BAUMER AG
VeriSens Provider

Version 1.0.2

User’s guide

June 24, 2013

NOTES:

[Revision history]

Version	Date	Contents
1.0.0	2012-8-02	First edition.
1.0.2	2013-6-24	TimeOut option included

[Hardware]

Model	Version	Notes
VeriSens ID Series		
VeriSens CS Series		
VeriSens XF Series		
VeriSens XC Series		

Contents

1. Introduction

This documentation describes how to use the VeriSens Provider for an easily communication with a VeriSens Smart Camera from BAUMER. The Provider can be used with a PC and high level languages or directly using the RC8 controller with the DENSO PacScript language.

The connection with the VeriSens is based on the TCP/IP Ethernet Communication Interface. The Provider is acting as a Client and the VeriSens camera as a Server waiting for commands.

2. Outline of provider

2.1. Outline

The VeriSens Provider offers one execution method by CaoController::Execute to send parameters and receive current data values using the VeriSens TCP/IP Ethernet Communication Interface.

Table 2-1 Baumer VeriSense provider

File name	BaumerVeriSens.dll
ProgID	CaoProv.BAUMER.VeriSens
Registry registration	regsvr32 [PATH]/ BaumerVeriSens.dll
Registry un-registration	regsvr32 /u [PATH]/ BaumerVeriSens.dll

2.2. Method and property

The VeriSens provider connects at the time when the method AddConntroller.is executed

2.3. CaoWorkspace::AddController method

Syntax

AddController (< bstrCtrlName:VT_BSTR > and < bstrProvName:VT_BSTR >
<bstrPcName:VT_BSTR > [,<bstrOption:VT_BSTR>])

bstrCtrlName :[in] Controller name arbitrary.

bstrProvName : [in] Provider name. (Fixed to " BAUMER.VeriSens")

bstrPcName : [in] Execution machine name of provider

bstrOption : [in] Option character string

Following is a list of option string items.

Option	Meaning
Conn=< connection parameter >	Set the communication form and connection parameters
TimeOut[=<Timoout period>]	Specify time-out time (millisecond) for sending and receiving (default:infinite)

Conn option

The following describes the connection parameter string for the Conn option for Ethernet.

"eth:<IP Address>[:<Port No>]"

<IP Address> : Mandatory. Specify IP address.
Example:"192.168.0.1"

<Port No> : Specify the port number to communicate:
Example:"192.168.0.1:23"
The standard port number is 23

2.4. CaoController::Execute method

The command name is specified in the first argument and the parameter of the command is specified for the second argument. Please refer to Chapter 3 Command reference for details of each command.

Syntax Execute (<bstrCommandName:VT_BSTR>, [<vntParam: VT_VARIANT])

bstrCommandName	:	[in] Command name (VT_BSTR)
vntParam	:	[in] Parameter
Return value	:	[out] Return value of command (VT_R4 | VT_ARRAY or VT_BSTR)

3. Command reference

3.1. Controller class

Table 3-1 CaoController::Execute command list

Command	Function	Page
IsConnected	Get info about the connection status	9
IsSetupMode		9
IsRunMode		9
RebootDevice	Restart the camera	9
GetCurrentJob	Get the current job number	10
ClearStatistics	Clear the statistics of the active job	10
GetData	Get the latest datas from the camera	10
GetDataXYR	Get the latest detection datas X,Y, and Rotation **(Refer Chapter 5)**	10
GetImage	Get the live image	11
DoTrigger	Trigger a new image	11
DoTriggerAndGetData	Trigger an image and receive datas directly	11
SwitchToSetupMode		12
SwitchToRunMode		12

3.2. CaoController::Execute("IsConnected") command

Get the current connection status.

Syntax IsConnected ()
Argument : None
Return Value : VT_BOOL

Example bool connected = false
connected = CaoCtrl.Execute("IsConnected")

3.3. CaoController::Execute("IsSetupMode") command

Syntax IsSetupMode ()
Argument : None
Return Value : VT_BOOL

Example bool mode = false
mode = CaoCtrl.Execute("IsSetupMode")

3.4. CaoController::Execute("IsRunMode") command

Syntax IsRunMode ()
Argument : None
Return Value : VT_BOOL

Example bool mode = false;
mode = CaoCtrl.Execute("IsRunMode")

3.5. CaoController::Execute("RebootDevice") command

Reboot the camera system.

Syntax RebootDevice()
Argument : None
Return Value : None

Example caoCtrl.Execute("RebootDevice")

3.6. CaoController::Execute("GetCurrentJob") command

Returns the current job number.

Syntax GetCurrentJob()
Argument : None
Return Value : VT_I2

Example short jobNumber = CaoCtrl.Execute("GetCurrentJob")

3.7. CaoController::Execute("ClearStatistics") command

Clears the statistics of the current job.

Syntax ClearStaticstics ()
Argument : None
Return Value : None

Example CaoCtrl.Execute("ClearStaticstics")

3.8. CaoController::Execute("GetData") command

Get the current data defined in the "Output process interface" in the VeriSens Application Suite software.

Syntax GetData ()
Argument : None
Return Value : VT_BSTR

Example string data = string.empty;
data = CaoCtrl.Execute("GetData")

3.9. CaoController::Execute("GetDataXYR") command

Get the position and rotation data of a detected item. If no item is detected, the return value would be (0,0,0)

Syntax GetDataXYR ()
Argument : None
Return Value : Detection values (VT_R4 | VT_ARRAY)

Example ReceiveValues = CreateArray(3, VT_R4)

```
ReceiveValues(0) = 0
ReceiveValues(1) = 0
ReceiveValues(2) = 0
ReceiveValues = BaumerController.Execute("GetDataXYR", "")
ebug.Print ReceiveValues(0) ‘ x position
Debug.Print ReceiveValues(1) ‘y position
Debug.Print ReceiveValues(2) ‘ rotation
```

3.10. CaoController::Execute(“GetImage”) command

Get the live image

Syntax GetImage ()

Argument : None

Return Value : Detection values (VT_R4 | VT_ARRAY)

Example

3.11. CaoController::Execute(“DoTrigger”) command

Trigger a new image. Notice that the camera has to set in trigger mode.

Syntax DoTrigger ()

Argument : None

Return Value : None

Example caoCtrl.Execute(“DoTrigger”)

3.12. CaoController::Execute(“DoTriggerAndGetData”) command

Trigger a new image and receive the datas immediately. Notice that the camera has to set in trigger mode.

Syntax DoTriggerAndGetData ()

Argument : None

Return Value : VT_BSTR

Example

```
string data = string.empty;
data = caoCtrl.Execute(“DoTriggerAndGetData”)
```

3.13. CaoController::Execute("SwitchToSetupMode") command

Switch to the setup mode to configure the camera with the VeriSens Application Suite software.

Syntax	SwitchToSetupMode ()		
	Argument	:	None
	Return Value	:	None
Example	caoCtrl.Execute("SwitchToSetupMode")		

3.14. CaoController::Execute("SwitchToRunMode") command

Switch from the Setup Mode to the Run Mode

Syntax	SwitchToRunMode ()		
	Argument	:	None
	Return Value	:	None
Example	caoCtrl.Execute("SwitchToRunMode")		

4. Sample program

List 4-1 BaumerExample.pcs

```
'!TITLE "Denso robot program"
#define VT_R4                   4
Sub Main
        Takearm Keep = 1
        Dim SendValues As Variant
        Dim ReceiveValues As Variant
        Dim BaumerController As Object
        BaumerController = CAO.AddController("Baumer","CaoProv. BAUMER.VeriSens ", "",
Conn=eth:172.20.57.42:23")

        ReceiveValues = CreateArray(3, VT_R4)
        ReceiveValues(0) = 0
        ReceiveValues(1) = 0
        ReceiveValues(2) = 0

    Do
        ReceiveValues = BaumerController.Execute("GetDataXYR", "")

        Debug.Print "Received Values"
        Debug.Print ReceiveValues(0)
        Debug.Print ReceiveValues(1)
        Debug.Print ReceiveValues(2)
    Loop
End Sub
```

5. Preparing the Baumer VeriSens Camera for the GetDataXYR function

5.1. Prepare the device settings

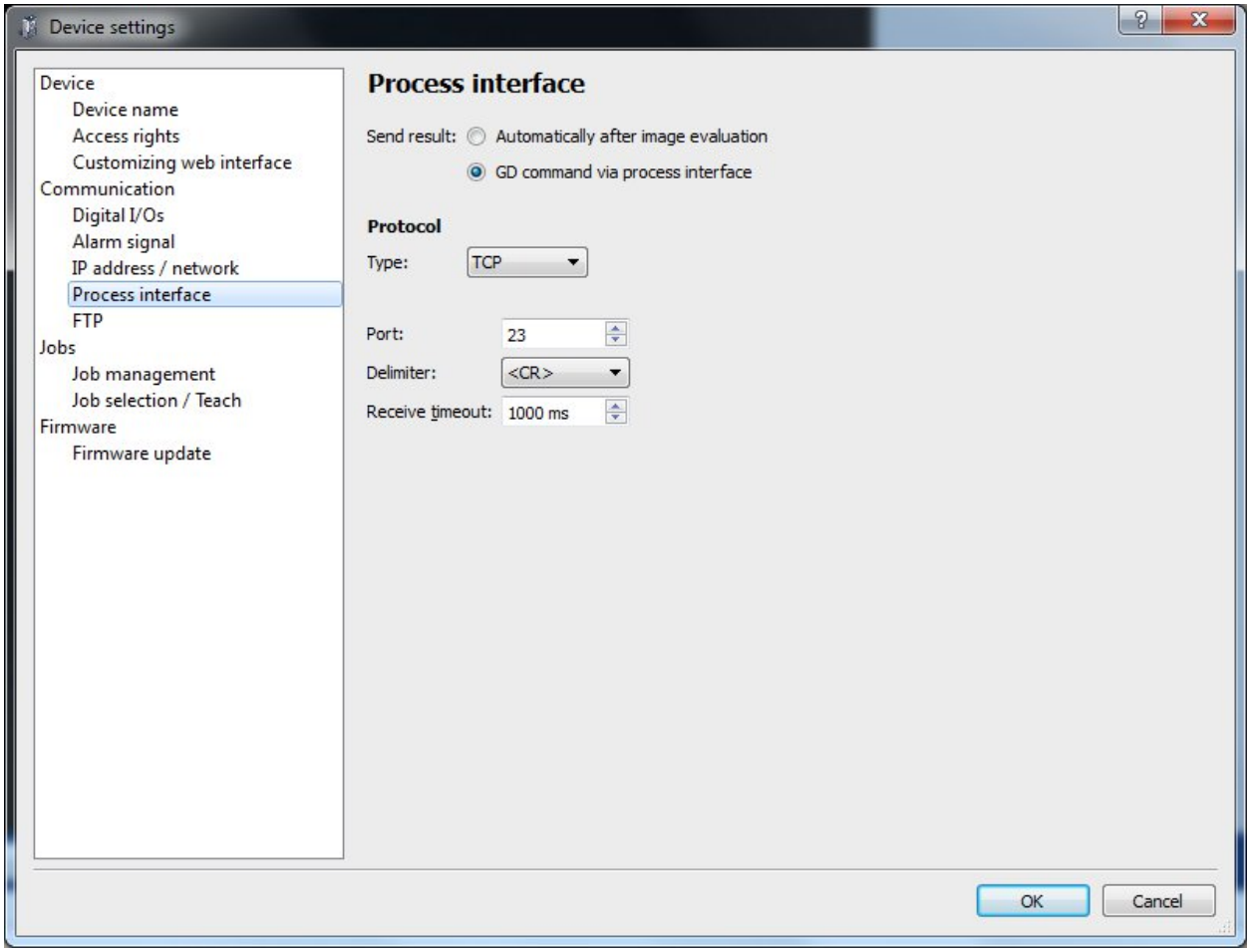
Make sure that the results of the image processing are send with an GD command process. You will find the settings under "Device" -> "Device Settings". Please be sure that you have the following settings:

Protocol setting

Type : TCP

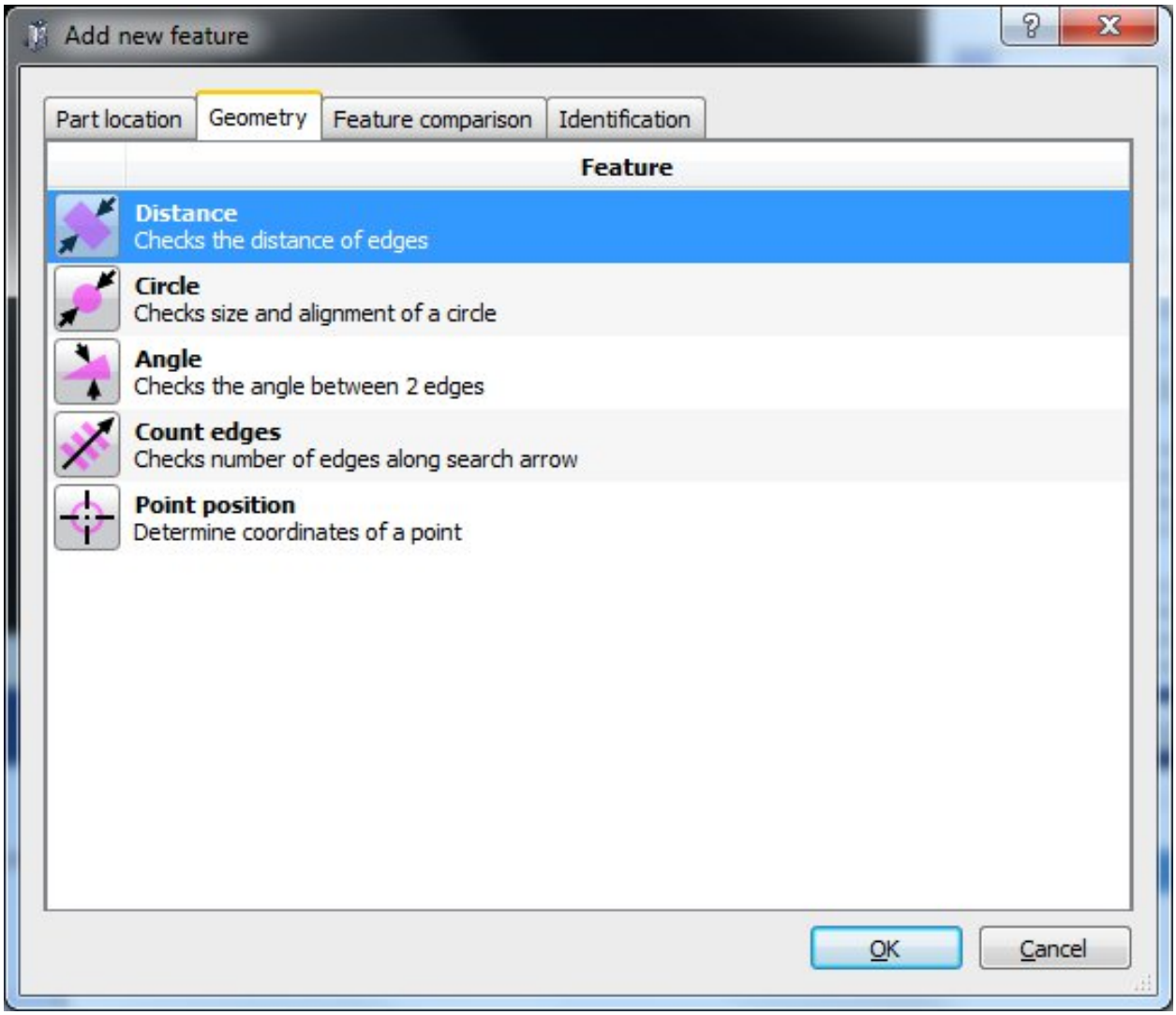
Port : 23

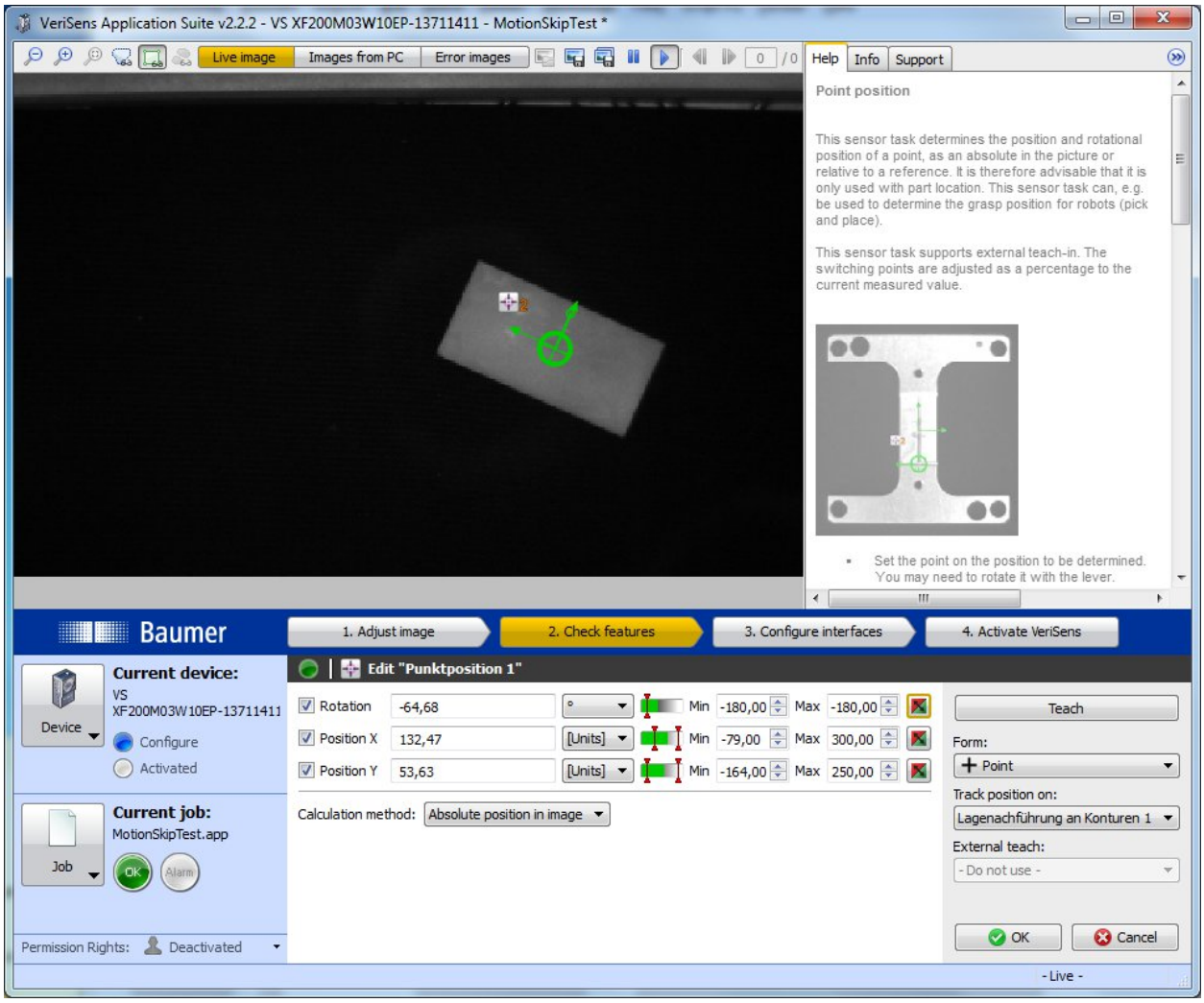
Delimiter: <CR>



5.2. Set point position of the detected part for the GetDataXYR function

Make sure you have to determine the coordinates of a point. For example the middle point of a detected part. To do this, you have to include the "Point position" feature to your current job.



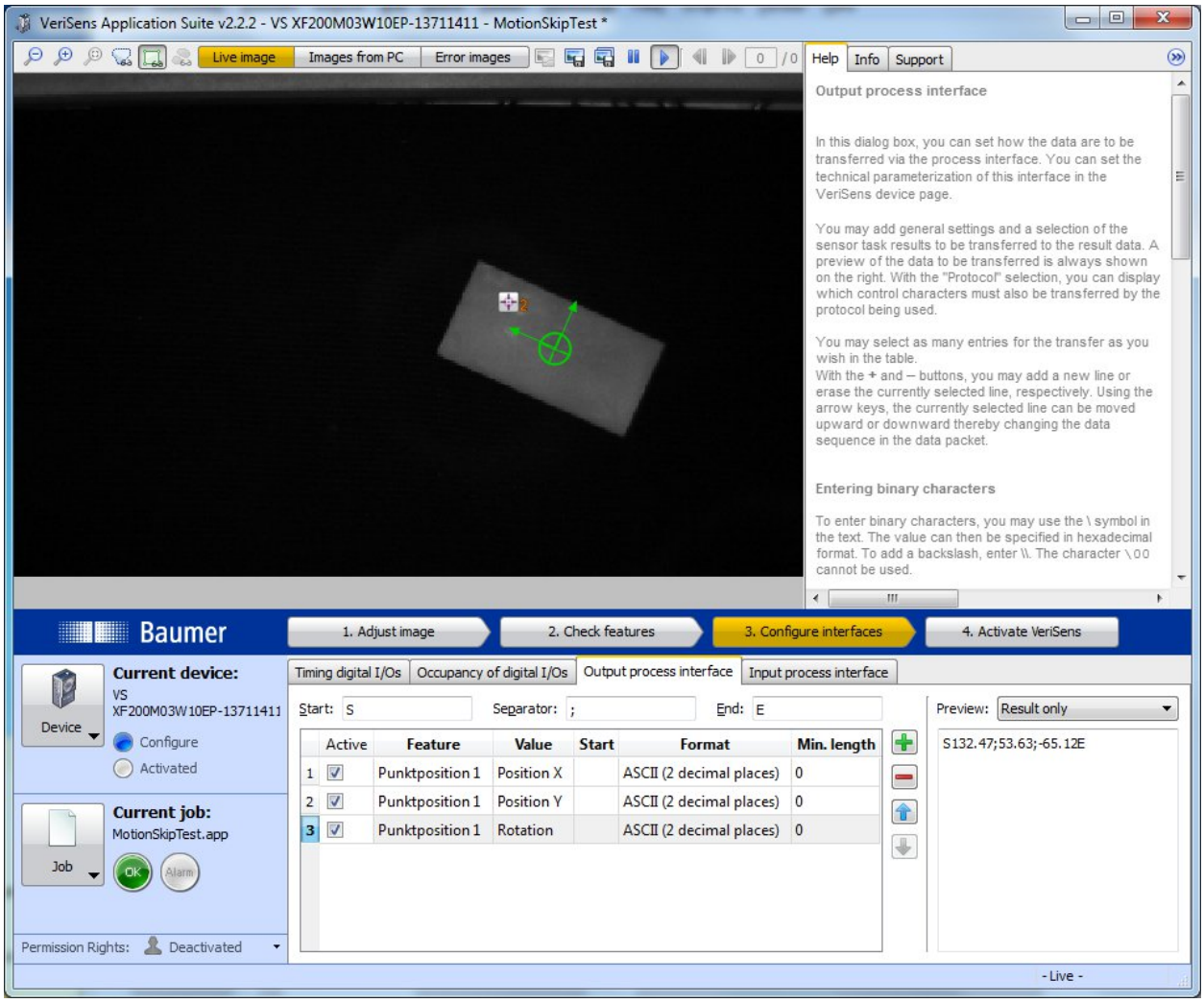


5.3. Set up the output process interface for the GetDataXYR function

To make sure the provider can read the output of the image processing results from the camera you have to create the following output interface.

The output string has to start with a "S" and end with a "E". Between two coordinates there have to be a semicolon (;) as separator.

Example for an output string: **S150.34;80.21;90.00E**


VeriSens Application Suite v2.2.2 - VS XF200M03W10EP-13711411 - MotionSkipTest *
Live image
Images from PC
Error images
Help
Info
Support
Output process interface
In this dialog box, you can set how the data are to be transferred via the process interface. You can set the technical parameterization of this interface in the VeriSens device page.
You may add general settings and a selection of the sensor task results to be transferred to the result data. A preview of the data to be transferred is always shown on the right. With the "Protocol" selection, you can display which control characters must also be transferred by the protocol being used.
You may select as many entries for the transfer as you wish in the table.
With the + and – buttons, you may add a new line or erase the currently selected line, respectively. Using the arrow keys, the currently selected line can be moved upward or downward thereby changing the data sequence in the data packet.
Entering binary characters
To enter binary characters, you may use the \ symbol in the text. The value can then be specified in hexadecimal format. To add a backslash, enter \\. The character \00 cannot be used.
Baumer
1. Adjust image
2. Check features
3. Configure interfaces
4. Activate VeriSens
Current device:
VS XF200M03W10EP-13711411
Device
Configure
Activated
Current job:
MotionSkipTest.app
Job
OK
Alarm
Permission Rights:
Deactivated
Timing digital I/Os
Occupancy of digital I/Os
Output process interface
Input process interface
Start: S
Separator: ;
End: E
Preview:
Result only
Active
Feature
Value
Start
Format
Min. length
1 Punktposition 1 Position X ASCII (2 decimal places) 0
2 Punktposition 1 Position Y ASCII (2 decimal places) 0
3 Punktposition 1 Rotation ASCII (2 decimal places) 0
S132.47;53.63;-65.12E
- Live -