungsregelung	Belichtung einstellen (siehe Kapitel 6.7.2)
	Einlernbereich falsch eingestellt	Einlernbereich einstellen (siehe Kapitel 6.7.3)
Messfehler	Oberflächen mit ausgeprägter Riefenstruktur und spiegelnde Oberflächen	bessere Anordnung der Sensorkomponenten zum Messobjekt
	Temperaturänderung im Sensor	Sensor ca. 15 Minuten warmlaufen lassen, bevor der Messvorgang gestartet wird.
	Falscher Abstand zum Messobjekt	Abstandsangaben beachten
	Gehäuses falsch befestigt	Gehäuse korrekt montieren (siehe Kapitel 3.4)
Präsentationsmodus funktioniert nicht	Präsentationsmodus nicht aktiviert	Presentation mode und Autotrigger aktivieren und mit "Save settings" bestätigen
Keine Verbindung zum Sensor	Wechselspannung oder zu hohe Versorgungsspannung	Sensor ausschließlich an Gleichspannung (DC) anschließen. Stellen Sie sicher, dass die Höhe der Versorgungsspannung im spezifizierten Bereich des Sensors liegt.

- Falls keiner der vorherigen Punkte zum Ziel geführt hat, nehmen Sie bitte Kontakt zu unserem Service-Center auf. Halten Sie hier bitte die Fehlerbilder und die Versionsnummer der Firmware bereit. Die Firmware-Versionsnummer finden Sie auf der Bedienoberfläche oben rechts.

Your automation, our passion.

Explosionsschutz

- Eigensichere Barrieren
- Signaltrenner
- Feldbusinfrastruktur FieldConnex®
- Remote-I/O-Systeme
- Elektrisches Ex-Equipment
- Überdruckkapselungssysteme
- Bedien- und Beobachtungssysteme
- Mobile Computing und Kommunikation
- HART Interface Solutions
- Überspannungsschutz
- Wireless Solutions
- Füllstandsmesstechnik

Industrielle Sensoren

- Näherungsschalter
- Optoelektronische Sensoren
- Bildverarbeitung
- Ultraschallsensoren
- Drehgeber
- Positioniersysteme
- Neigungs- und Beschleunigungssensoren
- Feldbusmodule
- AS-Interface
- Identifikationssysteme
- Anzeigen und Signalverarbeitung
- Connectivity

Pepperl+Fuchs Qualität

Informieren Sie sich über unsere Qualitätspolitik:

www.pepperl-fuchs.com/qualitaet

