COGNEX

In-Sight

GETTING STARTED



Provisur EX1039
IPR2024-00235
U.S. Patent No. 8,529,321

WEBER-0005121
DTX-0255_0001
PR01925161
US1450_003692553

Confidential

Copyright, Trademarks, Patents

The software described in this document is furnished under license, and may be used or copied only in accordance with the terms of such license and with the inclusion of the copyright notice shown on this page. Neither the software, this document, nor any copies thereof may be provided to or otherwise made available to anyone other than the licensee. Title to and ownership of this software remains with Cognex Corporation or its licensor. Cognex Corporation assumes no responsibility for the use or reliability of its software on equipment that is not supplied by Cognex Corporation. Cognex Corporation makes no warranties, either express or implied, regarding the described software, its merchantability or its fitness for any particular purpose.

The information in this document is subject to change without notice and should not be construed as a commitment by Cognex Corporation. Cognex Corporation is not responsible for any errors that may be present in either this document or the associated software.

Copyright © 1999–2000
Cognex Corporation
All Rights Reserved

This document may not be copied in whole or in part, nor transferred to any other media or language, without the written permission of Cognex Corporation.

WEBER-0005122
DTX-0255_0002
Confidential
PR01925162
US1450_003692553

The hardware and portions of the software described in this document may be covered by one or more of the following U.S. patents. Other U.S. and foreign patents are pending.

In-Sight	Patents pending
acuLight	5,861,910
Checkpoint	5,481,712 and other Vision Tools and Hardware patents
Custom Software	4,728,195; 5,673,334
Hardware	4,972,359; 5,526,050; 5,657,403; 5,793,899
Print Quality Inspection	5,859,923
Surface Mount Devices	5,367,439; 5,371,690; 5,500,906; 5,801,966; 5,805,722; 5,822,053
Vision Tools	5,495,537; 5,548,326; 5,583,954; 5,602,937; 5,640,200; 5,717,785; 5,742,037; 5,751,853; 5,768,443; 5,796,868; 5,818,443; 5,825,483; 5,825,913; 5,845,007; 5,859,466; 5,872,870; 5,909,504
Wire Bonder	5,550,763; 5,581,632; 5,640,199; 5,642,158; 5,676,302; 5,754,679; 5,757,956; 5,796,868; 5,835,622; 5,861,909; 5,901,241; 5,912,984

The following are registered trademarks of Cognex Corporation:

Cognex Cognex, Vision for Industry

The following are trademarks of Cognex Corporation:

The Cognex logo In-Sight PatFind
PatMax PatQuick

Other product and company names mentioned herein may be the trademarks of their respective owners.

WEBER-0005123
DTX-0255_0003
Confidential
PR01925163
US1450_003692553

WEBER-0005124
DTX-0255_0004
Confidential
PR01925164
US1450_003692553

TABLE OF CONTENTS

Introduction 1
For More Information 2

PART 1—USING THE IN-SIGHT INTERFACE 3

Chapter 1: Acquiring a Working Image 4
For More Information 4
Acquiring a Working Image 4

Chapter 2: Moving Around the Interface 7
What is a Spreadsheet? 7
Navigating the In-Sight Interface 7
The Control Pad 8
Lesson: Navigating Menus 9
Lesson: Navigating Dialogs 10

Chapter 3: Saving & Loading Jobs 13
What is a Job? 13
Lesson: Saving a Job 13
Lesson: Loading a Job 14

Chapter 4: Entering Formulas 15
What is a Formula? 15
Cell References 15
The Formula Editor 16
Lesson: Entering Simple Formulas 16
Lesson: Entering Formulas with Cell References 17
Lesson: Entering Simple Functions 18
Lesson: Editing Formulas 19

Chapter 5: Working With Blocks 21
What is a Block? 21
Lesson: Marking a Block 21
Lesson: Editing Blocks 22

Chapter 6: Using Property Sheets 25
What is a Property Sheet? 25
What is a Structure? 26
Lesson: Opening A Property Sheet 26
Lesson: Changing Parameter Values 26
Lesson: Using Interactive Graphics 28

WEBER-0005125
DTX-0255_0005
Confidential
PR01925165
US1450_003692553

GS.book Page vi Friday, April 14, 2000 4:04 PM

Lesson: Inserting Cell References in Values 29
Lesson: Inserting Formulas in Values 29

PART 2—VISION PROCESSING WITH IN-SIGHT 31

Chapter 7: Acquiring Images 33
How Does the Spreadsheet Update? 33
Lesson: Acquiring From Manual Triggers 33
Lesson: Acquiring from External Triggers 34

Chapter 8: Finding Features 37
Why Do I Need to Find Features? 37
Finding Connected Regions 38
Lesson: Finding Connected Regions Using ExtractBlobs 38
Lesson: Finding Connected Regions Using FindBlobs 40
Lesson: Finding Connected Regions Using SortBlobs 42
Finding Edges 44
Lesson: Finding Edges Using FindLine 44
Lesson: Finding Edges Using FindSegment 47
Lesson: Finding Edges Using FindMultiLine 48
Lesson: Finding Edges Using FindCurve 49
Lesson: Finding Edges Using FindCircle 51
Finding Patterns 53
Lesson: Finding Patterns Using FindPatterns 53

Chapter 9: Using Features to Find Objects 57
Why Combine Features? 57
Lesson: Using PointToLine 58
Lesson: Using LineToLine 60
Lesson: Using PointToPointAngle 63
Lesson: Creating a Fixture 65

Chapter 10: Making Measurements 69
What is a Measurement? 69
Lesson: Counting Pixels 69
Lesson: Counting Features 73
Lesson: Measuring Distances 73

Chapter 11: Creating Decision Points 77
What is a Decision Point? 77
Converting Measurements into Decision Points 77
Comparison 78
Logical 78
Conditionals 78

WEBER-0005126
DTX-0255_0006
Confidential
PR01925166
US1450_003692553

GS.book Page vii Friday, April 14, 2000 4:04 PM

Decision Point Formula Examples 78
Chapter 12: Communicating Results 81
Lesson: Using Discrete Output 81
Configuring Discrete Output Options 81
Writing Discrete Outputs 82
Lesson: Using Serial Output 84
Configuring Serial Output Options 84
Writing Serial Outputs 85
Chapter 13: Making a Run-time Interface 87
What is a Custom View? 87
Lesson: Understanding Graphics Functions 88
Lesson: Defining a Custom View 90
Lesson: Setting the Access Mode 92
Chapter 14: Configuring for Startup 97
Lesson: Selecting the Startup Job 97
Appendix A: The Profiler 99
Appendix B: The Serial Interface 101
Configuring the Serial Ports 102
Appendix C: The PC Server 103
Backing Up and Restoring Jobs 103
Appendix D: Remote Job Selection 107
Initiating a Job Load 107
Responding to a Job Load 108
Appendix E: Sample Images 109

WEBER-0005127
DTX-0255_0007
Confidential
PR01925167
US1450_003692553

GS.book Page viii Friday, April 14, 2000 4:04 PM

WEBER-0005128
DTX-0255_0008
Confidential
PR01925168
US1450_003692553

INTRODUCTION

Welcome to In-Sight™, a compact, stand-alone machine vision sensor for automated inspection and process control. In-Sight's innovative, built-in spreadsheet interface greatly simplifies the task of developing, operating, and maintaining vision applications.

Who Should Read This Book?

You should read *Getting Started with In-Sight* to quickly learn the most important features of In-Sight's spreadsheet-based user interface. It is written primarily for engineers and programmers who are configuring In-Sight systems, though novice users will also find it helpful.

Getting Started with In-Sight does not assume an in-depth knownledge of machine vision, but by reading it you will become familiar enough with In-Sight to start creating your own machine vision applications.

This book is not a comprehensive reference to every detail of the interface; nor is it a generalized tutorial on machine vision. Its goal is to make you comfortable enough with the In-Sight interface to explore it on your own, and to prepare you for understanding its more advanced features.

How to Use This Book

Getting Started with In-Sight consists primarily of a series of lessons designed to introduce you to the fundamentals of the In-Sight spreadsheet interface. Some lessons depend on concepts introduced in previous lessons, so it is generally best to read them in order, particularly if you are new to machine vision.

To assist you in following along with the lessons, bitmap images of the part used for this book's lessons are included on the CD-ROM that came with your In-Sight system (image1_0.bmp, image1_1.bmp, and image1_2.bmp). Refer to *Appendix C: The PC Server* for information on loading these files from a PC into the In-Sight processor. Alternatively, you can setup In-Sight's camera to acquire the images directly from their printouts in *Appendix E: Sample Images*.

NOTE The screen captures shown and the features described n this book are current as of version 1.02 of the In-Sight firmware. Visit the In-Sight Online Support Center at www.cognex.com/inSight to download updated documentation, if available.

WEBER-0005129
DTX-0255_0009
Confidential
PR01925169
US1450_003692553

GS.book Page 2 Friday, April 14, 2000 4:04 PM

For More Information

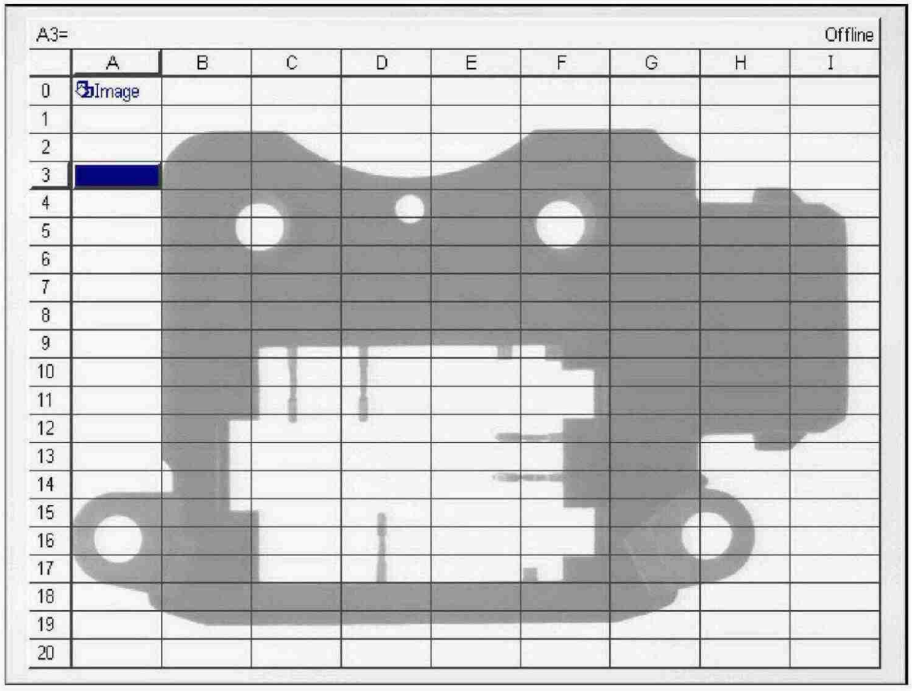
Other resources available to help you learn about the In-Sight vision sensor and its spreadsheet interface include:

- *Installing In-Sight*, Cognex P/N 590-6279 (English), 590-6279-F (French), or 590-6279-J (Japanese). Describes how to connect the vision processor, digital camera, control pad, and other components. Also contains specifications, mechanical drawings, and other hardware details.
- *In-Sight Guide and Reference*, an on-line book provided on the In-Sight CD-ROM.
- In-Sight computer-based tutorials, which are included on CD-ROM with In-Sight starter accessories kits.
- The In-Sight Online Support Center at www.cognex.com/insight.

WEBER-0005130
DTX-0255_0010
Confidential
PR01925170
US1450_003692553

Using the Interface

The chapters in this part of the book introduce the basic components of the spreadsheet interface. You will learn how to acquire an image from the In-Sight camera and how to navigate the interface using the control pad.



WEBER-0005131
DTX-0255_0011
Confidential
PR01925171
US1450_003692553

GS.book Page 4 Friday, April 14, 2000 4:04 PM

CHAPTER 1: ACQUIRING A WORKING IMAGE

In this chapter, you will see how to start In-Sight and capture a working image to use in learning the In-Sight graphic user interface.

Acquiring a Working Image

Before you can learn how to use In-Sight's interface, you will need to capture a working image. The following procedure describes how to do this.

Step 1 — Start In-Sight

To start the In-Sight system:

1. Make sure the hardware is connected properly as described in *Installing In-Sight*.
2. Switch on the VGA monitor, then apply power to the In-Sight processor.
3. Make sure the In-Sight spreadsheet overlay is visible on the VGA monitor. If not, refer to Chapter 2 of *Installing In-Sight*.

Step 2 — Adjust the Camera

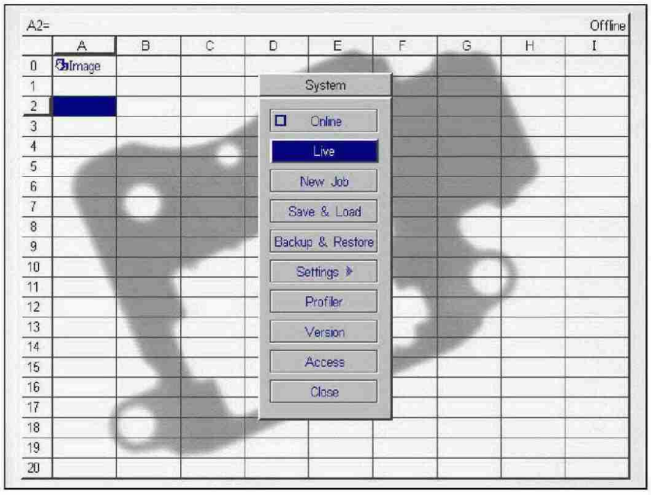
The first time you use the system, you will need to adjust the camera's focus and aperture to produce a clear image on the system monitor.

To adjust the camera:

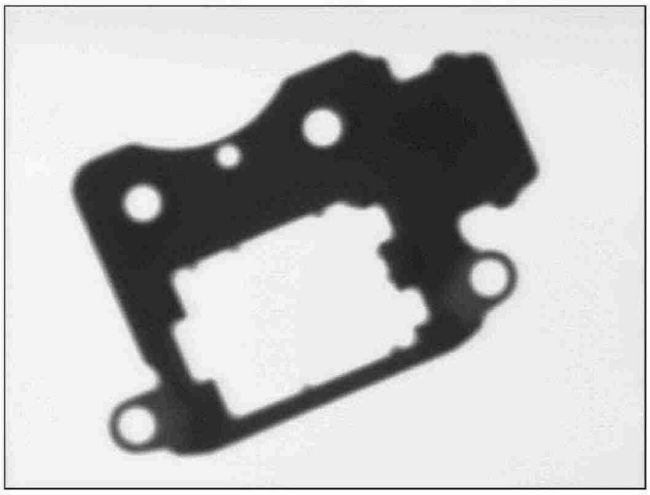
1. Press the control pad button marked △ to open the **System** menu.

WEBER-0005132
DTX-0255_0012
Confidential
PR01925172
US1450_003692553

2. Use the control pad's △ or ▽ direction buttons to highlight **Live** in the menu.



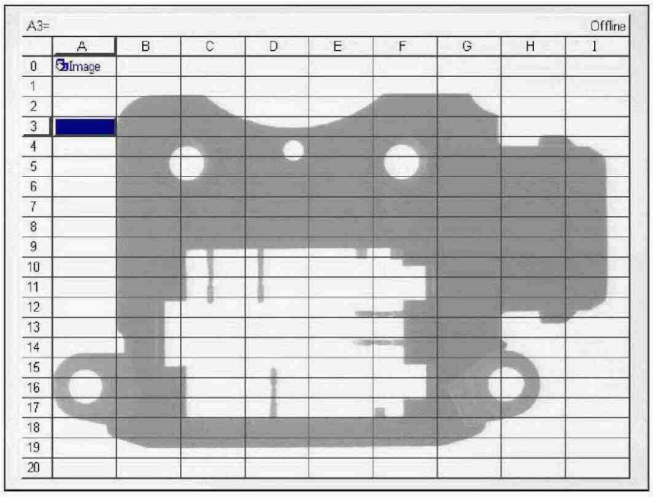
3. Press ⊗ on the control pad to put the system into live video mode, and temporarily remove the semitransparent spreadsheet overlay.



4. Adjust the camera's position, focus ring, and lens aperture until you get a clear image on the monitor.

WEBER-0005133
DTX-0255_0013
Confidential
PR01925173
US1450_003692553

5. Press any button on the control pad to return to the spreadsheet overlay, as shown below.



You now have a working image to use as you learn about the In-Sight spreadsheet environment in the following chapters.

WEBER-0005134
DTX-0255_0014
Confidential
PR01925174
US1450_003692553

CHAPTER 2: MOVING AROUND THE INTERFACE

In this chapter, you'll learn how to use the control pad to navigate the In-Sight spreadsheet, its pop-up menus, and its dialogs.

What is a Spreadsheet?

In-Sight's graphical user interface (GUI) consists of a semitransparent spreadsheet superimposed on a 640x480-pixel video image.

The spreadsheet is a table of cells arranged in up to 400 rows (0 to 399) and 26 columns (labeled A through Z). Individual cells are referred to by column letter and row number: for example, cell C4 is at the intersection of column C and row 4. In most respects, the In-Sight spreadsheet works much like other spreadsheets you may be familiar with.

You will use the spreadsheet to configure vision applications by inserting expressions called *formulas* into its cells. (Formulas will be discussed in detail in *Chapter 4: Entering Formulas*.) Since the spreadsheet is semitransparent, you will configure In-Sight with the image always in view.

You configure the spreadsheet one cell at a time. When a cell is highlighted, its formula is displayed in the ***Formula Bar*** at the top-left corner of the spreadsheet.

You use the control pad to move around the spreadsheet. If you try to move off the edge of the screen, the spreadsheet will automatically scroll to keep the selected cell in view.

Navigating the In-Sight Interface

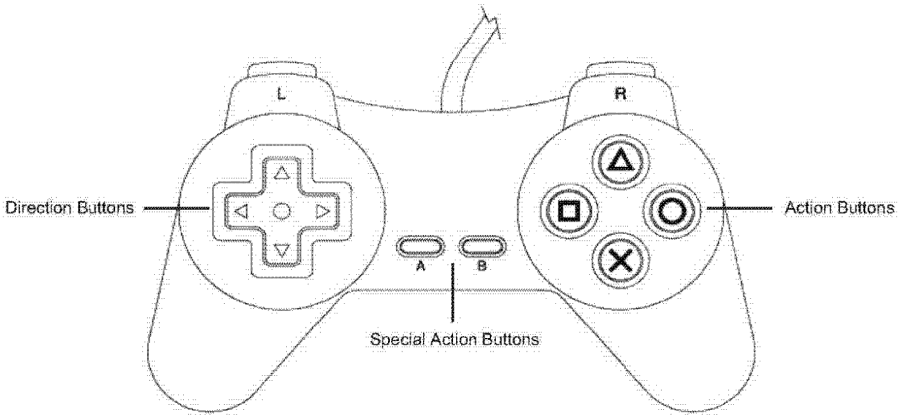
Unlike a traditional vision system that requires programming from a PC and a keyboard, In-Sight is configured using only a simple control pad.

WEBER-0005135
DTX-0255_0015
Confidential
PR01925175
US1450_003692553

GS.book Page 8 Friday, April 14, 2000 4:04 PM

The Control Pad

The In-Sight control pad has ten buttons. The four direction buttons on the left side control onscreen movement, while the four buttons on the right side are used to enter formula data and manipulate spreadsheet cells. The two center buttons perform special actions.



Direction Buttons

- △ The top direction button moves the highlight up.
- ▽ The bottom direction button moves the highlight down.
- ◁ The left direction button moves the highlight left.
- ▷ The right direction button moves the highlight right.

Action Buttons

- △ The top action button cancels changes and escapes to the previous menu or dialog. It works much like pressing the ESC key on a computer keyboard.

Tip	No matter where you are in the In-Sight user interface, you can return to the main spreadsheet by pressing △ repeatedly.

- ✕ The bottom action button selects the highlighted item, and confirms any changes you make. Its function is similar to pressing **ENTER** on a computer keyboard.
- □ The left action button accesses various editing shortcuts, including backspace in the *Formula Editor*, and zeroing numbers when in numeric edit mode. However, the most common use of the left action button is to access the image and it's graphics without displaying the spreadsheet overlay. While in this I*nteractive Graphics* mode you can zoom 5x by pressing the left action button again.
- ○ The right action button activates context-sensitive editing options for selected cells, and maximizes numbers when in numeric edit mode.

WEBER-0005136
DTX-0255_0016
Confidential
PR01925176
US1450_003692553

GS.book Page 9 Friday, April 14, 2000 4:04 PM

Special Buttons:

A The left-center button manually triggers an image acquire action and updates the spreadsheet.

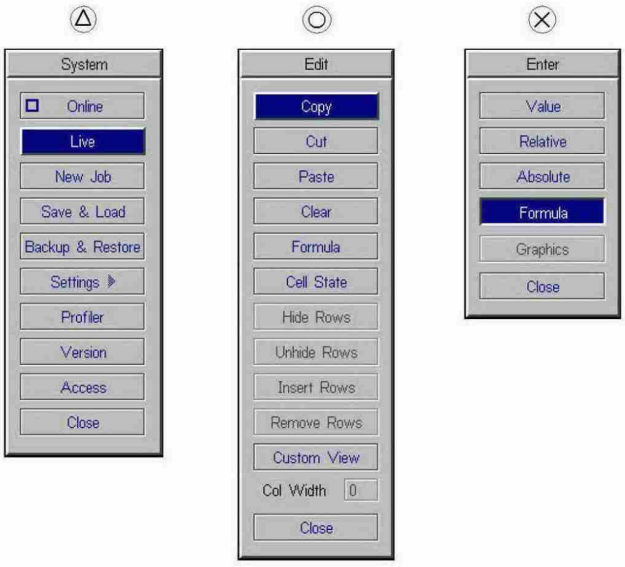
B The right-center button switches between the *Program View* and *Custom View* modes of the current spreadsheet. (These modes are discussed in *Chapter 13*.)

Lesson: Navigating Menus

The In-Sight interface has three main menus: **System**, **Edit**, and **Enter**.

- The **System** menu contains selections for system-level actions such as taking the system online and offline, saving or loading spreadsheet *Jobs*, customizing settings, and controlling access to the interface.
- The **Edit** menu contains selections for manipulating the cells in the spreadsheet.
- The **Enter** menu contains selections for entering values and formula expressions.

Note	Selections on the menus are context-sensitive. Available selections are shown in blue type. Unavailable selections are grayed out.



WEBER-0005137
DTX-0255_0017
Confidential
PR01925177
US1450_003692553

To open and navigate the menus from anywhere in the main spreadsheet:

1. Press △ for the **System** menu.
   Press ○ for the **Edit** menu.
   Press and hold ⊗ for the **Enter** menu.

Note	The **System** menu includes a submenu called **Settings**. Navigate this submenu just as you would the **System** menu.

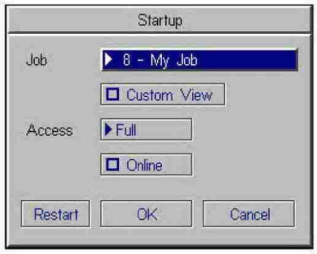
2. Press △ or ▽ to highlight a menu item.
3. Press ⊗ to select the item.
4. Press △ to close the menu, or highlight **Close** and press ⊗.

Tip	To dismiss the **Enter** menu without making a selection, highlight the **Close** menu button and then release ⊗.

Lesson: Navigating Dialogs

Several of the menus items activate *dialogs*, such as the following:





Dialogs contain controls that you use to configure the system, including:

List Box: Contains a list of selections.



Edit Box: Contains a number you can change as needed.



Check Box: Turns a feature on or off.



WEBER-0005138
DTX-0255_0018
Confidential
PR01925178
US1450_003692553

GS.book Page 11 Friday, April 14, 2000 4:04 PM

To navigate dialogs and configure dialog controls:

1. Highlight the control using the direction buttons.
2. Select the control by pressing ⊗.
3. Configure the control:

   Edit Boxes

   Press ◁ or ▷ to highlight a digit.

   Press △ to increase the highlighted digit.

   Press ▽ to decrease the highlighted digit.

   Press □ to enter a zero.

   Press ○ to increase the number to its maximum allowable value.

   Check Boxes

   Press ⊗ to check or uncheck the box.

   List Boxes

   Press △ or ▽ to highlight an item.

   Press ⊗ to select the highlighted item.
4. To confirm your changes, highlight the **OK** button and press ⊗.
5. To cancel any changes, highlight the **Cancel** button and press ⊗.

Tip	Press △ to cancel the changes you've made to a dialog.

WEBER-0005139
DTX-0255_0019
Confidential
PR01925179
US1450_003692553

GS.book Page 12 Friday, April 14, 2000 4:04 PM

WEBER-0005140
DTX-0255_0020
Confidential
PR01925180
US1450_003692553

CHAPTER 3: SAVING & LOADING JOBS

In this chapter, you will learn how to save and load In-Sight Jobs.

What is a Job?

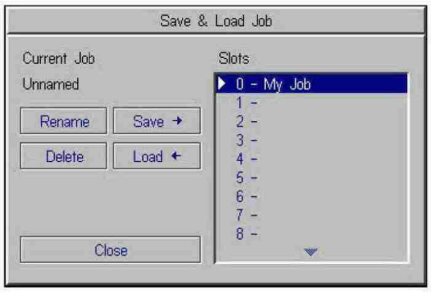
An In-Sight *Job* is a spreadsheet containing a collection of formulas and text. Once you have created a Job, you can store the Job in In-Sight's non-volatile memory. In-Sight can store up to 20 Jobs, but only one Job is active at a time.

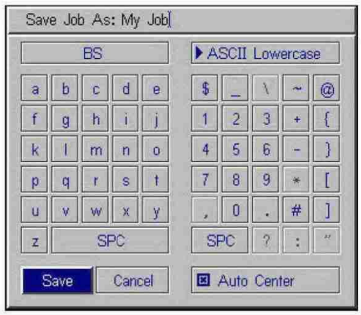
Note	Everything you enter in the spreadsheet remains in memory as long as the processor has power. However, if the processor loses power, you will lose all unsaved data. Therefore, make sure you save your Jobs.

Lesson: Saving a Job

To save a Job:

1. From the spreadsheet, press △ to open the **System** menu.
2. Highlight the **Save & Load** selection, then press ⊗ to open the **Save & Load Job** dialog.
3. Highlight one of the memory **Slots** and press ⊗. An arrow appears to the left of the Slot number you selected.





4. Highlight the **Save** button and press ⊗ to open the **Save Job** dialog.
5. Use the onscreen Keypad to enter a name for the Job.

Tip	If you make a mistake entering your Job name, press △ to backspace over the last character entered. If you want to start over, press △ to cancel the save.

6. Highlight the **Save** button and press ⊗ to confirm your changes.

WEBER-0005141
DTX-0255_0021
Confidential
PR01925181
US1450_003692553

GS.book Page 14 Friday, April 14, 2000 4:04 PM

Lesson: Loading a Job

Loading a Job retrieves it from In-Sight's non-volatile memory. The steps for loading a job are similar to the steps for saving a job:

1. From the main spreadsheet, press △ to open the **System** menu.
2. Highlight the **Save & Load** selection, then press ⊗ to open the **Save & Load Job** dialog.
3. Highlight a Job in the list of Slots and press ⊗.
4. Highlight the **Load** button and press ⊗.

Tip	You can move a Job from one memory slot to another by loading the Job, saving it to another slot, and then deleting the original.

WEBER-0005142
DTX-0255_0022
Confidential
PR01925182
US1450_003692553

CHAPTER 4: ENTERING FORMULAS

In this chapter, you will learn how to enter Formulas into In-Sight spreadsheet cells.

What is a Formula?

A formula is a mathematical expression consisting of values and references to values, connected by mathematical operators, and resulting in a new value or set of values. In-Sight considers everything you enter into a cell—whether a single constant or a complex vision-processing function—to be a formula.

Formulas can return three types of results:

Values Numbers, such as 1.00 or –1.00.

Strings Alphanumeric characters, such as "PASS" or "FAIL".

Structures A vision-processing function. (Structures are discussed in *Chapter 6: Using Property Sheets*.)

Cell References

Formulas can refer to other cells in the spreadsheet by using *absolute* and *relative* cell references.

Relative cell references point to cells relative to the position of the formula. When copied or moved to a new location in the spreadsheet, these references automatically adjust to maintain the same position relative to the formula, as in the following examples:

A5=A3+A4 If this formula is copied to cell B5, it becomes B5=B3+B4.

A5=A3+A4 If this formula is copied to cell A6, it becomes A6=A4+A5.

Absolute cell references point to *fixed* locations in the spreadsheet, regardless of the formula's location. They do not change when the formula is copied or moved. Absolute cell references are designated by a dollar sign ($) preceding the cell's row or column label. By placing the dollar sign appropriately, you can refer to a fixed row, column, or cell, as in the following examples:

A5=$A$3+$A$4 If this formula is copied to cell B5, it becomes B5=$A$3+$A$4.

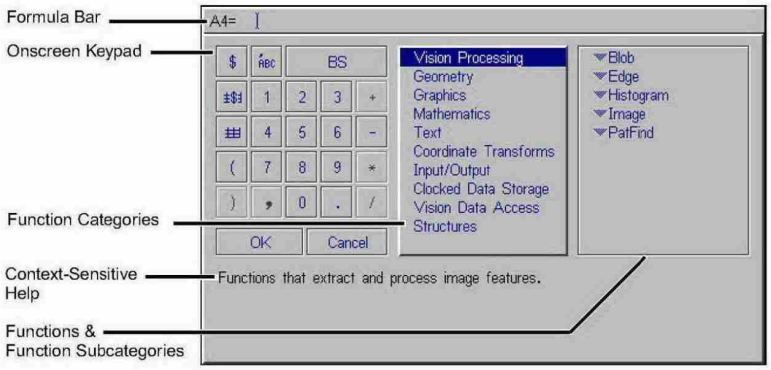
A5=A$3+A$4 If this formula is copied to cell B5, it becomes B5=B$3+B$4.

WEBER-0005143
DTX-0255_0023
Confidential
PR01925183
US1450_003692553

GS.book Page 16 Friday, April 14, 2000 4:04 PM

The Formula Editor

In-Sight includes a Formula Editor that you will use to enter formulas into the spreadsheet.



The Formula Editor has two cell reference buttons on the left side of its onscreen keypad:

- The button with the spreadsheet icon selects relative cell references.
- The button with the spreadsheet icon including a dollar sign selects absolute cell references.

Lesson: Entering Simple Formulas

This procedure describes the general procedure for entering simple formulas that do not contain cell references.

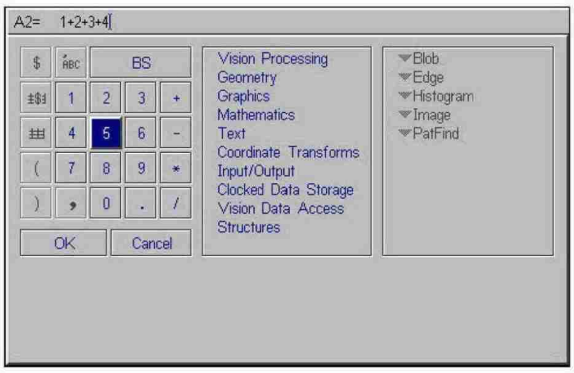
To enter a simple formula into the spreadsheet:

1. Use the direction buttons to highlight the cell you want to put the formula in.
2. Press ⊗ to open the Formula Editor.
3. Use the direction buttons to highlight the number you want to enter in the onscreen keypad.
4. Press ⊗ to insert the highlighted number into the formula.

WEBER-0005144
DTX-0255_0024
Confidential
PR01925184
US1450_003692553

GS.book Page 17 Friday, April 14, 2000 4:04 PM

5. Repeat steps 4 and 5 as needed to insert additional numbers and basic arithmetic operators (+, -, /, *) until the formula is complete.



6. Highlight **OK** and press ⊗ to insert the formula into the cell.
7. Repeat steps 1–6 to insert additional formulas into other cells in the spreadsheet.

Lesson: Entering Formulas with Cell References

This lesson shows how cell references are included in formulas.

To enter a formula that includes cell references:

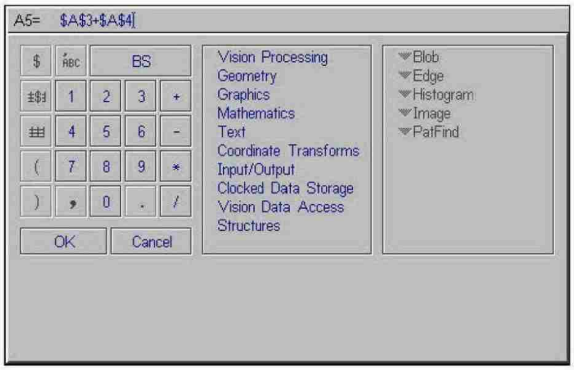
1. Use the direction buttons to highlight the cell you want to put the formula in.
2. Press and hold ⊗ to open the **Enter** menu.
3. Highlight **Formula** and release ⊗ to open the Formula Editor.
4. Highlight the **Absolute Cell Reference** button or the **Relative Cell Reference** button in the onscreen keypad, and press ⊗ to enter *Cell Selection* mode.

Note	Cell Selection mode temporarily changes the display back to the spreadsheet, where you highlight and select the cells to use as references in the formula you are editing.

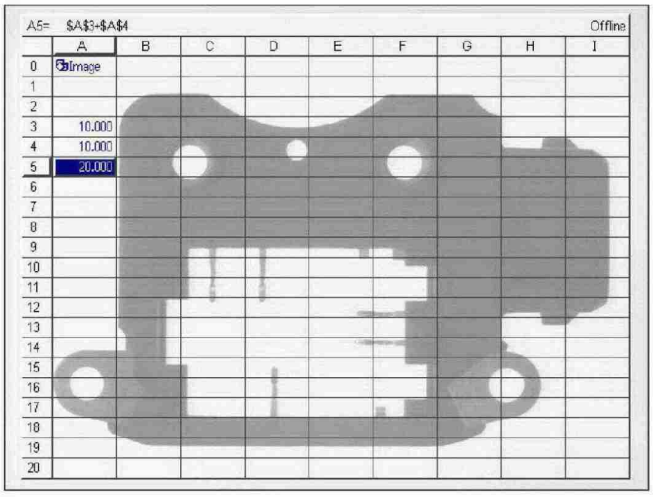
5. Highlight a cell containing a formula and press ⊗ to insert the cell reference into the formula.
6. Highlight a basic math operator in the onscreen keypad and press ⊗ to insert it into the formula.

WEBER-0005145
DTX-0255_0025
Confidential
PR01925185
US1450_003692553

7. Repeat step 3, but highlight and select a different cell reference to insert in the formula.



8. Highlight the **OK** button in the Formula Editor and press ⊗ to enter the formula.



Lesson: Entering Simple Functions

In-Sight includes many standard math, statistic, and trigonometric functions. This lesson describes the general procedure for entering these functions into a formula.

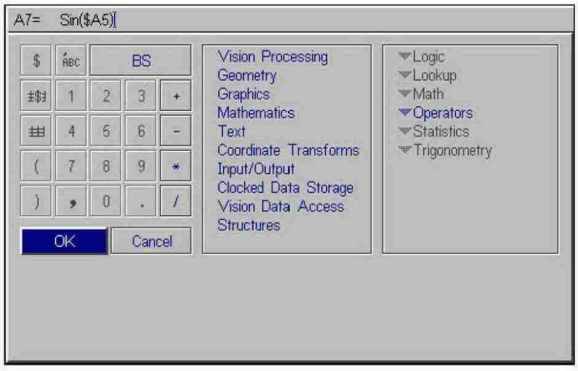
To enter a simple function:

1. Highlight an empty cell in the spreadsheet and press ⊗ to open the Formula Editor.

WEBER-0005146
DTX-0255_0026
Confidential
PR01925186
US1450_003692553

GS.book Page 19 Friday, April 14, 2000 4:04 PM

2. Highlight the **Math** function category, and press ▷ to change the focus to the function sub-categories on the right side of the dialog.
3. Highlight the **Trigonometry** subcategory, then press ⊗ to expand the function list.
4. Highlight the function you want and press ⊗ to insert it in the Formula Bar.
5. Highlight the **Absolute Cell Reference** button or **Relative Cell Reference** button, and press ⊗ to enter Cell Selection mode.
6. Highlight a cell containing data, and press ⊗ to insert the cell reference into the formula.



7. Highlight the **)** button in the Formula Editor to complete the formula.
8. Highlight the **OK** button in the Formula Editor and press ⊗ to enter the formula in the spreadsheet.

Lesson: Editing Formulas

After you've entered a formula into a spreadsheet cell, you can return to it and edit it as needed.

To edit a formula string:

1. Highlight the cell containing the formula you want to edit, and press ⊗ to open the Formula Editor.
2. Press △ until the highlight moves to the Formula Bar. A text cursor shows the editing position.
3. Press ◁ and ▷ to move the cursor through the formula string.

Tip	Press □ to backspace over characters in the formula.

4. When you finish editing, press ▽ to move back through the onscreen keypad, highlight **OK,** and press ⊗.

WEBER-0005147
DTX-0255_0027
Confidential
PR01925187
US1450_003692553

GS.book Page 20 Friday, April 14, 2000 4:04 PM

WEBER-0005148
DTX-0255_0028
Confidential
PR01925188
US1450_003692553

CHAPTER 5: WORKING WITH BLOCKS

In this chapter, you will learn how to manipulate blocks of cells in a spreadsheet.

What is a Block?

A *Block* is a contiguous, rectangular group of cells in a spreadsheet. You can cut, copy, paste, clear, hide, insert, and remove blocks using functions in the Edit menu.

Lesson: Marking a Block

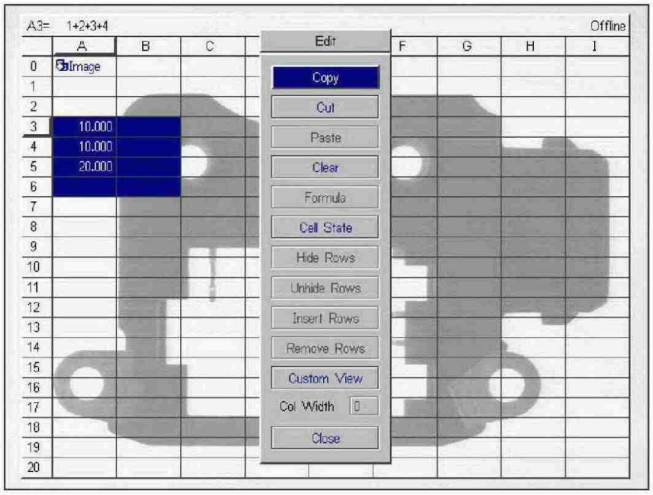
Before you can copy, cut, hide, or perform other operations on a block of cells, you need to mark the block.

To mark a block of cells:

1. Highlight the upper-left cell of the block.
2. Press and hold ◎.
3. Use the direction buttons to select the lower-right cell in the block.

Note You can mark an entire row of the spreadsheet by pressing ◁ while holding down ◎. Likewise, press △ while holding down ◎ to mark an entire column.

4. Release ◎ to open the **Edit** menu.



WEBER-0005149
DTX-0255_0029
Confidential
PR01925189
US1450_003692553

Lesson: Editing Blocks

This lesson describes each of the block editing functions accessible from the Edit menu.

Note The procedures in this section can be applied to single cells as well as to blocks of cells.

Copy

Copying a block of cells stores its formulas in the system's internal clipboard, and leaves the original formulas in place in the spreadsheet.

To copy a block of cells:

1. Mark the block you want to copy, as described in the previous lesson.
2. Highlight **Cut** on the **Edit** menu and press ⊗.

Cut

Cutting a block of cells stores its formulas in the system's internal clipboard, and removes the formulas from the spreadsheet.

To cut a block of cells:

1. Mark the block you want to cut, as described in the previous lesson.
2. Highlight **Cut** on the **Edit** menu and press ⊗.

Paste

Pasting copies the formulas contained in the system's internal clipboard back to the spreadsheet.

To paste a block of cells:

1. Highlight the upper-left cell of the block's new location.
2. Press ◎ to open the **Edit** menu.
3. Highlight **Paste** in the **Edit** menu and press ⊗.

Clear

Clearing cells permanently deletes their contents. Cleared cells cannot be pasted back into the spreadsheet.

To clear a block of cells:

1. Mark the block of cells you want to clear, as described above.
2. Highlight **Clear** in the **Edit** menu and press ⊗.

Hide/Unhide Rows & Columns

Hiding rows and columns in the spreadsheet removes them from view. Their data remains in the spreadsheet, however, and becomes visible again when you unhide them.

To select rows or columns of cells:

WEBER-0005150
DTX-0255_0030
Confidential
PR01925190
US1450_003692553

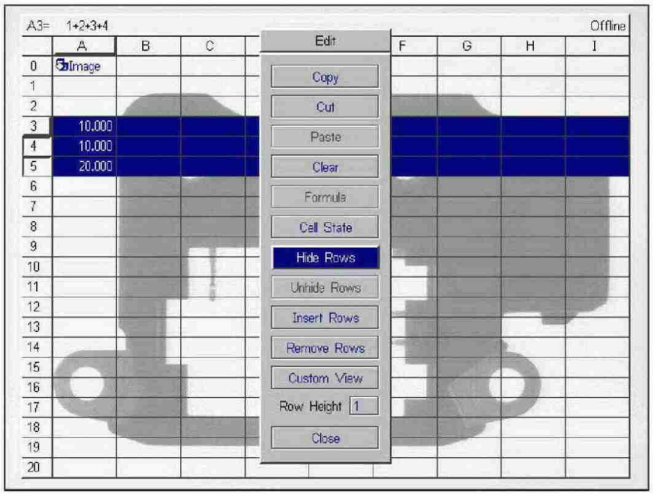
1. Move the highlight to the *uppermost row* or *leftmost column* you want to hide.
2. Press and hold ◎, and select the rows or columns you want to hide:
   - Press ◁ to select the row, and ▽ to select additional rows.
   - Press △ to select the column, and ▷ to select additional rows.

Note	You cannot select rows and columns at the same time.

3. Release ◎ to open the Edit menu.

To hide rows or columns of cells:

1. Select the rows or columns you want to hide, as described above.
2. Highlight **Hide Rows** or **Hide Cols** in the **Edit** menu and press ⊗.



To unhide rows and columns:

1. Mark as described above the rows or columns that fall on both sides of the hidden rows or columns.
2. Highlight **Unhide Rows** or **Unhide Cols** in the **Edit** menu and press ⊗.

Note	You can find hidden rows and columns by looking for gaps in the row and column labels. For example, if column A appears next to column D, then columns B and C are hidden.

WEBER-0005151
DTX-0255_0031
Confidential
PR01925191
US1450_003692553

GS.book Page 24 Friday, April 14, 2000 4:04 PM

Inserting and Removing Rows

You can also insert rows into or delete rows from the spreadsheet.

To insert rows:

1. Mark a block that includes the number of rows that you want to insert. The new rows will be inserted immediately above the block you mark.
2. Highlight **Insert Rows** in the **Edit** menu and press ⊗.

To remove rows:

1. Mark a block that includes the rows you want to delete.
2. Highlight **Remove Rows** in the **Edit** menu and press ⊗.

Note In-Sight cannot insert or delete columns.

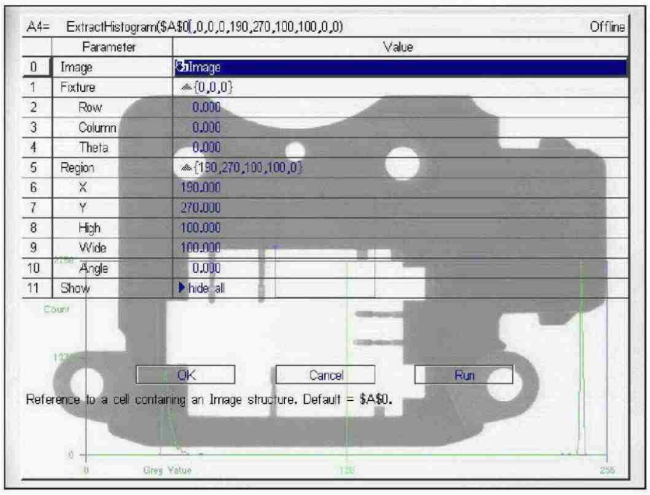
WEBER-0005152
DTX-0255_0032
Confidential
PR01925192
US1450_003692553

CHAPTER 6: USING PROPERTY SHEETS

In this chapter, you will learn how to use functions by configuring arguments in Property Sheets.

What is a Property Sheet?

A *Property Sheet* is a special In-Sight dialog containing a table that allows you to easily identify and configure *parameters* for a particular function. Whenever a function returns a *Structure* (see below) or has a large number of input parameters, the In-Sight user interface provides a Property Sheet to simplify the task of configuring the function. Shown below is an example of a typical Property Sheet.



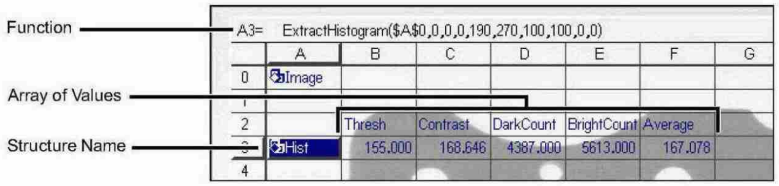
Note	This Property Sheet contains a Vision Processing function called ExtractHistogram, which is discussed in *Chapter 10: Making Measurements*. For now, ExtractHistogram provides a convenient example of a Property Sheet.

When activated, the Property Sheet temporarily replaces the main spreadsheet with a table that lists the default parameter values. Each row of the Property Sheet contains a single value, or expands into a group of values.

WEBER-0005153
DTX-0255_0033
Confidential
PR01925193
US1450_003692553

What is a Structure?

In-Sight functions that return Structures differ from conventional spreadsheet functions. Instead of returning a single value or text string to a cell, most functions that return Structures also emit an array of additional functions into adjacent cells.



Note	Structure functions cannot be combined in a single formula with conventional values and operators. When a Structure function is assigned to a cell, it must be the only entity in that cell. The In-Sight Formula Editor enforces this restriction by activating and deactivating selections as necessary.

Lesson: Opening A Property Sheet

To see an example of a Property Sheet:

1. Highlight an empty cell in the spreadsheet and press ⊗ to open the Formula Editor.
2. Highlight the **Vision Processing** category, then press ▷ to change the focus to the function subcategories.
3. Press △ / ▽ to highlight the **Histogram** subcategory, then press ⊗ to expand the list of functions.
4. Highlight **ExtractHistogram** and press ⊗ to open the Property Sheet.
5. Highlight **OK** and press ⊗. A Structure called **Hist** then appears in the spreadsheet.

Keep the spreadsheet open for use in the following lessons.

Lesson: Changing Parameter Values

Property Sheets are initially configured with the default parameter values for the selected function. You can edit these values as necessary to configure the Property Sheet for your application.

To change the values in a Property Sheet:

1. Highlight the cell in the spreadsheet containing the **Hist** Structure created in Lesson 5.1, then press ⊗ to open its Property Sheet.
2. Press △ / ▽ to highlight the value you want to change.

WEBER-0005154
DTX-0255_0034
Confidential
PR01925194
US1450_003692553

GS.book Page 27 Friday, April 14, 2000 4:04 PM

3. Press and hold ⊗ to open the **Enter** menu, which includes the following items:

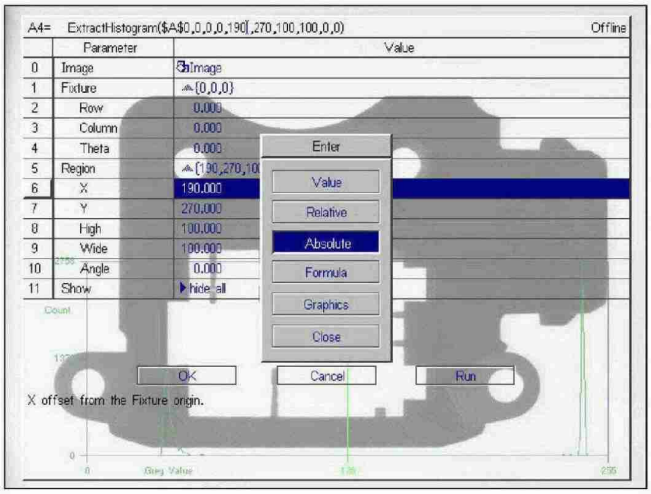
   Value: Inserts a constant into an edit box, check box, list box, or *Interactive Graphic*.

   Relative: Switches to Cell Selection mode for selecting a cell from the spreadsheet to use as a relative reference.

   Absolute: Switches to Cell Selection mode for selecting a cell from the spreadsheet to use as an absolute reference.

   Formula: Activates the Formula Editor.

   Graphics: Switches to Interactive Graphics mode for changing the size and position of the interactive graphic, if available.



1. Highlight the item you want to use, and release ⊗ to select it.
2. Change the selected value as needed, and press ⊗ to confirm.
3. Highlight **OK** and press ⊗ to confirm your changes to the Property Sheet, or highlight **Cancel** and press ⊗ to leave the Property Sheet unchanged.

Note	Pressing ◁ moves the Property Sheet focus from the current highlighted **Value** to the **OK** button below the table. Pressing ▷ moves the focus from the highlighted **Value** to the **Run** button below the table. You can quickly return to the previously highlighted row by pressing △.

WEBER-0005155
DTX-0255_0035
Confidential
PR01925195
US1450_003692553

Lesson: Using Interactive Graphics

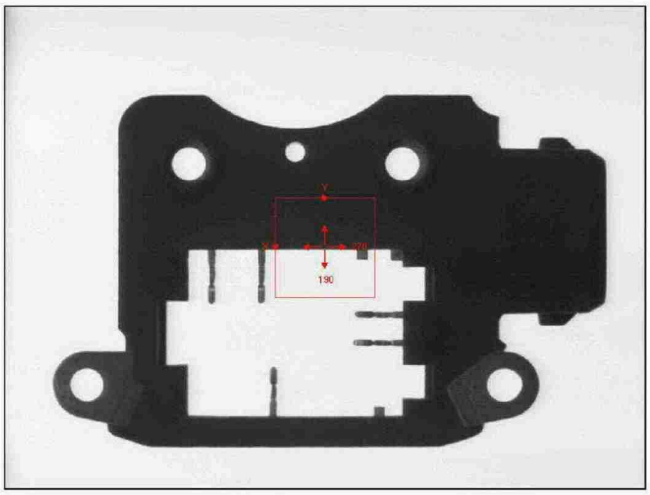
Some Property Sheet values are contained in an expandable Parameter group. These Parameter groups are common in Structure functions that include a graphical representation, such as an image *Region*.

Note	The down arrow (▼) to the left of a Value indicates an expandable Parameter group. To collapse a group of expanded values, highlight the value with an up arrow (▲) on its left and press ⊗.

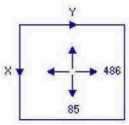
A Region consists of a set of image coordinates that define a contiguous section of the image for processing. Since adjusting Region coordinates one value at a time would be tedious, Property Sheets allow you to adjust most image-related Parameter groups as a single *Interactive Graphic*.

To change an Interactive Graphic:

1. Highlight the cell in the spreadsheet containing the **Hist** Structure created in Lesson 5.1.
2. Press ⊗ to open the **ExtractHistogram** Property Sheet.
3. Highlight the value group labeled **Region**, and press ⊗ to expand the list of values.
4. Highlight **X**, and press ⊡ to enter Interactive Graphics mode.



5. Press ◎ to cycle through each of the available Interactive Graphics modes:

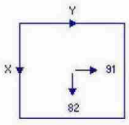


Move

- Use the direction buttons to move the Region on the image.
- Move is the default mode for Interactive Graphics.

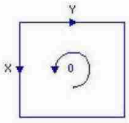
WEBER-0005156
DTX-0255_0036
Confidential
PR01925196
US1450_003692553

GS.book Page 29 Friday, April 14, 2000 4:04 PM



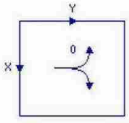
Resize

- Press △ / ▽ to enlarge/reduce the Region's height.
- Press ▷ / ◁ to enlarge/reduce the Region's width.
- Resize always enlarges or reduces from the center of the Region.



Rotate

- Press ▷ / ◁ to rotate the Region clockwise/counterclockwise.



Curve

- Press △ / ▽ to bend the Region into a curve facing up/down.
- Curve is available only on selected functions.

6. Press ⊗ to confirm your changes and return to the Property Sheet, or press △ to cancel changes.

Lesson: Inserting Cell References in Values

Function parameters can include references to other cells in the spreadsheet.

To insert a cell reference as a function parameter:

1. Highlight the row in the Property Sheet into which you want to insert a cell reference.
2. Press and hold ⊗ to open the **Enter** menu.
3. Highlight **Relative** or **Absolute**, then release ⊗ to enter Cell Selection mode.
4. Highlight the cell you want to refer to, then press ⊗ to confirm or △ to cancel.

Tip	You can replace multiple function arguments with cell references by marking a block of cells in the spreadsheet while in Cell Selection mode.

Lesson: Inserting Formulas in Values

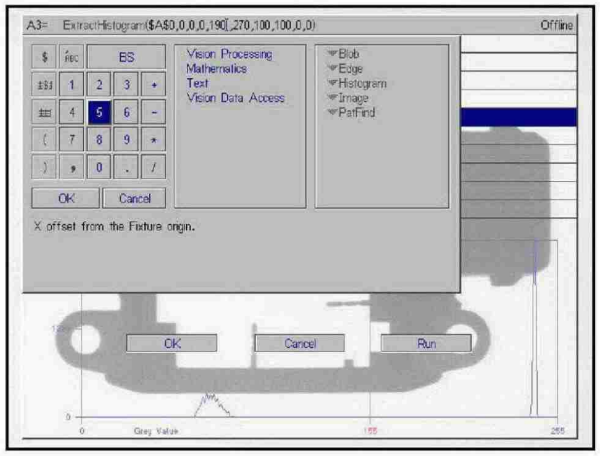
Spreadsheet formulas that compute a single value can also be used as parameter values in a Property Sheet.

To use the Formula Editor to insert a formula:

1. Highlight the row in the Property Sheet into which you want to insert a formula.
2. Press and hold ⊗ to open the **Enter** menu.

WEBER-0005157
DTX-0255_0037
Confidential
PR01925197
US1450_003692553

3. Highlight **Formula** and release ⊗ to open the Formula Editor.



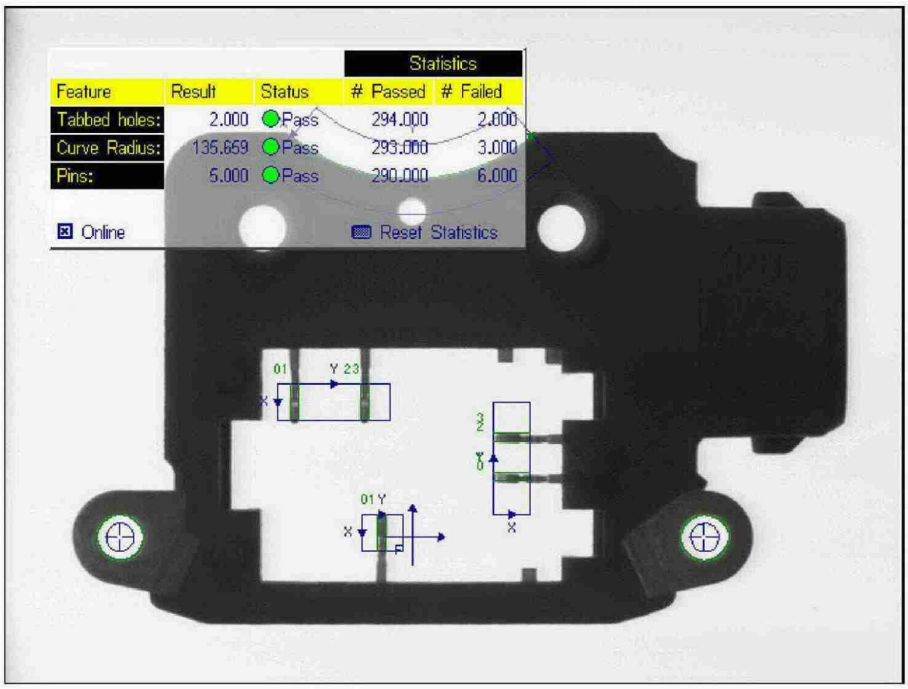
Tip	You can also activate the Formula Editor by pressing ○ to open the **Edit** menu, then selecting **Formula**. This method is useful for overriding Property Sheet check boxes, list boxes, and other embedded controls.

1. Use the Formula Editor to insert your formula directly into the parameter value.
2. Highlight **OK** and press ⊗ confirm the change, or highlight **Cancel** and press ⊗ to cancel.

WEBER-0005158
DTX-0255_0038
Confidential
PR01925198
US1450_003692553

Vision Processing

In the previous chapters you learned how to navigate the In-Sight interface using the control pad. In the following chapters, you will learn the basic techniques for configuring a vision application in the spreadsheet.



WEBER-0005159
DTX-0255_0039
Confidential
PR01925199
US1450_003692553

GS.book Page 32 Friday, April 14, 2000 4:04 PM

WEBER-0005160
DTX-0255_0040
Confidential
PR01925200
US1450_003692553

GS.book Page 33 Friday, April 14, 2000 4:04 PM

CHAPTER 7: ACQUIRING IMAGES

In this chapter, you'll learn how to acquire images, set image acquisition parameters, and trigger processing.

How Does the Spreadsheet Update?

Cell dependencies drive spreadsheet updates. For example, in the formula **A1 = B1 + C1**, cell **A1** is said to be **dependent** on cells **B1** and **C1**. In this case, **A1** automatically updates whenever **B1** or **C1** changes.

In-Sight dedicates cell **A0** to the **AcquireImage** function**.** When **AcquireImage** receives a trigger from any source, it takes a digital image from the camera and assigns it to cell **A0**. And because the results of a visual inspection are always dependent on the input image in **A0**, the spreadsheet updates, or *runs*, whenever the input image changes.

Lesson: Acquiring From Manual Triggers

Manually triggered image acquisition is useful during development and testing. In-Sight has two manual modes for acquiring images: *Live* and *Single* acquire.

Live Mode

Live video mode acquires and displays images in rapid succession while the spreadsheet overlay is temporarily disabled. This is useful when making camera adjustments.

To initiate Live video mode:

1. Press △ to open the **System** menu.
2. Highlight **Live** and press ⊗ to enter Live mode.
3. Press △ to terminate Live mode.

Note	Any spreadsheet function that depends on the acquired image updates automatically when Live mode terminates.

Single Acquire Mode

To manually acquire one image at a time:

- Press the control pad button **A**. A new image then appears in the display, and any spreadsheet function that depends on the acquired image automatically updates.

WEBER-0005161
DTX-0255_0041
Confidential
PR01925201
US1450_003692553

GS.book Page 34 Friday, April 14, 2000 4:04 PM

- Press and hold **A** to simulate multiple single acquisition triggers occurring in succession. The spreadsheet updates with each acquired image until you press **A** a second time.

Note	Disabling the Manual trigger checkbox in the AcquireImage Property sheet also disables the single acquisition trigger from the control pad.

Lesson: Acquiring from External Triggers

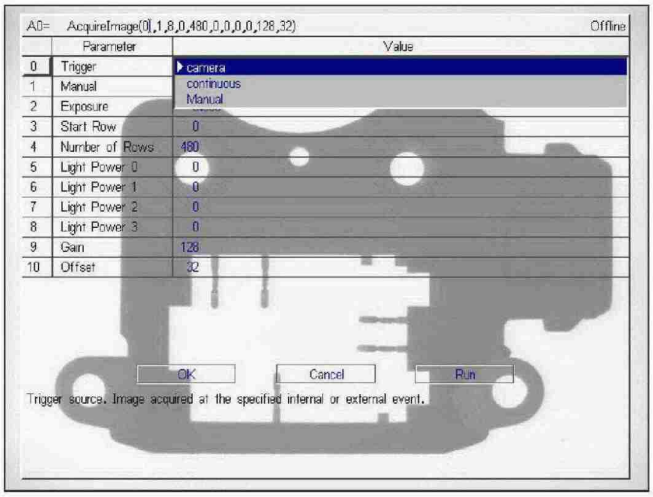
In order for In-Sight to run a machine vision application automatically, the system must have automatically triggered image acquisitions. This trigger is typically supplied by an external signal.

To select an acquisition trigger source:

1. Highlight cell **A0** in the spreadsheet.
2. Press ⊗ to open the **AcquireImage** Property Sheet.
3. Highlight the row labeled **Trigger**, and press ⊗ to open the drop-down list of trigger sources.

   Camera Acquires an image upon receiving a pulse to the trigger input located directly on the camera.

   Continuous Acquires images as fast as the camera will allow.



Note	All external trigger sources are disabled when the system is **Offline**. To go **Online**, return to the spreadsheet, press △ to open the **System** menu, and check the **Online** option.

WEBER-0005162
DTX-0255_0042
Confidential
PR01925202
US1450_003692553

GS.book Page 35 Friday, April 14, 2000 4:04 PM

Acquisition Parameters

Other parameter values in the AcquireImage function also control important image acquisition parameters. These parameters include:

Exposure Image exposure time can be set in milliseconds. A smaller number reduces the blurring of an object in motion. A larger numbers allows more light on the imaging sensor.

Light Power Light power parameters can be set to control the current on up to four independent LED light sources.

Gain/Offset Analog gain and offset parameters can be set to adjust the voltage of the analog video signal before it is converted a digital signal.

To adjust any of these parameters go to cell A0 and activate the AcquireImage Property Sheet. Changes will take effect when you confirm changes to the Property Sheet.

WEBER-0005163
DTX-0255_0043
Confidential
PR01925203
US1450_003692553

GS.book Page 36 Friday, April 14, 2000 4:04 PM

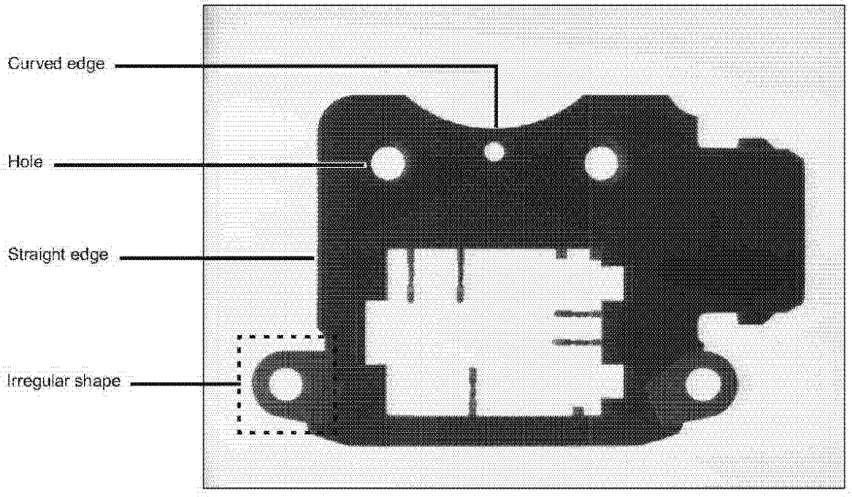
WEBER-0005164
DTX-0255_0044
Confidential
PR01925204
US1450_003692553

CHAPTER 8: FINDING FEATURES

In this chapter, you will learn several techniques for locating individual object features in the field of view.

Why Do I Need to Find Features?

An object can be represented by the location of its coordinates in the image: row, column, and theta (angle of rotation). But objects are also composed of a variety of individual features. For example, an object can include simple features such as holes, curved edges, and straight edges, or more complex features such as irregular shapes or grayscale patterns.



While In-Sight can find and report the position of an object in its entirety, you will generally want to identify and find the object's most important features first. In fact, finding individual features is sometimes easier than finding the object itself.

This approach not only allows you to extract data about individual features, but also enables you to efficiently combine those features to locate the object or to evaluate its quality relative to a known specification.

This chapter discusses In-Sight's primary tools for finding these types of features:

- **Connected regions (Blobs)**
- **Edges**
- **Patterns**

WEBER-0005165
DTX-0255_0045
Confidential
PR01925205
US1450_003692553

GS.book Page 38 Friday, April 14, 2000 4:04 PM

Subsequent chapters describe basic techniques for combining features to find objects, make measurements, and create decision points based on those measurements.

Finding Connected Regions

Many object features can be distinguished from their backgrounds according to how bright or dark they appear in the image. These bright or dark features, commonly known as "Blobs", can be described by a set of statistical characteristics.

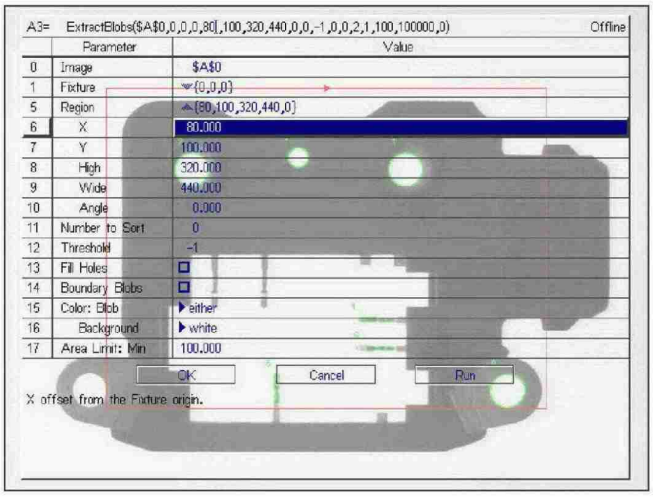
In-Sight provides several functions for extracting and identifying Blob features in an image:

ExtractBlobs	Extracts Blobs from an image.
FindBlobs	Compares extracted Blobs to a reference Blob.
SortBlobs	Sorts extracted Blobs by coordinates.

Lesson: Finding Connected Regions Using ExtractBlobs

To insert an ExtractBlobs function into the spreadsheet:

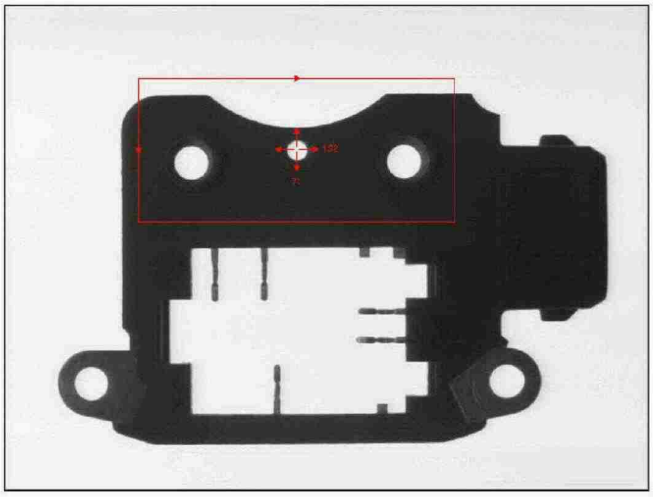
1. In column **A**, select a cell in an empty row that also has an empty row immediately above it.
2. Press ⊗ to open the Formula Editor.
3. Highlight **ExtractBlobs** in the **Blob** subcategory under **Vision Processing**, then press ⊗ to open the ExtractBlobs Property Sheet.



4. Highlight the **Region** row, then press ⊗ to expand the values.

WEBER-0005166
DTX-0255_0046
Confidential
PR01925206
US1450_003692553

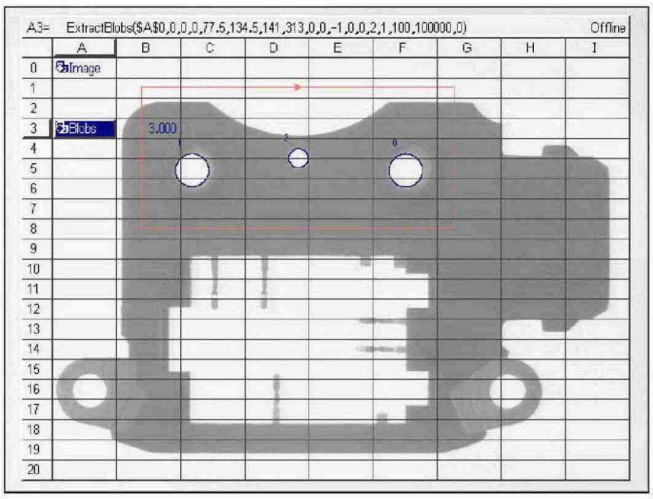
5. Highlight the **X** value and press ⓞ to switch to Interactive Graphics mode.



6. Position the Region so that it completely encloses the Blobs you want to locate, then press ⓧ to confirm changes.
7. Complete the remainder of the Property Sheet.

Note	You can press the **Run** button to see the highlighted boundaries of the located Blobs.

8. Highlight **OK** and press ⓧ to confirm.



WEBER-0005167
DTX-0255_0047
Confidential
PR01925207
US1450_003692553

The ExtractBlobs function inserts a Blobs Structure in the spreadsheet, along with the Vision Data Access function GetNFound:

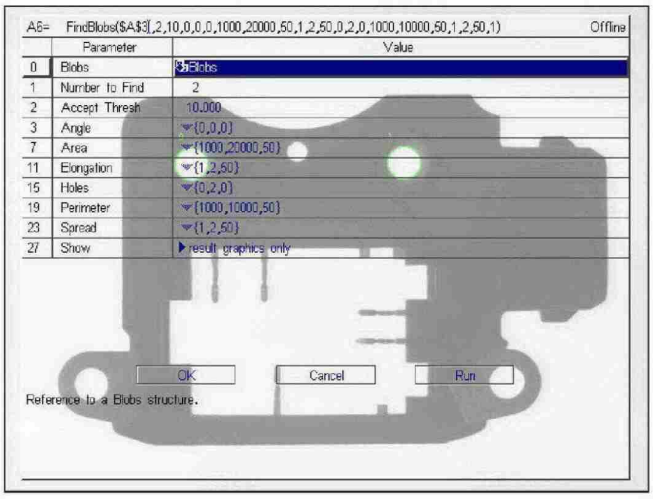
GetNFound Returns the number of Blobs found.

Note	You can configure ExtractBlobs to return more detailed data about each of the Blobs it extracts by entering a value for **Number to Sort** in the Property Sheet.

Lesson: Finding Connected Regions Using FindBlobs

To insert a FindBlobs function into the spreadsheet:

1. In the spreadsheet, verify that a Blobs Structure containing the ExtractBlobs function is present.
2. In column **A**, select a cell in an empty row that also has one empty row immediately above and below for each Blob you expect to report. For example, if you expect five Blobs, you would insert FindBlobs into a cell with one empty row immediately above it and five empty rows immediately below it.
3. Press ⊗ to open the Formula Editor.
4. Highlight **FindBlobs** in the **Blob** subcategory of **Vision Processing**, and press ⊗ to open the FindBlobs Property Sheet.



5. Highlight the **Blobs** parameter value in row 0, and press ⊗ to switch to Cell Selection mode. Highlight the Blobs Structure cell containing the ExtractBlobs function, and press ⊗ to confirm your selection.

Note	The cell you select in the spreadsheet will provide the source Blob data for the FindBlobs function.

WEBER-0005168
DTX-0255_0048
Confidential
PR01925208
US1450_003692553

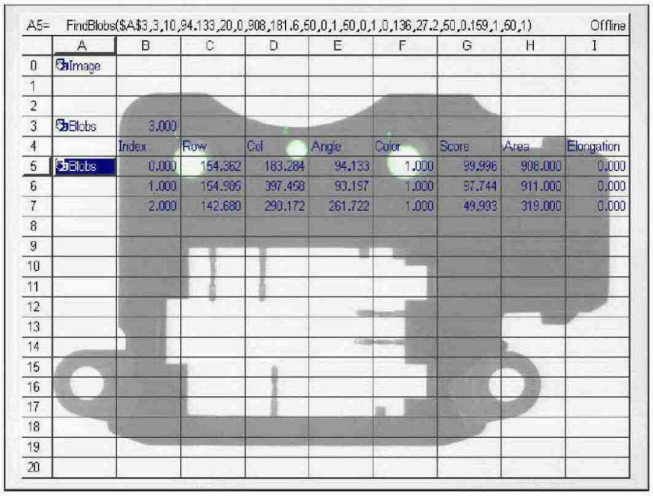
6. Highlight the **Area** parameter group and press ⓞ to open the interactive Blob Selection mode. Press ◁/ ▷ to highlight a Blob in the image, then press ⓧ to confirm your selection..

Note	The Blob you select creates the **Blob Model** for this FindBlobs function. The features of that Blob Model are then automatically entered into the Property Sheet. You can adjust individual Blob Model features manually if necessary to improve recognition.

7. Complete the Property Sheet by specifying the number of Blobs to find.

Note	You can press the **Run** button to see the highlighted boundaries of the Blobs located.

8. Highlight **OK** and press ⓧ to confirm changes.



The FindBlobs function inserts a Blobs Structure in the spreadsheet, plus several Vision Data Access functions that return data describing each Blob found. The most significant of these functions are these:

GetRow	Returns the row centroid (**Row**).
GetCol	Returns the column centroid (**Col**).
GetAngle	Returns the Blob angle (**Angle**).
GetScore	Returns the magnitude of the match (**Score**).

Note	Our objective in this lesson is simply to find Blob objects. We are not concerned about Blob Color, Area, Elongation, Perimeter, or Spread at this point, so you can safely delete these functions from the spreadsheet.

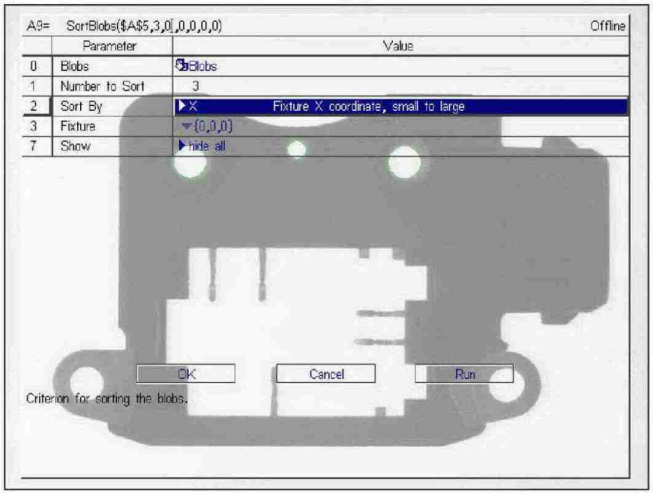
WEBER-0005169
DTX-0255_0049
Confidential
PR01925209
US1450_003692553

GS.book Page 42 Friday, April 14, 2000 4:04 PM

Lesson: Finding Connected Regions Using SortBlobs

To insert a SortBlobs function into the spreadsheet:

1. In column **A**, select a cell in an empty row that also has one empty row immediately above and below for each Blob you expect to report. For example, if you expect five Blobs, you would insert FindBlobs into a cell with one empty row immediately above it and five empty rows immediately below it.
2. Press ⊗ to open the Formula Editor.
3. Highlight **SortBlobs** in the **Blob** subcategory under **Vision Processing**, then press ⊗ to open the SortBlobs Property Sheet.
4. Highlight the **Blobs** parameter value in row 0, and press ⊗ to switch to Cell Selection mode.
5. Highlight the Blobs Structure cell containing the ExtractBlobs function, and press ⊗ to confirm.

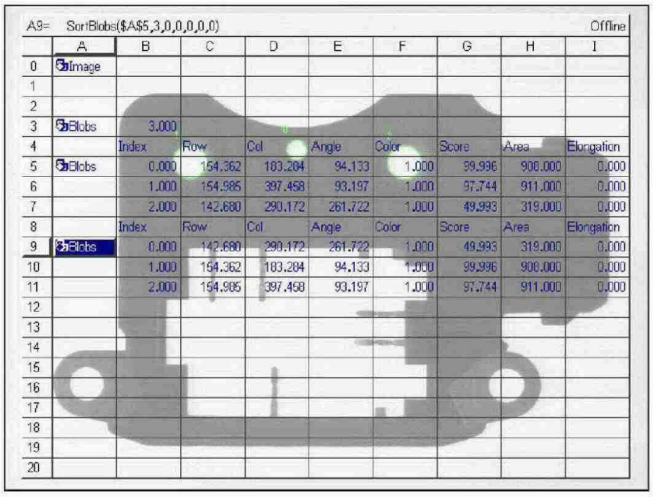


6. Complete the remainder of the Property Sheet by specifying how many Blobs to report, and the sorting method.

Note	You can press the **Run** button to observe the highlighted boundaries of the located Blobs.

WEBER-0005170
DTX-0255_0050
Confidential
PR01925210
US1450_003692553

7. Highlight **OK** and press ⊗ to confirm changes.



The SortBlobs function inserts a Blobs Structure in the spreadsheet, plus several Vision Data Access functions that return data describing the sorted Blobs:

GetRow	Returns the row centroid (**Row**).
GetCol	Returns the column centroid (**Col**).
GetAngle	Returns the Blob angle (**Angle**).
GetScore	Returns the magnitude of the match (**Score**).

WEBER-0005171
DTX-0255_0051
Confidential
PR01925211
US1450_003692553

GS.book Page 44 Friday, April 14, 2000 4:04 PM

Finding Edges

Edges are places in the image where the brightness changes abruptly from dark-to-light or light-to-dark. Edges found by In-Sight are referred to as "line segments", and can be straight or curved, or can even form a complete circle.

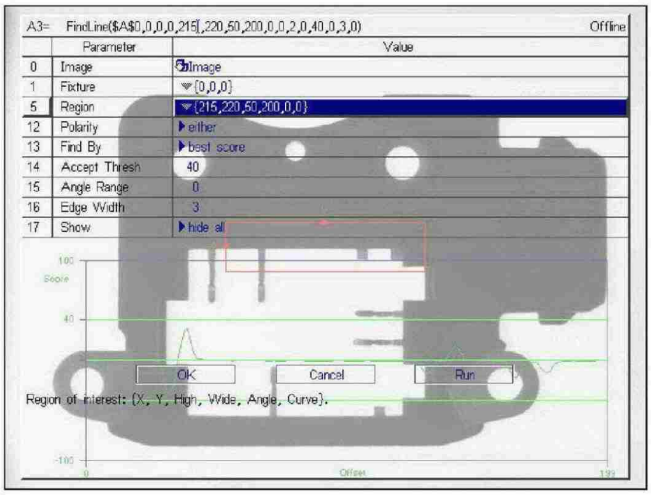
The In-Sight interface provides a variety of functions for locating straight and curved edges:

FindLine	Finds a single straight line segment.
FindSegment	Finds a pair of straight line segments.
FindMultiLine	Finds multiple straight line segments.
FindCurve	Finds an arc segment.
FindCircle	Finds a complete circular edge.

Lesson: Finding Edges Using FindLine

To insert a FindLine function into the spreadsheet:

1. In column **A**, select a cell in an empty row that also has an empty row immediately above it.
2. Press ⊗ to open the Formula Editor.
3. Highlight **FindLine** in the **Edge** subcategory of **Vision Processing**, then press ⊗ to open the FindLine Property Sheet.

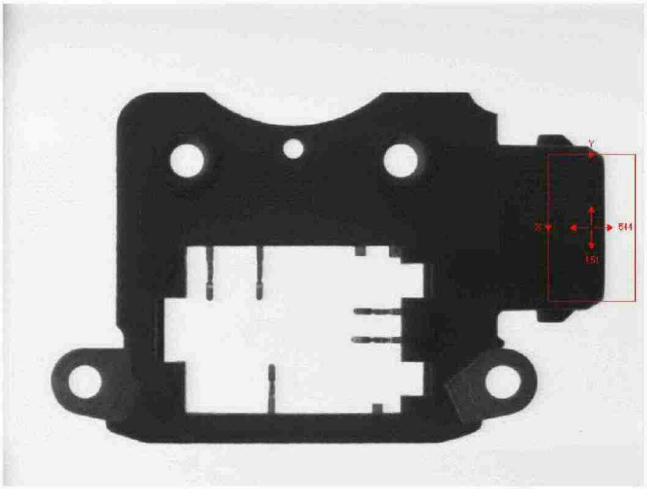


WEBER-0005172
DTX-0255_0052
Confidential
PR01925212
US1450_003692553

4. Highlight the **Region** parameter group and press ⊗ to expand the values. Highlight the X parameter's value, and press ▢ to switch to Interactive Graphics mode.
5. Position the Region so that the Y axis is perpendicular to the edge to be located, then press ⊗ to confirm the changes

Note	The FindLine Region box has two direction arrows, one initially pointing down, and the other pointing right. The arrow pointing down is on the X axis, and the arrow pointing right is on the Y axis. Scanning for straight edges occurs in the direction indicated by the arrow on the Region's Y axis.

.



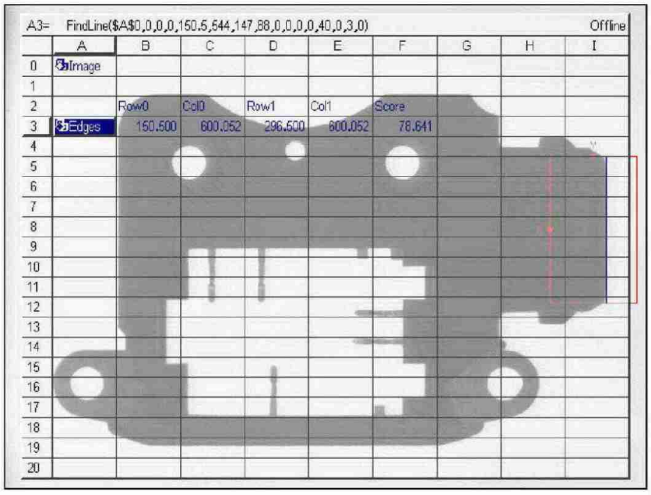
6. Adjust the remaining parameters as necessary.

Note	When parameters in the FindLine Property Sheet refer to black-to-white or white-to-black polarity, they are referring to brightness transitions occurring in the scan direction.

Tip	You can press the **Run** button to observe the peak edge response relative to the threshold.

WEBER-0005173
DTX-0255_0053
Confidential
PR01925213
US1450_003692553

7. Highlight **OK** and press ⊗ to confirm changes.



The FindLine function inserts an Edges Structure in the spreadsheet, along with several Vision Data Access functions that return data about the edge segment found:

GetRow	Returns the starting and ending row coordinates (**Row0** and **Row1**).
GetCol	Returns the starting and ending column coordinates (**Col0** and **Col1**).
GetScore	Returns the magnitude of the edge peak (**Score**).

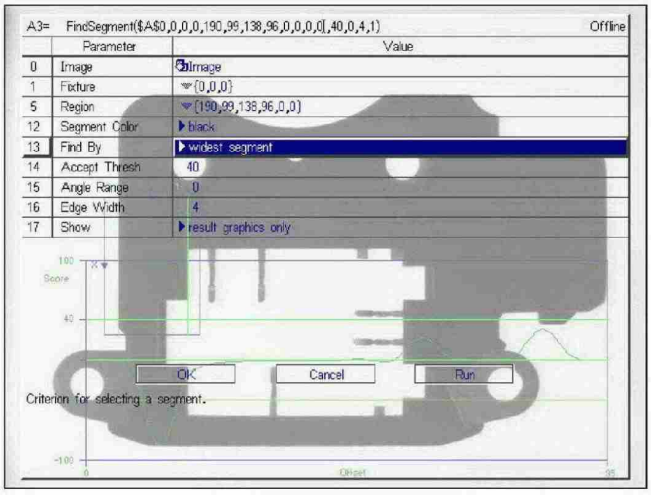
Note The reported score is positive when the FindLine Region is oriented to scan from dark to bright features, and negative when scanning from bright to dark features.

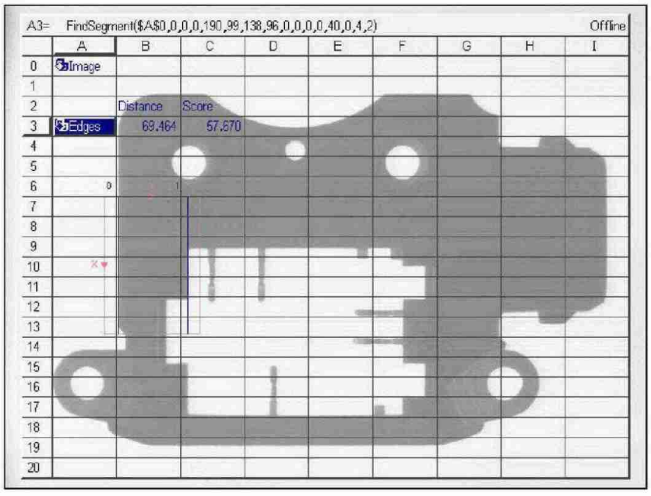
WEBER-0005174
DTX-0255_0054
Confidential
PR01925214
US1450_003692553

Lesson: Finding Edges Using FindSegment

The FindSegment function is similar to FindLine, but instead of finding and reporting a single straight edge, FindSegment locates a pair of edges and computes the distance between them.

You configure the FindSegment function in the same way as the FindLine function.





WEBER-0005175
DTX-0255_0055
Confidential
PR01925215
US1450_003692553

GS.book Page 48 Friday, April 14, 2000 4:04 PM

The FindSegment function inserts an Edges Structure in the spreadsheet, along with two additional Vision Data Access functions:

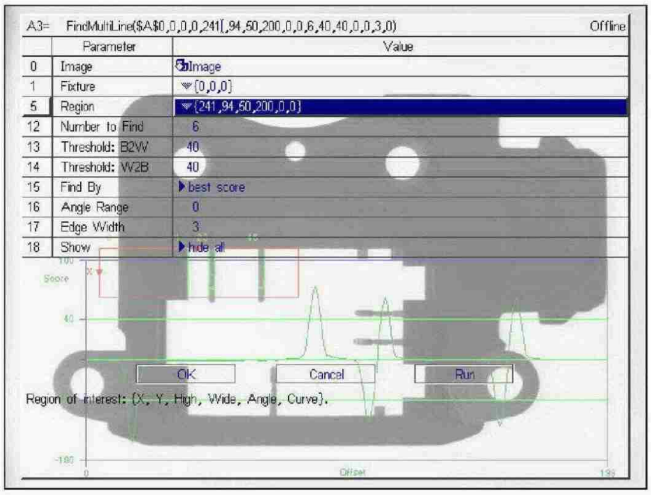
PairDistance Returns the distance between the two edge segments (**Distance**).

GetScore Returns the average magnitude of the two edge peaks (**Score**).

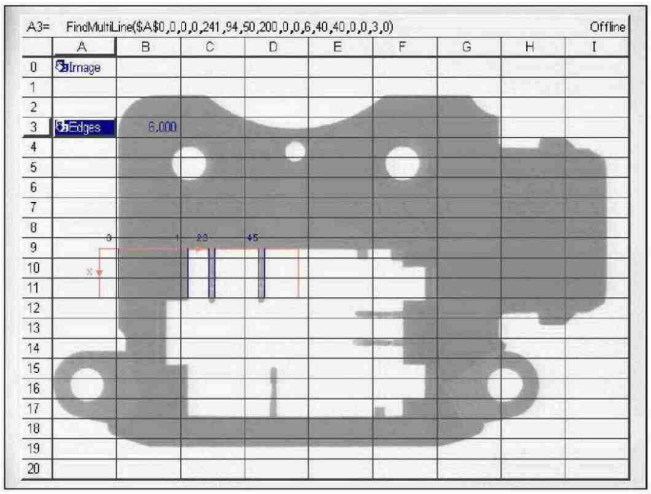
Lesson: Finding Edges Using FindMultiLine

The FindMultiLine function is also similar to FindLine, but instead locates multiple straight edges. Edges found by FindMultiLine can be sorted into edge pairs, and their statistics computed. The lessons that follow discuss functions you can use to pair edges and compute edge pair statistics.

You configure the FindMultiLine function in the same way as the FindLine function.



WEBER-0005176
DTX-0255_0056
Confidential
PR01925216
US1450_003692553



The FindMultiLine function inserts an Edges Structure in the spreadsheet, along with the Vision Data Access function GetNFound:

GetNFound Returns the number of edges found.

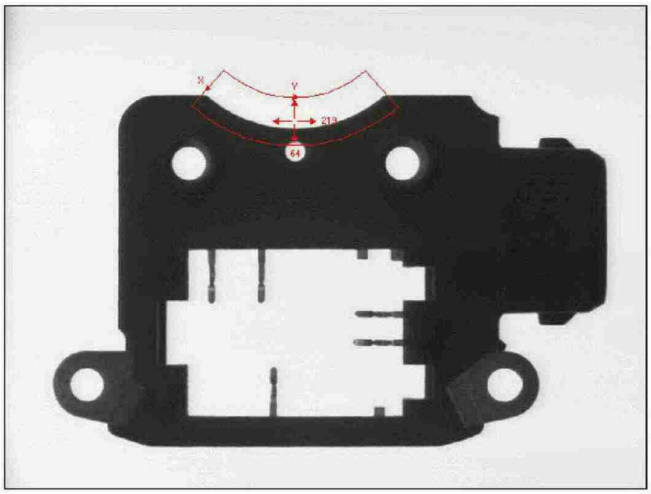
Note	Unlike **FindLine**, **FindMultiLine** does not automatically place row and column coordinate access functions into the **Edges** Structure for all straight edges found. If you need to access the coordinates of a specific edge or pair of edges, you can manually insert the **GetRow** and **GetCol** functions into the spreadsheet.

Lesson: Finding Edges Using FindCurve

To insert a FindCurve function into the spreadsheet:

1. In column **A**, select a cell in an empty row that also has an empty row immediately above it.
2. Press ⊗ to open the Formula Editor.
3. Highlight **FindCurve** in the **Edge** subcategory under **Vision Processing**, then press ⊗ to open the FindCurve Property Sheet.
4. Highlight the **Region** row and press ⊗ to expand the values.
5. Highlight the **X** value and press ▢ to enter Interactive Graphics mode.
6. Press ◎ to cycle through the available Interactive Graphics modes, stopping on the curved mode.
7. Adjust the Region so that its Y axis is concentric with the curved edge to be located.

WEBER-0005177
DTX-0255_0057
Confidential
PR01925217
US1450_003692553



8. Press ⊗ to confirm the changes.

Note	The FindCurve Region box has two direction arrows, one initially pointing down, and the other pointing right. The arrow pointing down is on the X axis, and the arrow pointing right is on the Y axis. Scanning for curved edges occurs in the direction indicated by the arrow on the Region's X axis.

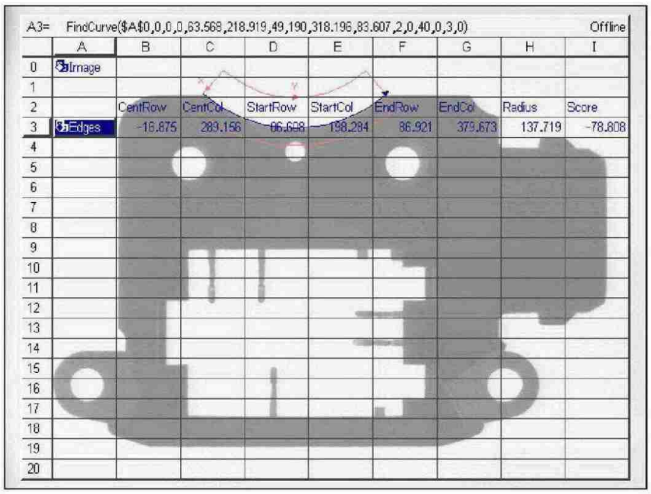
9. Adjust the remaining parameter values as needed.

Note	When parameters in the FindCurve Property Sheet refer to black-to-white or white-to-black polarity, they are referring to brightness transitions occurring in the scan direction.

10. Highlight **OK** and press ⊗ to confirm changes.

WEBER-0005178
DTX-0255_0058
Confidential
PR01925218
US1450_003692553

GS.book Page 51 Friday, April 14, 2000 4:04 PM



The FindCurve function inserts an Edges Structure in the spreadsheet, along with several Vision Data Access functions that return data about the curved edge that was found:

GetRow	Returns the center, starting, and ending row coordinates (**CentRow**, **StartRow**, and **EndRow**).
GetCol	Returns the center, starting, and ending column coordinates (**CentCol**, **StartCol**, and **EndCol**).
GetRadius	Returns the **Radius**.
GetScore	Returns the magnitude of the edge peak (**Score**).

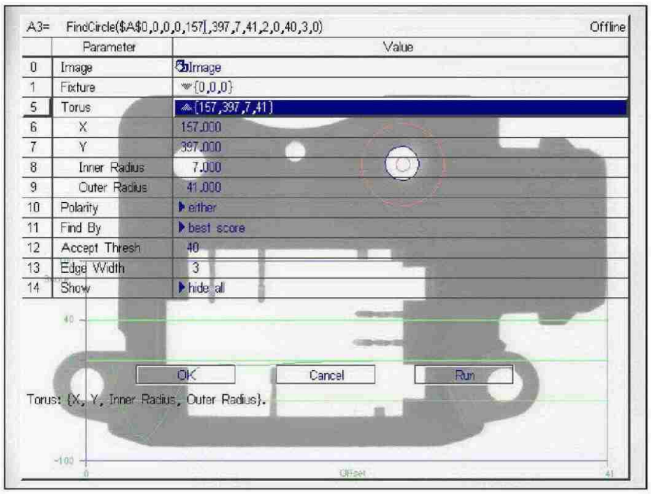
Note The reported score is positive when the **FindCurve** Region is oriented to scan from dark to bright features, and negative when scanning from bright to dark features.

Lesson: Finding Edges Using FindCircle

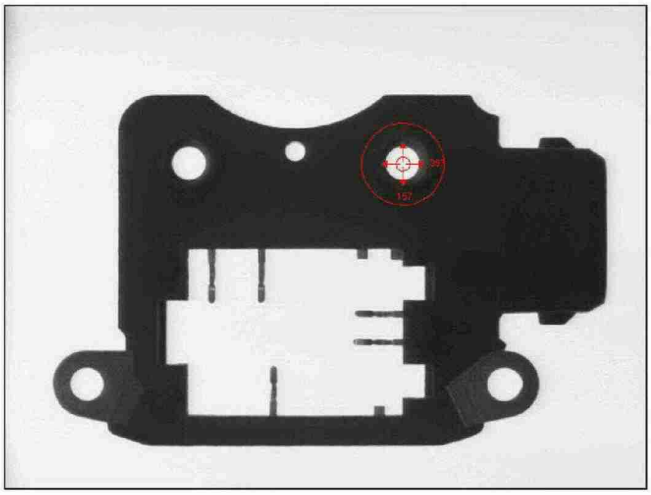
The FindCircle function is similar to FindCurve, but instead of finding and reporting an arc segment, it reports an edge along a complete, 360-degree circle.

You configure a FindCircle function in the same way as other edge finding functions, except that instead of defining a Region with Curve and Angle, you define a Torus with an Inner Radius and Outer Radius.

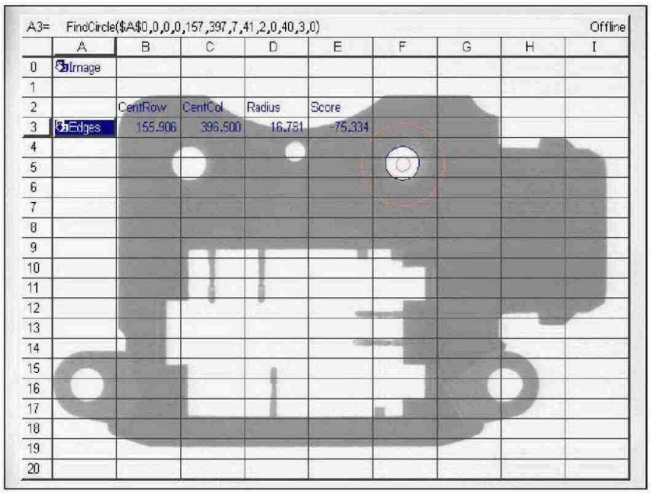
WEBER-0005179
DTX-0255_0059
Confidential
PR01925219
US1450_003692553



Note	The interactive graphics used to position the concentric ring in a FindCircle Region are different from used by **FindCurve.** Use **(A) / (V)** to enlarge/reduce the inner circle, and **(<) / (>)** to enlarge/reduce the outer circle.



WEBER-0005180
DTX-0255_0060
Confidential
PR01925220
US1450_003692553



The FindCircle function inserts an Edges Structure in the spreadsheet, along with several Vision Data Access functions that return data about the circular edge that was found:

GetRow	Returns the center row coordinate (**CentRow**).
GetCol	Returns the center column coordinate (**CentCol**).
GetRadius	Returns the radius (**Radius**).
GetScore	Returns the magnitude of the edge peak (**Score**).

Note	Scanning for circular edges occurs from the center of the circle outward. When parameters in the FindCircle Property Sheet refer to black-to-white or white-to-black polarity, they are referring to brightness transitions occurring in the scan direction.

Finding Patterns

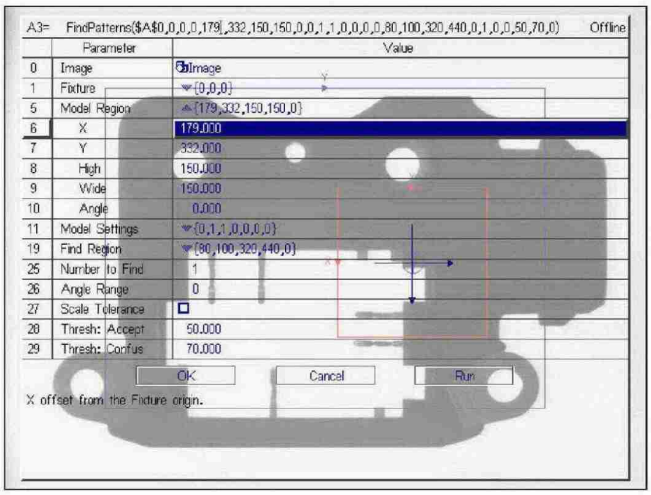
In-Sight can locate objects and other image features in the field of view by "training" a model of a pattern, then searching subsequent images for matching patterns. In-Sight's FindPatterns function combines model training and pattern matching into a single function.

Lesson: Finding Patterns Using FindPatterns

To insert a FindPatterns function into the spreadsheet:

WEBER-0005181
DTX-0255_0061
Confidential
PR01925221
US1450_003692553

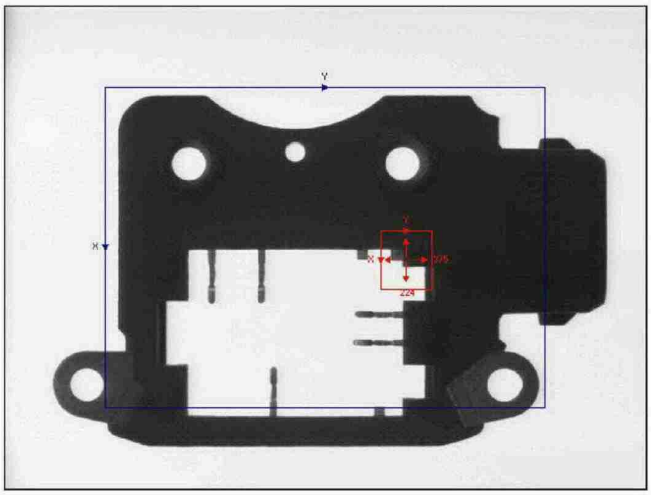
1. In column **A**, select a cell in an empty row that also has one empty row immediately above and below for each pattern you expect to report. For example, if you expect to find two patterns, you would insert FindPatterns into a cell with one empty row immediately above it and two empty rows immediately below it.
2. Press ⊗ to open the Formula Editor.
3. Highlight **FindPatterns** in the **PatFind** subcategory under **Vision Processing**, and press ⊗ to open the FindPatterns Property Sheet.



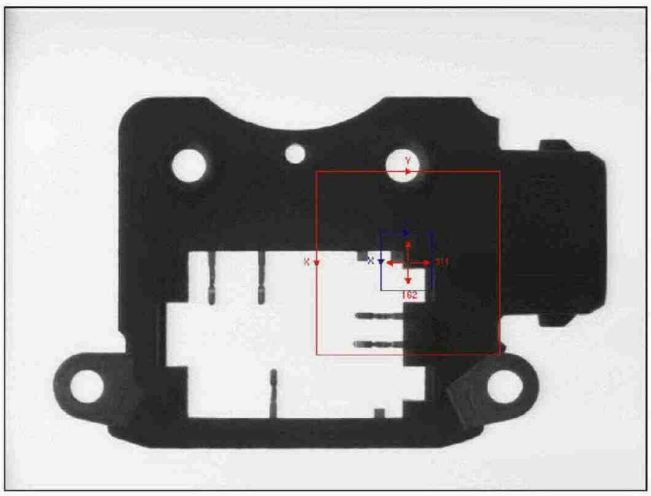
4. Highlight **Model Region** and press ⊗ to expand the group.

WEBER-0005182
DTX-0255_0062
Confidential
PR01925222
US1450_003692553

5. Highlight the **X** parameter and press ⊡ to enter Interactive Graphics mode.



6. Position the Region so that it completely encloses the model you intend to train, then press ⊗ to confirm changes.
7. Highlight **Find Region** and press ⊗ to expand the values. Highlight the **X** parameter and press ⊡ to enter Interactive Graphics mode.



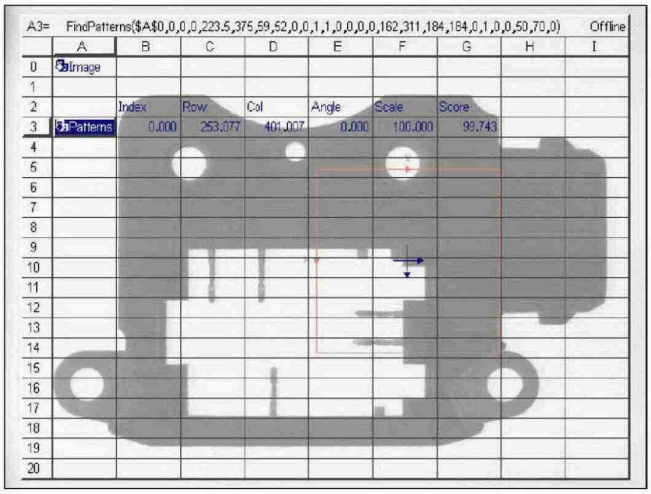
WEBER-0005183
DTX-0255_0063
Confidential
PR01925223
US1450_003692553

GS.book Page 56 Friday, April 14, 2000 4:04 PM

8. Position the Region so that it completely encloses the section of the image containing the pattern to be located, then press ⊗ to confirm changes.
9. Complete the remainder of the Property Sheet.

Note	You can press the **Run** button to observe crosses positioned on pattern matches found in the image.

10. Highlight **OK** and press ⊗ to confirm changes.



The **FindPatterns** function inserts a **Patterns** Structure in the spreadsheet, along with several **Vision Data Access** functions that return data about the pattern that was found:

GetRow	Returns the row coordinate (**Row**).
GetCol	Returns the column coordinate (**Col**).
GetScale	Returns the pattern scale offset (**Scale**).
GetAngle	Returns the pattern angle offset (**Angle**).
GetScore	Returns the magnitude of the match (**Score**).

WEBER-0005184
DTX-0255_0064
Confidential
PR01925224
US1450_003692553

CHAPTER 9: USING FEATURES TO FIND OBJECTS

In this chapter, you'll learn how to use the results of the features found in Chapter 8 to reliably find objects.

Why Combine Features?

When an object appears in the field of view, its position is never perfectly aligned in the expected location. In other words, there is always some degree of positional uncertainty. To compensate, you will sometimes need to create a *Fixture* using one or more located features. You can then use this Fixture to help track the object in the field of view, and to reliably position tools for subsequent inspections and measurements. You can create a Fixture from any set of coordinates consisting of a *row*, *column*, and *theta*. However, individual features are often impractical sources for Fixture coordinates. Consider the following:

- By the very nature of the edge algorithm, edge-finding routines such as FindLine can only work over a limited range of rotation.
- Not all Connected Regions found by ExtractBlobs have well-defined orientations, and so most Blobs cannot by themselves return the data needed for a Fixture. For example, you could not create a Fixture from a hole, because it has no reliable orientation, and therefore it cannot provide the theta coordinate that is needed.
- The FindPatterns function has the fewest theoretical limitations, but can be limited by performance requirements. Searches for rotated or scaled patterns within large image Regions demand significant computational resources, and require a geometrically stable grayscale pattern with a clearly recognizable orientation.
- Combining features helps overcome the limitations that all feature locating functions have in their ability to report orientation. In-Sight's flexible spreadsheet environment enables you to use any convenient combination of edge-, pattern-, and Blob -finding functions to establish the location and orientation of an object relative to a Fixture.

For convenience in combining the results of feature locating functions, In-Sight also supplies several standard geometry functions. The most important of these functions are the following:

- **PointToLine**
- **LineToLine**
- **PointToPointAngle**

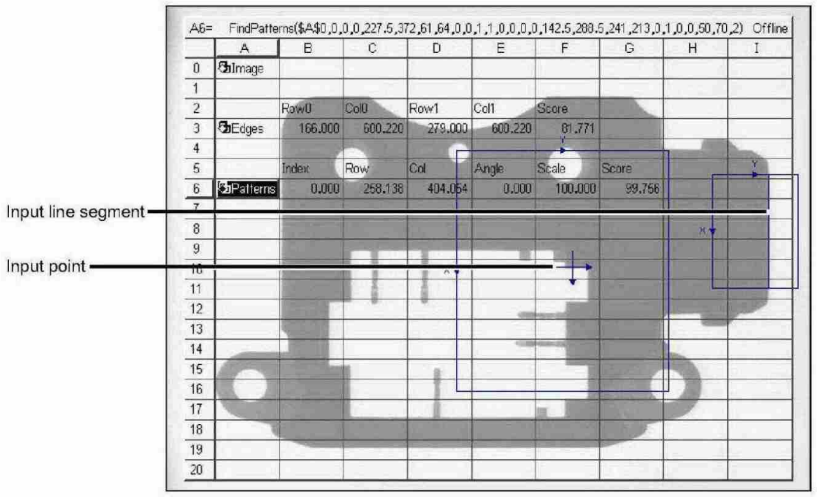
The following lessons show how to use these functions, and the chapter's final lesson shows how to create a Fixture from combined feature results.

WEBER-0005185
DTX-0255_0065
Confidential
PR01925225
US1450_003692553

GS.book Page 58 Friday, April 14, 2000 4:04 PM

Lesson: Using PointToLine

The PointToLine function takes two inputs: a point, defined by (Row, Col), and a line segment, defined by (Row0,Col0,Row1,Col1). PointToLine creates a Structure called *Dist* (short for distance) that returns the coordinates, angle, and distance of a line segment that lies perpendicular to the input line segment, and passes through the input point.

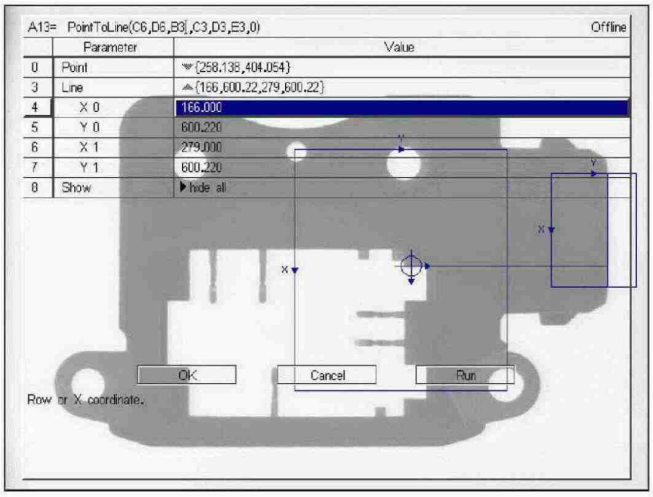


To insert a PointToLine function into the spreadsheet:

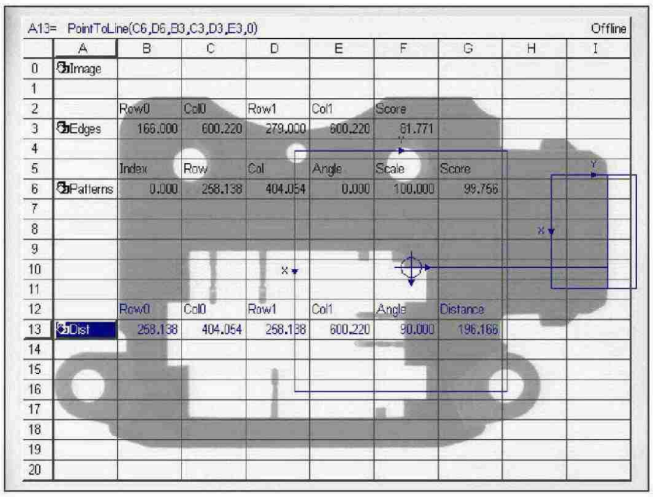
1. In column **A**, select a cell in an empty row that also has an empty row immediately above it.
2. Press ⊗ to open the Formula Editor.
3. Highlight **PointToLine** in the **Measure** subcategory under **Geometry**, and press ⊗ to open the PointToLine Property Sheet.
4. Highlight the **Point** row and press ⊗ to expand the values, then highlight the **X** value and press ⊗ to enter Cell Selection mode.
5. Highlight the spreadsheet cell that contains the row value of the point you want, then press and hold ◎ and mark a block of cells that contains that row value and an adjacent column value.

WEBER-0005186
DTX-0255_0066
Confidential
PR01925226
US1450_003692553

6. Press ⊗ to confirm your selection.



7. Highlight **Line** and press ⊗ to expand the data, then highlight **X 0** in the expanded data and press ⊗ to enter Cell Selection mode.
8. Highlight the spreadsheet cell that contains the row value of the first point of the line segment, then press and hold ◎ and mark a block that contains the row and column values for the starting and ending points of the line segment. Press ⊗ to confirm your selection.
9. Highlight **OK** and press ⊗ to confirm your changes.



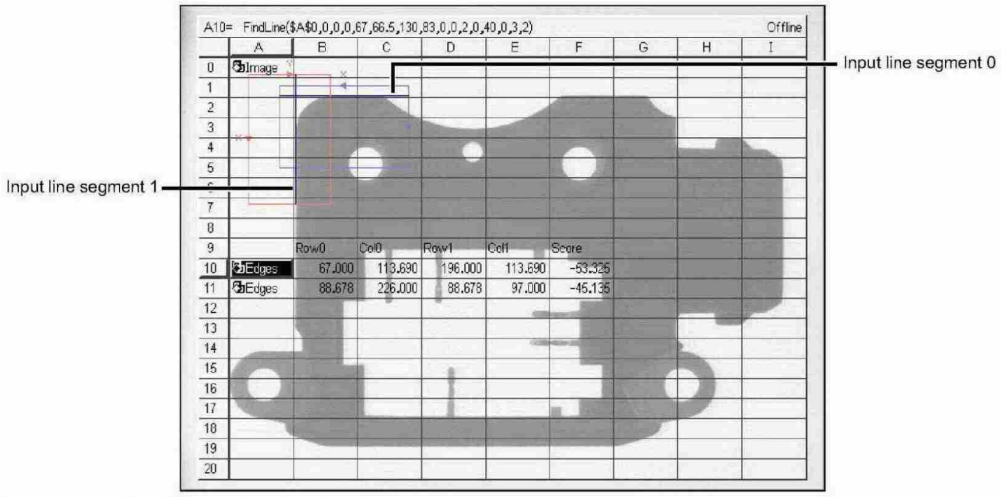
WEBER-0005187
DTX-0255_0067
Confidential
PR01925227
US1450_003692553

The PointToLine function automatically inserts a **Dist** Structure in the spreadsheet, along with several Vision Data Access functions that return data about the newly created line segment between the input point and the input line segment:

GetRow	Returns the starting and ending row coordinates (**Row0** and **Row1**).
GetCol	Returns the starting and ending column coordinates (**Column0** and **Column1**).
GetAngle	Returns the **Angle** between the input point and line segment.
GetDistance	Returns the **Distance** between the input point and the line segment.

Lesson: Using LineToLine

The LineToLine function takes as its inputs a pair of line segments defined by (Row0,Col0,Row1,Col1). From these, it creates a Dist Structure that returns the coordinates, angle, and distance of a line segment that intersects the two input lines. The starting and ending points of the Dist line segment are the points of intersection with the two input lines.

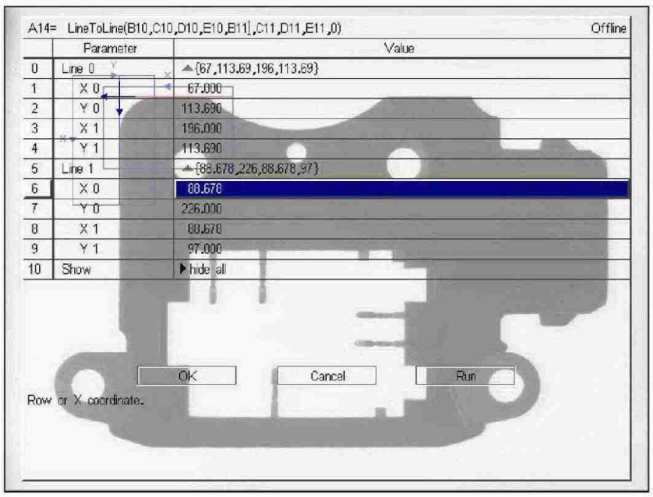


To insert a LineToLine function into the spreadsheet:

1. In column **A**, select a cell in an empty row that also has an empty row immediately above it.
2. Press ⊗ to open the Formula Editor
3. Highlight **LineToLine** in the **Measurements** subcategory under **Geometry**, and press ⊗ to open the LineToLine Property Sheet.
4. Highlight **Line 0** and press ⊗ to expand the values, then highlight **X 0** in the expanded data and press ⊗ to enter Cell Selection mode.

WEBER-0005188
DTX-0255_0068
Confidential
PR01925228
US1450_003692553

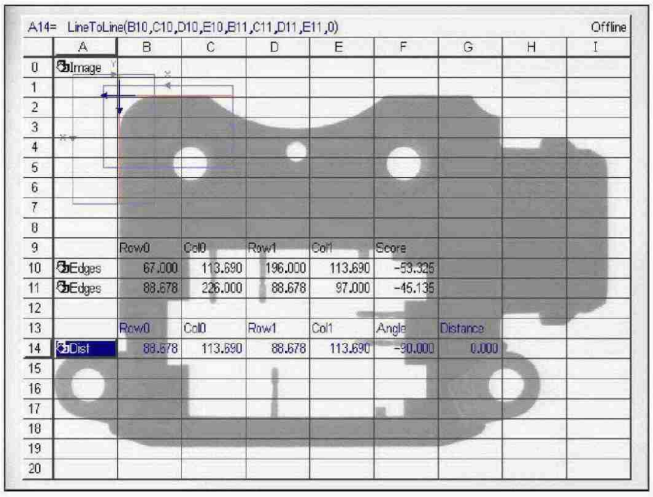
5. Highlight the spreadsheet cell that contains the row value for the starting point of the first line segment, then press and hold ◎ and mark a block of cells that contains the row and column values for the starting and ending points of the first line segment.
6. Press ⊗ to confirm your selection.



7. Highlight **Line 1** and press ⊗ to expand the values, then highlight **X 0** in the expanded data and press ⊗ to enter Cell Selection mode.
8. Highlight the spreadsheet cell that contains the row value for the starting point of the second line segment, then press and hold ◎ and mark a block of cells that contains the row and column values for the starting and ending points of the first line segment.
9. Press ⊗ to confirm your selection.

WEBER-0005189
DTX-0255_0069
Confidential
PR01925229
US1450_003692553

10. Highlight **OK** and press ⊗ to confirm your changes.



The LineToLine function inserts a Dist Structure in the spreadsheet, along with several Vision Data Access functions that return data about newly created line segment:

GetRow	Returns the starting and ending row coordinates (**Row0** and **Row1**).
GetCol	Returns the starting and ending column coordinates (**Column0** and **Column1**).
GetAngle	Returns the **Angle** between the two input lines.
GetDistance	Returns the **Distance** between the two lines.

Note	The **GetDistance** function returns a 0 if the two input lines intersect. If the lines are parallel, GetDistance returns a positive value and GetAngle returns a 0.

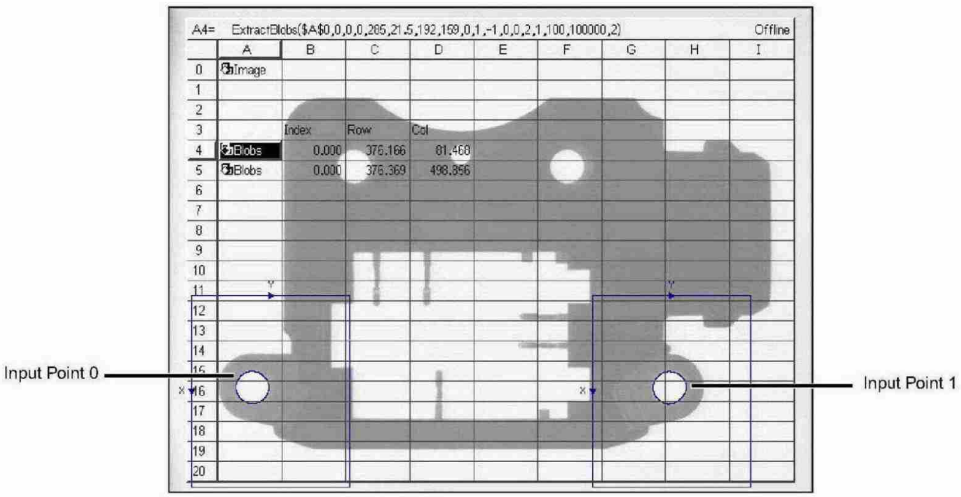
WEBER-0005190
DTX-0255_0070
Confidential
PR01925230
US1450_003692553

GS.book Page 63 Friday, April 14, 2000 4:04 PM

Lesson: Using PointToPointAngle

One of the more common methods of combining results is to find two features of an object, then use the angle between them to define the object's orientation.

The **PointToPointAngle** function takes as its inputs two points defined by (Row0,Col0,Row1,Col1), and from these creates a Dist Structure that returns the angle of a line segment that runs between the two input points.

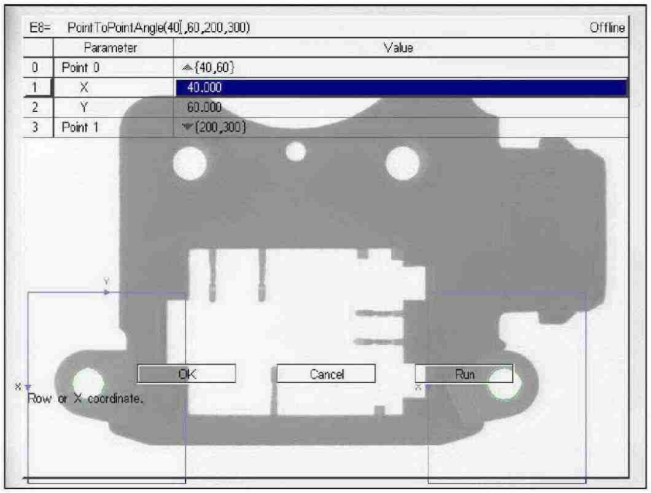


To insert a PointToPointAngle function into the spreadsheet:

1. In column **A**, select a cell in an empty row that also has an empty row immediately above it.
2. Press Ⓧ to open the Formula Editor
3. Highlight **PointToPointAngle** in the **Measurements** subcategory under **Geometry**, and press Ⓧ to open the PointToPointAngle Property Sheet.
4. Highlight **Point 0** and press Ⓧ to expand the values, then highlight **X** in the expanded values and press Ⓧ to enter Cell Selection mode.
5. Highlight the spreadsheet cell that contains the row value for the first point, then press and hold ◎ and mark a block of cells that contains the row value and an adjacent column value.

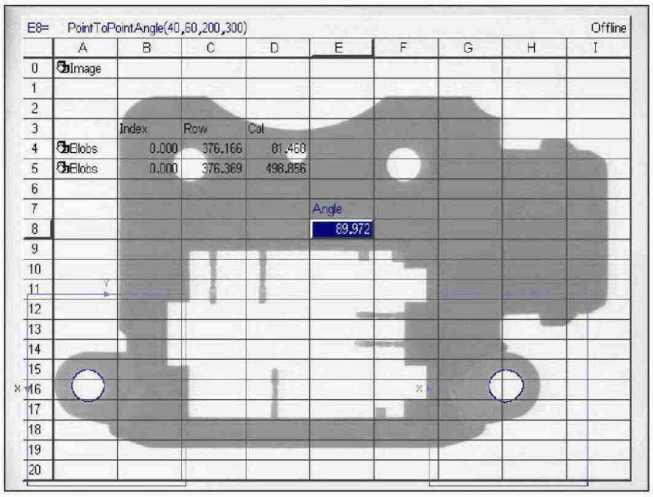
WEBER-0005191
DTX-0255_0071
Confidential
PR01925231
US1450_003692553

6. Press ⊗ to confirm your selection.



7. Highlight **Point 1** and press ⊗ to expand the values, then highlight **X** in the expanded values and press ⊗ to enter Cell Selection mode.
8. Highlight the spreadsheet cell that contains the row value for the second point. Press and hold ◎ to mark a block of cells that contains the row value and an adjacent column value.
9. Press ⊗ to confirm your selection.

Highlight **OK** in the Formula Editor, and press ⊗ to confirm the **PointToPoint** function.



WEBER-0005192
DTX-0255_0072
Confidential
PR01925232
US1450_003692553

GS.book Page 65 Friday, April 14, 2000 4:04 PM

The PointToPointAngle function inserts a single function into the spreadsheet:

GetAngle Returns the **angle** between the two input points.

Note	The **GetAngle** function simply returns the angle value to the spreadsheet. You may want to add a label for this value in an adjacent cell, as shown in the figure above.

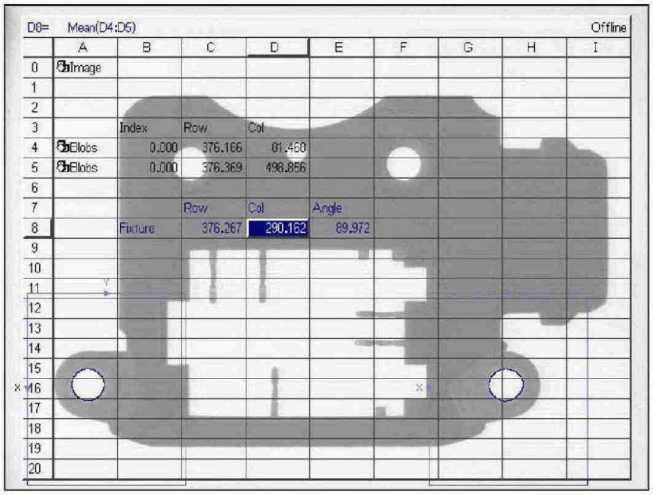
Lesson: Creating a Fixture

You can create a Fixture (defined by row, column, and theta values) from the coordinate locations of one or more located features. Once you have defined a Fixture, subsequent functions can refer to it to reliably position their Regions within the field of view.

Creating a Fixture

The most common type of Fixture uses the average row and column coordinates from two features, and the angle of rotation between them.
To create a Fixture:

1. Identify the coordinates in the spreadsheet for the two points you will use for the Fixture, and (if necessary) determine the angle of rotation between those points using the GetAngle function.
2. Highlight an empty cell and press ⊗ to open the Formula Editor. Enter a formula that calculates the average row value between the two points.
3. Press ▷ to highlight the empty cell adjacent to the row formula. Repeat step 2 to enter an average column value formula between the two points.
4. Press ▷ again to highlight the cell adjacent to the column value formula, then press and hold ⊗ and highlight **Relative** to enter Cell Selection mode.



WEBER-0005193
DTX-0255_0073
Confidential
PR01925233
US1450_003692553

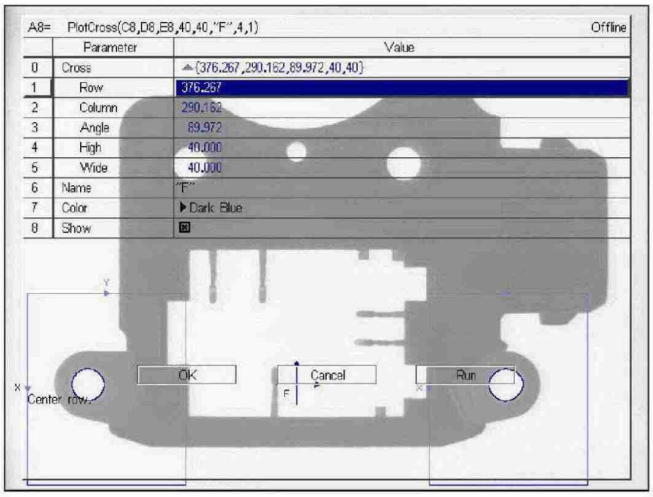
.

Note	It is generally a good idea to identify Fixture coordinates in the spreadsheet by adding a text label to an adjacent empty cell. This makes the Fixture coordinates easier to identify when referencing them from within other functions.

Plotting the Fixture

Once you have created a Fixture, it is useful to plot its location on the image to make it easy to find. To plot a fixture:

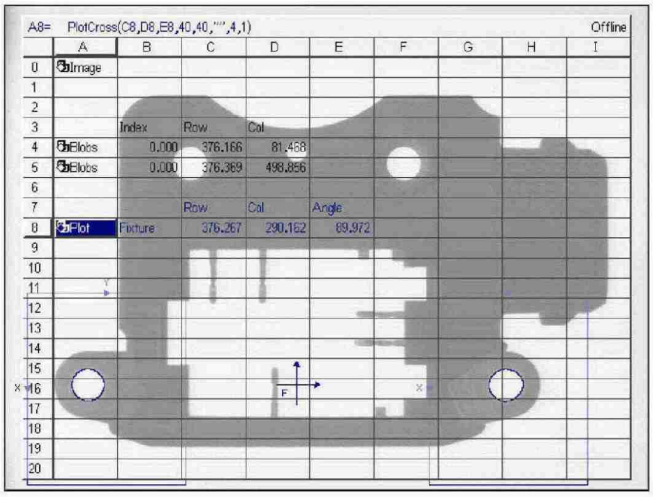
1. Highlight an empty cell and press ⊗ to open the Formula Editor.
2. Under the **Image** function subcategory of **Graphics**, highlight **PlotCross** and press ⊗ to open its Property Sheet.
3. In the PlotCross Property Sheet, expand the **Cross** group of parameters, then highlight the**Row** value and press ⊗ to enter Cell Selection mode.



4. Press and hold ◎, then press ▷ to highlight the cells containing the row, column, and angle coordinates of the Fixture.
5. Press ⊗ to confirm your selection and return to the PlotCross Property Sheet.
6. Configure the rest of the PlotCross Property Sheet. You can change the size and color of the cross, and assign a name.

WEBER-0005194
DTX-0255_0074
Confidential
PR01925234
US1450_003692553

7. Press **OK** to accept your changes and return to the spreadsheet.



A cross then appears on top of the image at the Fixture coordinate location. Any function that refers to the Fixture will position itself in the image relative to this location.

The following chapter includes an example of how to use a Fixture by referencing it from within another function.

WEBER-0005195
DTX-0255_0075
Confidential
PR01925235
US1450_003692553

GS.book Page 68 Friday, April 14, 2000 4:04 PM

WEBER-0005196
DTX-0255_0076
Confidential
PR01925236
US1450_003692553

CHAPTER 10: MAKING MEASUREMENTS

This chapter will show you how to use In-Sight's image measurement techniques.

What is a Measurement?

Machine vision inspections rely on *measurements* to yield decisions, where a measurement can be any useful data derived from the features located on an object.

In the previous lesson, we laid the groundwork for making measurements by describing various techniques for locating objects in the field of view. Once you register an object in the image by its (**Row**, **Column**, and **Theta**) coordinates, then measurements can be made in specific regions that are *fixtured* relative to the native coordinate system of the object.

Note	If the positional uncertainty of the features to measure is sufficiently small or if the intended inspection is very general, it may not be necessary to register measurements to an object coordinate system using a Fixture.

General measurement tasks might return a count of the number of objects in an image, the coordinates of a single object, or some grayscale statistic across the entire image. In each of these examples, creating a Fixture is unnecessary.

In-Sight's measurement capabilities are grouped into three categories:

- Counting Pixels
- Counting Features
- Measuring Distance

The following lessons discuss each category.

Lesson: Counting Pixels

The most common machine vision measurement is a count of the pixels within a region that fall above or below a specified grayscale threshold level. "Dark" pixels are below the threshold, and "bright" pixels are above the threshold. This simple counting technique is a member of the family of image measurements that derive from the grayscale **Histogram**.

Note	Grayscale images contain up to 256 gray levels, where each pixel has a grayscale value between 0 and 255. The Histogram counts the number of pixels in a region for each gray level. To count the pixels in a region below a specified threshold level, simply sum the Histogram array from zero up to the threshold level.

WEBER-0005197
DTX-0255_0077
Confidential
PR01925237
US1450_003692553

The In-Sight™interface supplies a number of functions that allow you to count pixels and make other useful grayscale measurements. The most important of these functions are:

ExtractHistogram Extract a histogram array from an image region.

HistThresh Compute an adaptive threshold level.

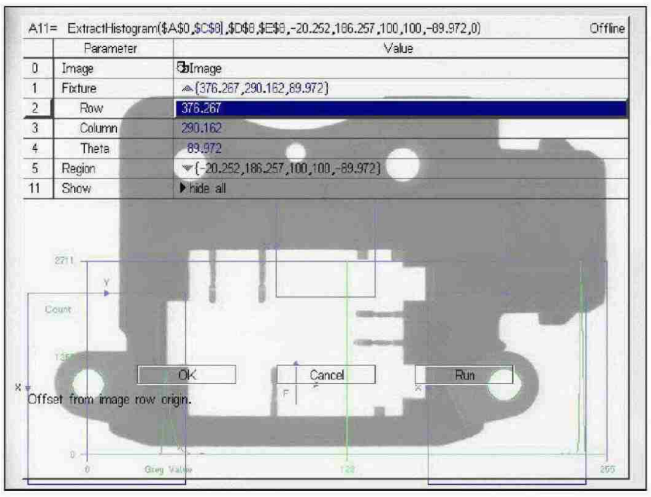
HistCount Sum a specified section of the histogram array.

Using ExtractHistogram

1. To insert the ExtractHistogram function into the spreadsheet:
2. Highlight an empty cell in column **A**. Press ⊗ to open the formula editor.

Note	The row you select plus one row above should be empty.

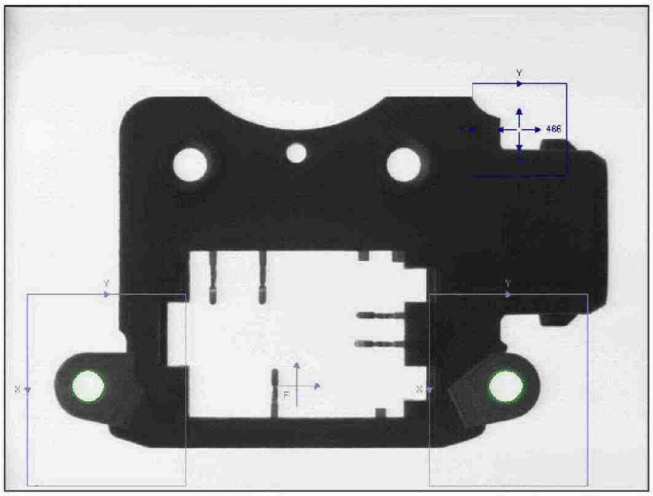
3. Highlight **ExtractHistogram** from the **Histogram** subcategory of the **Vision Processing** functions, and press ⊗ to open the **ExtractHistogram** property sheet.



4. If you have a fixture that you want to use to position this histogram measurement, highlight the **Fixture** parameter group and press ⊗ to expand. Highlight **Row** and press ⊗ to switch to *Cell Selection* mode. Press ▷ while pressing and holding ◎ to block mark the cells containing the (**Row**, **Column**, and **Theta**) coordinates that define the fixture. Press ⊗ to confirm your selection and return to the **ExtractHistogram** property sheet.

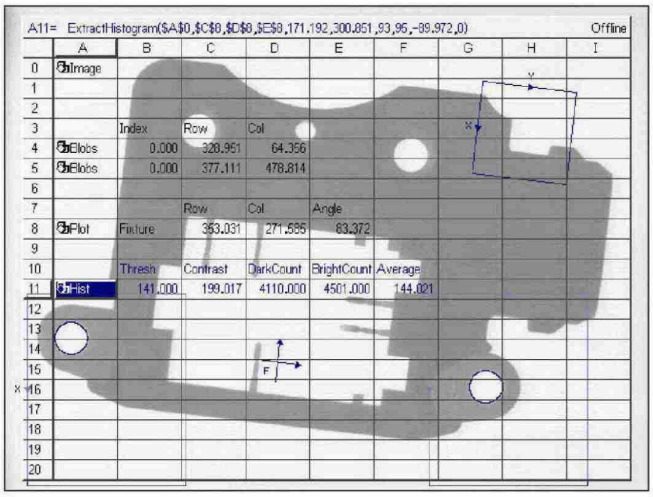
WEBER-0005198
DTX-0255_0078
Confidential
PR01925238
US1450_003692553

5. Highlight the **Region** group and press ⊗ to expand. Highlight the **X** value in the expanded data and press ⊡ to switch to *Interactive Graphics* mode. Position the region over the section of the image from which to extract the histogram. Press ⊗ to confirm changes and return to the **ExtractHistogram** property sheet.



6. Highlight **OK** in the property sheet, and press ⊗ to confirm changes.

The **ExtractHistogram** function is now positioned relative to the *Fixture* point. As the object's location and orientation vary, the *Fixture* tracks the object, and the **ExtractHistogram** function operates within the intended region.



WEBER-0005199
DTX-0255_0079
Confidential
PR01925239
US1450_003692553

GS.book Page 72 Friday, April 14, 2000 4:04 PM

The **ExtractHistogram** function inserts a **Hist** structure in the spreadsheet. It also inserts several related functions that return data from the histogram. These include:

HistThresh	The binary threshold (**Thresh**). "Dark" values are above the grayscale threshold; "bright" values are below the grayscale threshold.
HistContrast	The average grayscale of the bright pixels minus the average grayscale of the dark pixels (**Contrast**).
HistCount	The count of pixels above and below the threshold (**BrightCount** and **DarkCount**, respectively).
HistMean	The **Average** grayscale value of all pixels in the histogram..

Note It is useful to examine the functions created by ExtractHist to help you better understand the use of histogram related functions.

Using HistThresh

As previously described, the **ExtractHistogram** function automatically inserts the **HistThresh** function into the spreadsheet. **HistThresh** sets a reference to the cell containing the **Hist** structure and the range to compute the threshold. The maximum range is 0 to 255.

To change the threshold range:

1. Highlight an unused cell and press ⊗ to open the formula editor.
2. Highlight the **HistThresh** function under **Vision Processing | Histogram** and press ⊗.
3. Use the formula editor to insert a cell reference to an existing **Hist** structure and set the last two arguments to 0 and 255, respectively.
4. Highlight **OK** and press ⊗.

The spreadsheet will now contain a **HistThresh** function that returns a threshold level.

Note The threshold level returned by HistThresh is computed by finding the threshold level that produces the best match between the original grayscale image and a hypothetical binary image. This technique works quite well over a range of image conditions but is not suited for every application.

Using HistCount

This function takes as its input a reference to a cell containing an **ExtractHistogram** function and a range over which to count.

To insert HistCount into the spreadsheet:

1. Highlight an unused cell. Press ⊗ to open the formula editor.
2. Highlight **HistCount** under **Vision Processing | Histogram** and press ⊗.

WEBER-0005200
DTX-0255_0080
Confidential
PR01925240
US1450_003692553

GS.book Page 73 Friday, April 14, 2000 4:04 PM

3. Use the formula editor to insert a cell reference to an existing histogram structure and set the last two arguments to specify the range.

Note	The range to count is commonly set from zero to the threshold level or from the threshold level to 255.

4. Highlight **OK** and press ⊗.

The spreadsheet will now contain a **HistCount** function returning a count of the number of pixels in the histogram array whose grayscale values fall between the specified level.

Lesson: Counting Features

Another common machine vision measurement is a count of features found in a specific region of the image. As described in **Chapter 8, Finding Features**, In-Sight™ finds features using the **Blob**, **Edge**, and **Pattern** functions. An additional function, **GetNFound**, returns the feature count.

Note	For counting purposes, these functions are typically registered to the object being inspected using a Fixture. This decreases the processing time necessary to find the features, and prevents potential confusion with unwanted features.

Tip	When counting Edges and Patterns, be aware that the N2Find parameter in the property sheets of the FindMultiLine and FindPatterns functions determines the maximum number of features that can be reported.

To count Blob, Edge, or Pattern features:

1. Highlight an empty cell and press ⊗ to open the formula editor.
2. Highlight and expand the **Edges**, **Blobs**, or **Patterns** function subcategory under **Vision Data Access**.
3. Highlight the **GetNFound** function in the selected function subcategory and press ⊗**.** The formula editor will open again. Highlight the *Absolute Cell Reference* button and press ⊗ to enter *Cell Selection* mode in the spreadsheet.
4. Highlight the cell that refers to the source **Edges**, **Blobs**, or **Patterns** structure, and press ⊗. Press the **OK** button in the formula editor to return to the spreadsheet.
5. The **GetNFound** function will insert itself into the spreadsheet and return the number found for the selected feature.

Lesson: Measuring Distances

Relationships between image features can be an important measurement. Functions under the **Geometry** heading and **Measure** subcategory are useful for computing feature-to-feature angles and distances, including:

PointToPoint	The distance and angle established by two given points.
PointToLine	The distance and angle of a line segment that lies perpendicular to a given line segment and intersects a given point.

WEBER-0005201
DTX-0255_0081
Confidential
PR01925241
US1450_003692553

LineToLine	The point of intersection and the angle between two line segments.
PointToPointAngle	The angle established by two given points.
PointToPointDistance	The distance between two points.

Refer to **Combining Results** from the previous lesson for more information.

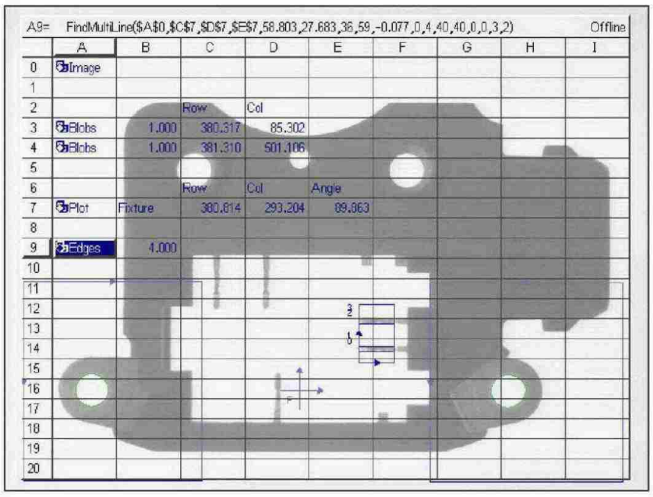
Using PairEdges

Edge pairs consist of two parallel edges that typically define the boundaries of an object feature. In an array of multiple edges, it is also possible to have multiple edge pairs. For example, an **Edges** structure containing six edges can have up to three edge pairs: (0,1), (2,3), and (4,5).

The **PairEdges** function inputs an array of edges and sorts them into pairs according to distance, polarity, and other characteristics.

To identify edge pairs:

1. Insert a **FindMultiLine** function in an empty spreadsheet cell. Verify that at least two edge features are found. The following example shows four edges being located



Note	For this lesson, configure FindMultiLine so its region intersects some repeating pattern of edges. Make sure you set the N2Find parameter large enough to cover the maximum number of edges expected.

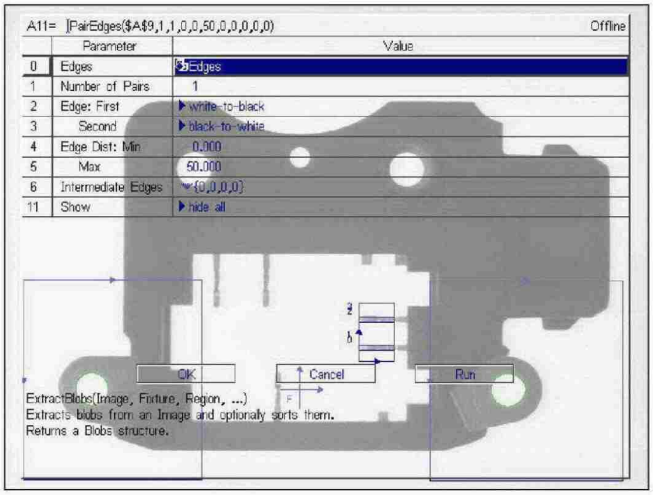
2. Highlight an empty spreadsheet cell below the **Edges** structure created in step 1. Press ⊗ to open the formula editor.

Note	The row you select plus one row above should be empty.

WEBER-0005202
DTX-0255_0082
Confidential
PR01925242
US1450_003692553

GS.book Page 75 Friday, April 14, 2000 4:04 PM

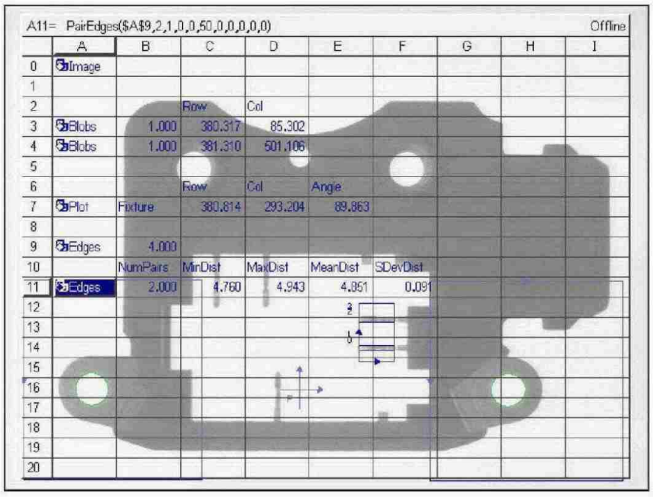
3. Under **Vision Processing**, highlight **Edge | PairEdges** and press ⊗ to open the **PairEdges** property sheet.
4. Highlight the **Edges** parameter and press ⊗ to enter *Cell Selection* mode in the spreadsheet. Highlight the **Edges** structure containing the **FindMultiLine** function and press ⊗ to confirm your selection. Return to the **PairEdges** property sheet.



5. Specify one or more of the following parameters in the **PairEdges** property sheet:
   - The Number of Pairs to find
   - The polarity of the First and Second edges of each pair
   - The Min and Max distance between the two edges of each pair
   - The minimum and maximum number of Intermediate Edges that come between the two edges of each pair

WEBER-0005203
DTX-0255_0083
Confidential
PR01925243
US1450_003692553

6. Highlight **OK** and press ⓧ. Return to the spreadsheet.



The spreadsheet will now contain the **PairEdges** function, plus several functions that report statistics about the edge pairs found:

NumPairs	The number of edge pairs found (GetNFound).
MinDist	The shortest distance between the two edges in a pair (PairMinDistance).
MaxDist	The farthest distance between the two edges in a pair (PairMaxDistance).
MeanDist	The average distance between the two edges in a pair (PairMeanDist).
SDevDist	The standard deviation of the distance between two edges in a pair (PairSDevDistance).

WEBER-0005204
DTX-0255_0084
Confidential
PR01925244
US1450_003692553

CHAPTER 11: CREATING DECISION POINTS

In this chapter, you will learn several basic techniques for converting measurements into decisions.

What is a Decision Point?

The ultimate goal of any visual inspection is to make some type of decision about the object that is inspected: Are the object's dimensions within tolerance? Does the object appear in the expected location? Is the object missing a feature? These are the types of inspection decisions that In-Sight can make.

But first, it is necessary to convert measurements, such as those discussed in the previous chapter, into *decision points*. Decision points segment measurements into specific categories. A *decision point* can be any unambiguous inspection result, such as:

- Yes/No
- Good/Bad
- Pass/Fail
- 1 / 0
- Present/Absent

A *decision point* might also take the form of a number containing an encoded inspection result for the purposes of classification, such as:

- Defect type
- Part ID
- Quality level

Converting Measurements into Decision Points

Creating measurement-based decisions can be a simple or complex task. At the simplest level, making a decision from a measurement can be based on a simple threshold score that determines the pass or fail status of an object inspection. Alternatively, a more complex decision point can be based on a logical expression involving multiple measurements designed to select one of several possible quality levels.

The In-Sight interface provides enormous flexibility to create virtually any type of decision point using standard spreadsheet operators and expressions. The following includes a description of some of those operators.

WEBER-0005205
DTX-0255_0085
Confidential
PR01925245
US1450_003692553

GS.book Page 78 Friday, April 14, 2000 4:04 PM

Comparison

Under the **Mathematics** heading, the **Operators** category contains a complete selection of comparison operators, including:

>	Greater than
<	Less than
=	Equal to
>=	Greater than or equal to
<=	Less than or equal to

Logical

Under the **Mathematics** heading, the **Logic** category contains a selection of logical operators, including:

And	Logically ANDs multiple input arguments
Or	Logically ORs multiple input arguments
Not	Logical inverse of the input argument

Conditionals

Under the **Mathematics** heading, the **Lookup** category contains a selection of logic and error handling functions that include:

If	Returns one of two input arguments
Choose	Returns one on N input arguments
CountErr	Counts #ERRs over a range of cells

Decision Point Formula Examples

The following examples assume that cell **A1** contains a measurement. It could be a pixel count, a distance between two points, a blob feature, or some other type of measurement. Also, the examples assume that cells **B1** and **C1** contain the decision limits and that cell **D1** will store the decision.

1. **D1 = A1>B1**

In this example, we simply compare the contents of the cell **A1** to the contents of the cell **B1**. If the value of **A1** is greater than the value of **B1** the function returns the value **1**; otherwise it returns **0**.

2. **D1 = And(A1>B1,A1<C1)**

In this example we use the **And** function to compare **A1** to a range of values that is bounded by the contents of the cells **B1** and **C1**. If the value of **A1** is greater than the value of **B1**, and less than the value of **C1**, the function returns the value **1**; otherwise, the function returns **0**.

WEBER-0005206
DTX-0255_0086
Confidential
PR01925246
US1450_003692553

GS.book Page 79 Friday, April 14, 2000 4:04 PM

3. **D1 = If(And(A1>B1,A1<C1),"PASS","FAIL")**

In this example, the text strings **PASS** and **FAIL** are substituted for the values **0** and **1** of the previous example. If the value of **A1** is greater than the value of **B1** and less than the value of **C1**, the function returns the string **PASS**; otherwise, the function returns **FAIL**.

4. **D1 = If(A1<B1,"FAIL",If(A1<C1,"WARNING","PASS"))**

This example demonstrates how to nest the **If** function to determine more than two categories. If **A1** is less than **B1**, it returns **FAIL**; however, if **A1** is less than **C1**, it returns **WARNING**, and if **A1** is greater than or equal to **C1** it returns **PASS**.

5. **D1 = Choose((A1>=B1)+(A1>=C1),"FAIL","WARNING","PASS"))**

This example uses the **Choose** function to achieve the same result as in the previous example. If **A1** is less than both **B1** and **C1,** the first argument evaluates to **0** and **FAIL** is the output. If **A1** is greater than **B1** but less than **C1,** the first argument evaluates to **1** and **WARNING** is the output. If **A1** is greater than or equal to both **B1** and **C1**, the first argument evaluates to **2**, and **PASS** is the output.

6. **D1 = IF(CountErr(A1),0,A1>B1)**

This example compares the contents of the cell **A1** to the contents of cell **B1**. If the value of **A1** is greater than the value of **B1**, and **A1** is not **#ERR**, then the function returns **1**; otherwise it returns **0**.

Note	This is an important example because it shows how to filter errors from the output. Normally, a function that inputs #ERR also outputs #ERR. The CountErr function blocks the propagation of #ERR throughout the spreadsheet by returning a count of all #ERRs within its input range.

WEBER-0005207
DTX-0255_0087
Confidential
PR01925247
US1450_003692553

GS.book Page 80 Friday, April 14, 2000 4:04 PM

WEBER-0005208
DTX-0255_0088
Confidential
PR01925248
US1450_003692553

CHAPTER 12: COMMUNICATING RESULTS

This chapter shows you how to configure In-Sight's discrete and serial outputs to communicate inspection results to external devices.

Lesson: Using Discrete Output

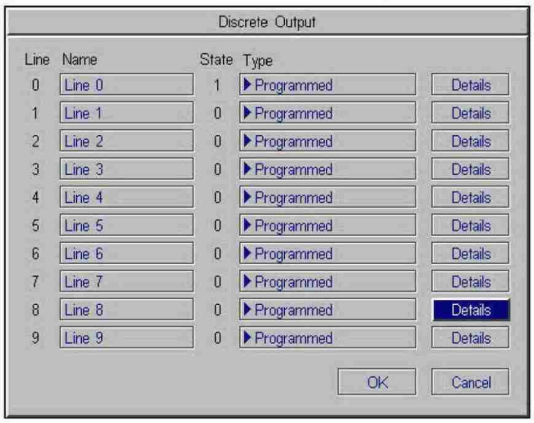
One of the easiest ways to communicate inspection results is to write discrete outputs. You can use the spreadsheet to assign the results of decisions, such as those discussed in the previous lesson, to individual points of discrete output.

Configuring Discrete Output Options

The In-Sight™ interface provides a dialog with several additional controls to configure discrete output.

To configure discrete output options:

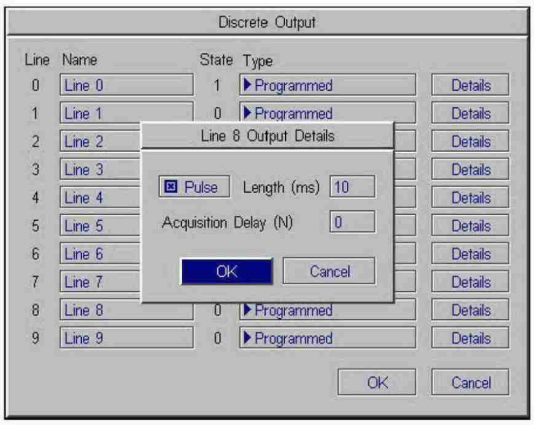
1. Press △ from anywhere in the spreadsheet to open the **System** menu.
2. Highlight **Settings** and press ⊗ to open the **Settings** menu.
3. Highlight **Discrete Output** and press ⊗ to open the **Discrete Output** dialog



Note	The default Type for all discrete output bits is Programmed. You can choose a different output configuration for a bit from the Type list box; however, if you configure a discrete output bit to another Type, that bit can no longer be controlled from the spreadsheet using the WriteDiscrete function.

WEBER-0005209
DTX-0255_0089
Confidential
PR01925249
US1450_003692553

4. Highlight **Details** and press ⊗ to open the **Discrete Output** dialog. This dialog contains three controls:



Pulse Enable for pulsed output (Delay Count must be 0); disable for steady state.

Length (ms) Duration of pulse.

Delay Count The number of acquisition or tracking pulses to delay the output.

Discrete Output port settings are system-wide and are automatically saved to nonvolatile memory when you press OK.

Writing Discrete Outputs

Use the **WriteDiscrete** function to connect individual cell values to the output lines.

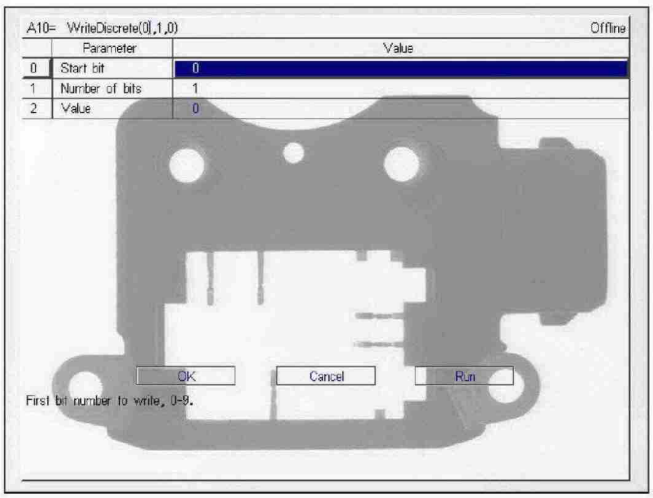
To write discrete output data from the spreadsheet:

1. Highlight an empty cell and press ⊗ to open the formula editor.

WEBER-0005210
DTX-0255_0090
Confidential
PR01925250
US1450_003692553

GS.book Page 83 Friday, April 14, 2000 4:04 PM

2. Highlight **WriteDiscrete** from the **Discrete** category under **Input/Output**; press ⊗ to open the **WriteDiscrete** property sheet.



3. Highlight the **Start bit** parameter and press ⊗ to enter the value of the discrete output bit to use. Highlight **Number of bits** and press ⊗ to enter the number of bits that will receive data.
4. Highlight **Value** and press ⊗ to enter *Cell Selection* mode. Highlight the cell containing a decision result integer to write to the selected bit(s).

Note	Discrete output bits will not change their state unless Online is checked in the System menu.

WEBER-0005211
DTX-0255_0091
Confidential
PR01925251
US1450_003692553

GS.book Page 84 Friday, April 14, 2000 4:04 PM

Lesson: Using Serial Output

Another way to communicate inspection results is to write them to one of the two serial ports. You can use **Stringf** under the **Text** function category to convert a value, or a set of values, into text messages that are written to the serial port.

Configuring Serial Output Options

The In-Sight™ interface provides a dialog with several additional controls to configure the serial output.

To configure serial output options:

1. Press △ from anywhere in the spreadsheet to open the System menu.
2. Highlight **Settings** and press ⊗ to open the **Settings** menu.
3. Highlight **Serial Port 1** and press ⊗ to open the **Serial Port 1** dialog.



Note	By default, Serial Port 0 is configured to support communication to the In-Sight™ PC Server utility. Serial Port 1 is configured to support programmed text output from the WriteSerial function. You can change defaults for both serial ports.

4. Highlight the **Mode** list box and select **Text**. Configure the remaining protocols in the **Serial Port 1** dialog as required.
5. Highlight **OK** and press ⊗ to confirm changes.

In-Sight™ automatically saves your serial port output settings to nonvolatile memory.

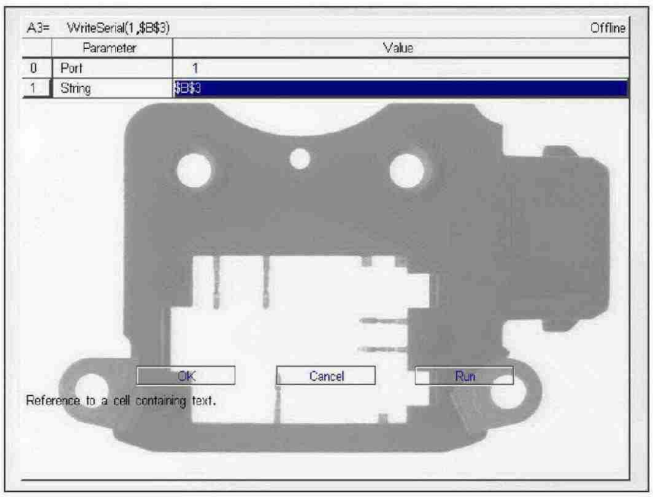
WEBER-0005212
DTX-0255_0092
Confidential
PR01925252
US1450_003692553

Writing Serial Outputs

The **WriteSerial** function transmits text strings.

To write serial output data from the spreadsheet:

1. Highlight an empty cell and press ⊗ to open the formula editor.
2. Highlight **WriteSerial**, located under the **Input/Output** heading and the **Serial** category, and press ⊗ to open the **WriteSerial** property sheet.



3. Use the property sheet to set the port number and to set a reference to the spreadsheet cell that contains the text string to be transmitted.

Note	Serial transmission initiated by WriteSerial is suppressed when the system is offline.

WEBER-0005213
DTX-0255_0093
Confidential
PR01925253
US1450_003692553

GS.book Page 86 Friday, April 14, 2000 4:04 PM

WEBER-0005214
DTX-0255_0094
Confidential
PR01925254
US1450_003692553

CHAPTER 13: MAKING A RUN-TIME INTERFACE

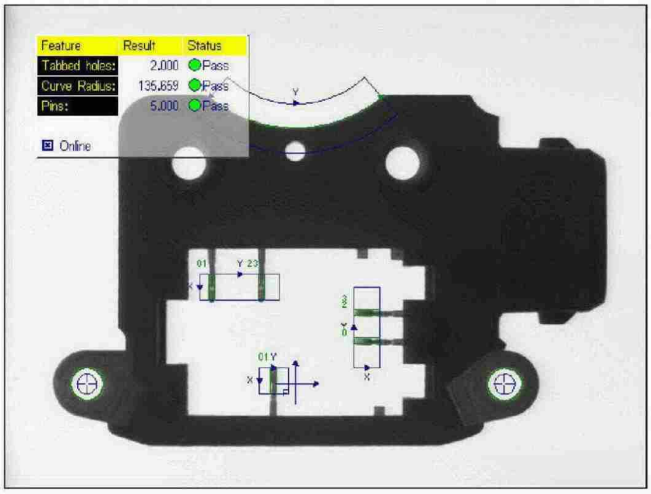
This chapter will show you how to construct a run-time user interface.

What is a Custom View?

The In-Sight™ spreadsheet is an excellent environment for configuring applications. You may wish, however, to present a more limited, application-specific interface to the operator on the production floor. Within In-Sight™, this tailor-made interface is called the Custom View.

The Custom View is essentially a dialog with user-definable features. While in the Custom View mode, access is restricted to the data displays and controls that you include in the view. Custom views can be password protected to prevent unauthorized changes.

To toggle between spreadsheet and custom views, simply press the **B** button. The following figure shows an example Custom View for a simple application.



WEBER-0005215
DTX-0255_0095
Confidential
PR01925255
US1450_003692553

GS.book Page 88 Friday, April 14, 2000 4:04 PM

The steps to creating a Custom View include:

Selecting the **Graphics** functions to include in the view and inserting them into a contiguous range of spreadsheet cells.

- Selecting the range of cells containing the **Graphics** functions and configuring the **Custom View** options.
- Setting the access level to use for this Custom View and defining passwords.

The following lessons define these steps in more detail.

Lesson: Understanding Graphics Functions

The first step to creating a Custom View is to select the data displays and graphic controls to include in the view. You include Graphics functions in the Custom View to simplify the run-time user interface and to limit the display of data to only the most important information.

Graphics functions for Custom View fall into two categories:

Controls	Interactive graphics for adjusting job parameters and settings
Displays	Graphics for displaying key status information

Note The **Graphics** function category includes an additional function subcategory called Image. These functions overlay points, lines, crosses, and text on top of the image, but they are independent of the Custom View mode.

Controls

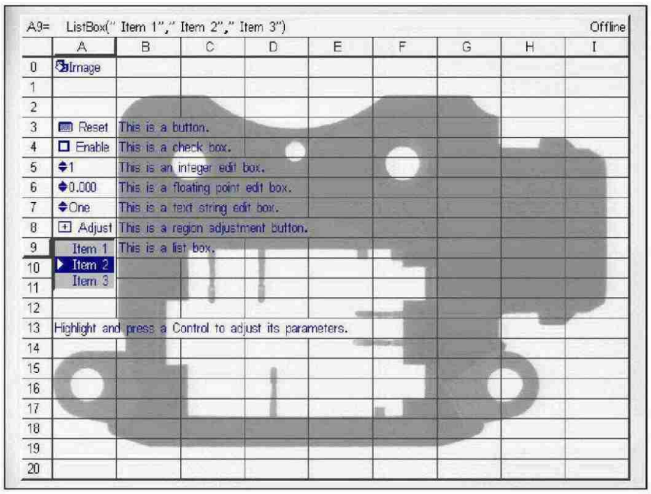
Graphical controls allow operators to adjust application parameters without entering *Spreadsheet View*. Control functions include:

Button	A push button control.
CheckBox	A check box control.
EditFloat	An edit box control for entering floating point values.
EditInt	An edit box control for entering integer values.
EditRegion	A button control for editing a region.

Note **EditRegion** cannot be referenced as a fixture by another function if that function is itself a reference to another function.

EditString	An edit box control for entering text strings.
ListBox	A list box control.

WEBER-0005216
DTX-0255_0096
Confidential
PR01925256
US1450_003692553



To insert a control into the spreadsheet:

1. Highlight an empty cell and press ⊗ to open the formula editor.
2. Highlight the desired function from the **Controls** subcategory under **Graphics** and press ⊗ to open the property sheet for the selected control.
3. Configure the control as necessary, and activate **OK** to insert the control into the spreadsheet.

In spreadsheet mode, you can "link" controls to generate events or to change the behavior or properties of other functions. For example, you can create a button that will reset the values accumulated in other cells by having those cells reference the button. When the button is pressed it returns a '1'; any cells with formulas that depend on the state of the button will be affected.

For a comprehensive review of controls, refer to the *In-Sight™ Guide and Reference* located in the CD-ROM.

WEBER-0005217
DTX-0255_0097
Confidential
PR01925257
US1450_003692553

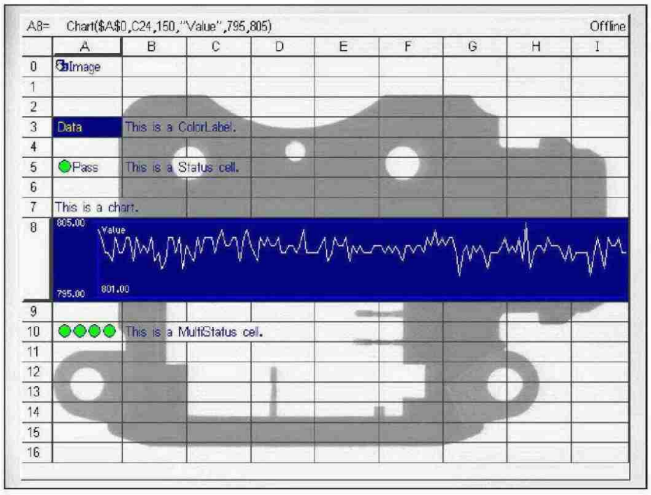
GS.book Page 90 Friday, April 14, 2000 4:04 PM

Displays

You will want to show only the most useful information in Custom View. For instance, if your application finds a blob and you would like the custom view to display one of the blob's row and column coordinates, simply include the cells containing the **GetRow** and **GetCol** function in the cells of the Custom View region.

In addition to the standard formulas, you can use special display functions to format data for visual verification. These **Graphics** functions are contained in the **Displays** subcategory and include:

Status	A status light that displays in red, yellow, or green depending on the current value.
ColorLabel	A text cell with programmable foreground and background colors.
Chart	A strip chart that can be inserted into the spreadsheet to graphically display the value of a cell over multiple events.
MultiStatus	Displays up to 16 status lights within a single cell, indicating bit values.



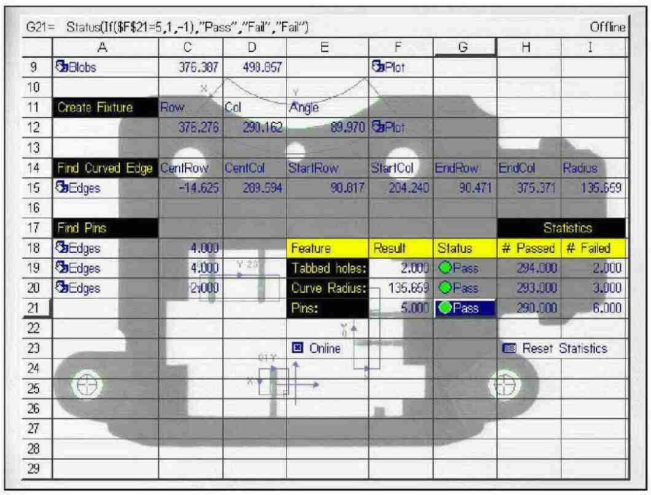
Lesson: Defining a Custom View

Making a Custom View dialog is a matter of creating an interface in an unused region of the spreadsheet and then defining that region to be the Custom View.

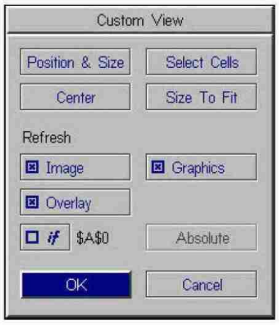
To configure the Custom View:

WEBER-0005218
DTX-0255_0098
Confidential
PR01925258
US1450_003692553

1. Choose an empty area of the spreadsheet to contain the displays and controls that will appear in the Custom View. Insert the desired **Displays** and **Controls** functions into this area of the spreadsheet, as described in the previous lesson.



2. Press Ⓞ from anywhere in the spreadsheet to open the **Edit** menu.
3. Highlight **Custom View** and press Ⓧ to open the Custom View dialog.



WEBER-0005219
DTX-0255_0099
Confidential
PR01925259
US1450_003692553

4. Configure the Custom View dialog using one or more of the following selections:

Position & Size Enters *Interactive Graphics* mode for repositioning and resizing the Custom View overlay.

Note	The size of the Custom View overlay is limited to the total size of the cell range chosen using Select Cells.

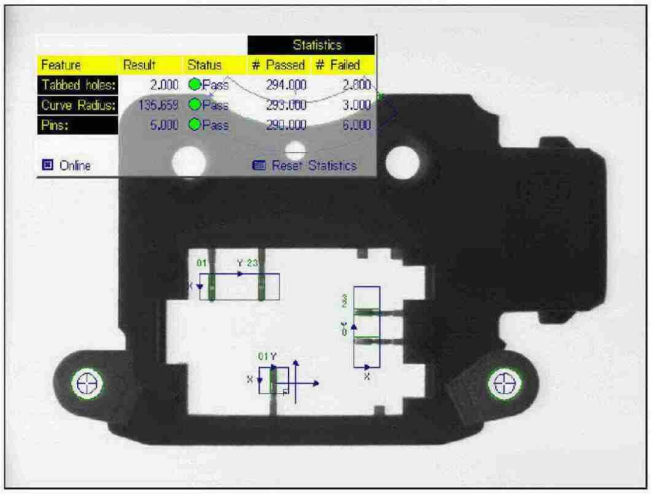
Select Cells Enters *Cell Selection* mode to select the cells to include in the Custom View.

Center Centers the Custom View overlay.

Size To Fit Sizes the Custom View overlay for the cell range chosen using Select Cells.

Refresh Enable/disables the Image display, the image Graphics, and the spreadsheet Overlay.

5. Highlight **OK** and press ⊗ to confirm changes.



To test your Custom View configuration, enter Custom View mode by pressing the **B** button. To return to the *Spreadsheet View*, press **B** again.

Lesson: Setting the Access Mode

The In-Sight™ interface can operate in one of three access modes:

Full Unsecured mode: Control pad navigation is active, and all cells, menus, dialogs, and view modes are accessible.

Protected Partially secured mode: Control pad navigation is active, but only interactive graphic controls can be adjusted. View mode can be *Custom* or *Spreadsheet*, but users cannot switch between the two views.

WEBER-0005220
DTX-0255_0100
Confidential
PR01925260
US1450_003692553

GS.book Page 93 Friday, April 14, 2000 4:04 PM

Locked Fully secured mode: Control pad navigation is inactive, and no cells, menus, dialogs, or view modes can be adjusted.

Note Pressing △ while in Locked Access mode will open the Access menu, where authorized users can change the Access mode to Full or Protected.

To set the Access mode:

1. Choose the view to run by pressing the **B** button to toggle between the *Spreadsheet View* and the *Custom View*.
2. Press △ to open the **System** menu. Select **Settings** and then **Passwords** to open the **Passwords** dialog.
3. From the dialog, enter passwords for **Full** and **Protected** access. These passwords are required whenever you change the access to a less secure mode. Press **OK** to accept the passwords. Press △ to return to the spreadsheet.



Note Record your passwords separately. If you forget the passwords you enter, there is no way to recover them and you will have to re-initialize all system settings.

4. Highlight **Access** on the **System** menu and press ⊗ to open the **Access** menu. Highlight the desired **Access** mode and press ⊗.



In-Sight™ is now running in the view mode you have chosen, using the **Access** mode you have selected.

WEBER-0005221
DTX-0255_0101
Confidential
PR01925261
US1450_003692553

GS.book Page 94 Friday, April 14, 2000 4:04 PM

WEBER-0005222
DTX-0255_0102
Confidential
PR01925262
US1450_003692553

GS.book Page 95 Friday, April 14, 2000 4:04 PM

WEBER-0005223
DTX-0255_0103
Confidential
PR01925263
US1450_003692553

GS.book Page 96 Friday, April 14, 2000 4:04 PM

WEBER-0005224
DTX-0255_0104
Confidential
PR01925264
US1450_003692553

CHAPTER 14: CONFIGURING FOR STARTUP

This chapter shows you how to configure your system to start a Job automatically on power-up.

Lesson: Selecting the Startup Job

By default, In-Sight starts a new Job on power-up. Once you've configured the Job for your application, however, you may want to install it as the default Job to load on power-up.

To load a Job automatically on power-up:

1. Press △ from anywhere in the spreadsheet to open the **System** menu.
2. Highlight **Settings** and press ⊗ to open the **Settings** menu.
3. Highlight **Startup** and press ⊗ to open the **Startup** dialog.



4. Select the job that you want to load automatically on power-up.
5. Configure the remaining **Startup** controls as required. These include:

Custom View Starts the selected Job using the Custom View instead of the default *Spreadsheet View*.

Access Starts the selected Job in one of three security modes: Full (default), Protected, or Locked. A password will be required to change modes when Access is Protected or Locked.

Note	When the system is **Locked** the spreadsheet Job and system configuration cannot be modified without a password. You set passwords in the Passwords dialog, accessed from the Settings menu.

Online Starts the selected Job in the Online instead of the default Offline mode.

Note	When the system is Offline, the spreadsheet program will not respond to any external events, including discrete and serial inputs.

WEBER-0005225
DTX-0255_0105
Confidential
PR01925265
US1450_003692553

GS.book Page 98 Friday, April 14, 2000 4:04 PM

6. Highlight **OK** and press ⊗ to confirm changes.

In-Sight™ automatically saves to nonvolatile memory all changes to the startup configuration. The next time power is applied to the In-Sight™ processor it will automatically load the selected job in the specified modes and states.

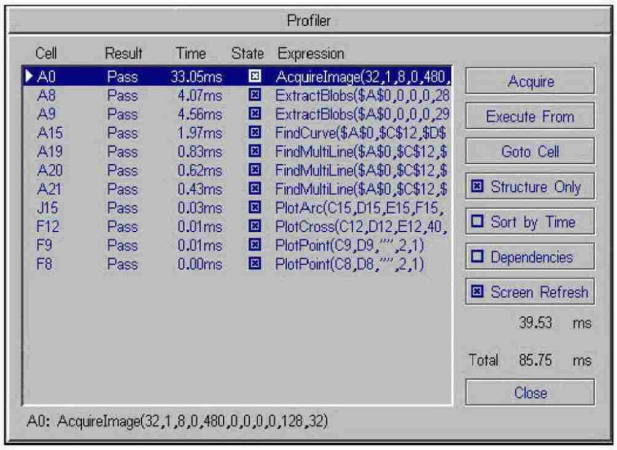
WEBER-0005226
DTX-0255_0106
Confidential
PR01925266
US1450_003692553

APPENDIX A: THE PROFILER

The In-Sight™ **Profiler** is an extremely useful utility that allows you to analyze the execution of every formula cell in the spreadsheet. With the **Profiler**, you can:

- View formulas according to their execution order in the spreadsheet.
- View formulas in the order of their dependencies on other spreadsheet cells.
- View formulas in order of their execution times.
- Compute the total elapsed time for a typical inspection cycle, including image acquisition.
- Compute the overhead of updating the display during an inspection cycle.

The **Profiler** is particularly useful for debugging and optimizing your application. For example, if the overall cycle time for an inspection is too long you can sort the formulas according their times of execution, and then focus on optimizing the formulas that take the longest to execute. If a formula in a cell is returning #ERR, you can quickly see the results from any referenced cells in order to isolate the problem.



To open the Profiler:

1. Press △ from anywhere in the spreadsheet to open the **System** menu.
2. Highlight **Profiler** and press ⊗ to open the **Profiler** menu.
3. Highlight **Execute From** and press ⊗ to profile the current job.

WEBER-0005227
DTX-0255_0107
Confidential
PR01925267
US1450_003692553

GS.book Page 100 Friday, April 14, 2000 4:04 PM

WEBER-0005228
DTX-0255_0108
Confidential
PR01925268
US1450_003692553

APPENDIX B: THE SERIAL INTERFACE

In-Sight's serial interface supports four modes of communication between the In-Sight processor and a remote device:

Server — As described in *Appendix C*, *Server* mode allows In-Sight to backup jobs, images, and system settings to a Windows computer running the In-Sight PC Server utility.

Text — *Text* mode sends/receives raw ASCII strings. You can effectively create your own command set by using the Text functions in the In-Sight interface to parsethe input string. Text strings are received into the spreadsheet using the **ReadSerial** function in the spreadsheet, and sent from the spreadsheet using the **WriteSerial** function.

Native — *Native* mode is a basic ASCII serial protocol for remotely controlling an In-Sight system from a non-Windows™ host device. The *Native Mode* serial protocol includes returns a response to the host for every command issued. No messages are initiated from the In-Sight processor.

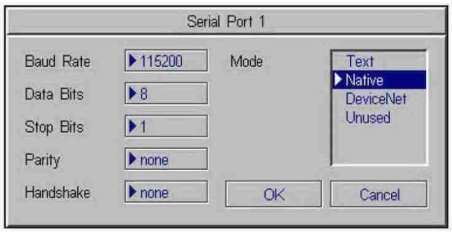
DeviceNet — Enables communications between an Allen-Bradley PLC and the In-Sight processor when the DeviceNet-to-RS232 Gateway adapter is used (Cognex P/N CIO-2550-00).

Note Refer to the *In-Sight™ Guide & Reference* contained on your In-Sight™ CD-ROM for more detailed information on the commands and functions availble for each serial mode.

Configuring the Serial Ports

To configure serial communications:

1. Press △ to open the **System** menu, and highlight **Settings**. Press ⊗ to open the **Settings** menu.
2. Highlight **Serial Port 0** or **Serial Port 1**. Press ⊗ to open the selected dialog.
3. Highlight the **Mode** list box and press ⊗. Then highlight the selected mode for serial communications and press ⊗.



WEBER-0005229
DTX-0255_0109
Confidential
PR01925269
US1450_003692553

GS.book Page 102 Friday, April 14, 2000 4:04 PM

WEBER-0005230
DTX-0255_0110
Confidential
PR01925270
US1450_003692553

APPENDIX C: THE PC SERVER

The In-Sight PC Server utility is a Windows based program that allows the vision processor to share data with a PC host.

When installed as described in the *Installing In-Sight* guide, the In-Sight server program allows you to backup and restore jobs, system settings, and images. The In-Sight PC Server also allows the In-Sight processor to receive firmware updates.

Backing Up and Restoring Jobs

Backing up a Job copies it from the In-Sight processor to a Windows PC over a serial connection; Restoring a Job copies it from the PC back to the In-Sight processor. Either operation can be initiated from the In-Sight interface, or from the PC Server utility on the PC.

Note	Backup & Restore operations are disabled when Access is set to Protected or Locked.

To backup a Job to a remote PC:

1. Configure the Serial 0 dialog in the In-Sight interface by selecting the Mode list box and pressing ⊗ to select Server mode.



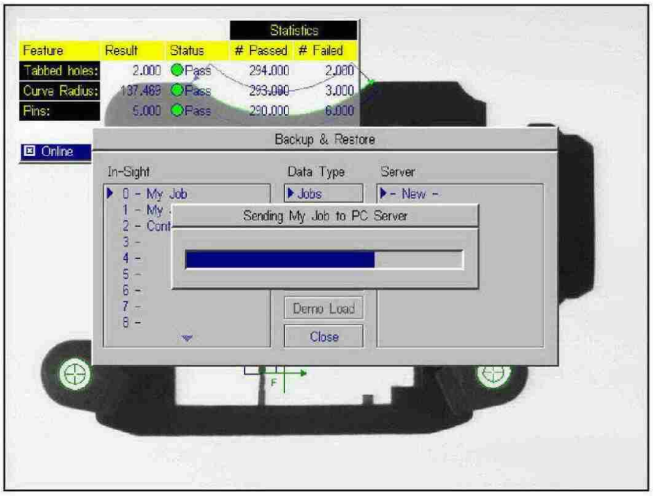
2. Start the In-Sight PC Server and verify the connection between the PC Server and the In-Sight processor.
3. Press △ from anywhere in the spreadsheet to open the System menu.

WEBER-0005231
DTX-0255_0111
Confidential
PR01925271
US1450_003692553

4. Highlight **Backup & Restore** and press ⊗ to open the **Backup & Restore** dialog.



5. Highlight a **Job** stored in the In-Sight processor and press ⊗ to select it.
6. Highlight the **Backup** button and press ⊗.



A progress bar will appear showing the status of the Job backup to the PC.

Note	To initiate a Job backup from the In-Sight PC Server, select Backup & Restore from the In-Sight menu in the Server software interface. During the backup, the progress bar will appear on the In-Sight display, though the Backup & Restore dialog will not open.

WEBER-0005232
DTX-0255_0112
Confidential
PR01925272
US1450_003692553

GS.book Page 105 Friday, April 14, 2000 4:04 PM

To restore a Job from a remote PC:

1. Highlight the Job on the PC that you want to restore to the processor. Press ⊗ to confirm your selection.
2. Highlight the memory slot in the In-Sight processor to which you want to restore the selected Job. Press ⊗ to confirm your selection.
3. Highlight **Restore** and press ⊗ to copy the job from the selected PC file to the specified vision processor memory slot.

Follow a similar procedure to backup and restore images and settings between the In-Sight processor and a remote PC.

WEBER-0005233
DTX-0255_0113
Confidential
PR01925273
US1450_003692553

GS.book Page 106 Friday, April 14, 2000 4:04 PM

WEBER-0005234
DTX-0255_0114
Confidential
PR01925274
US1450_003692553

APPENDIX D: REMOTE JOB SELECTION

In some applications, it is necessary to select a new Job by remote control. In-Sight™ makes this possible by allowing you to configure discrete input and output lines for the purpose of job selection.

Note	You must have physical I/O connections to the In-Sight™ processor as described in the Installing In-Sight guide before following the procedures described below.

Initiating a Job Load

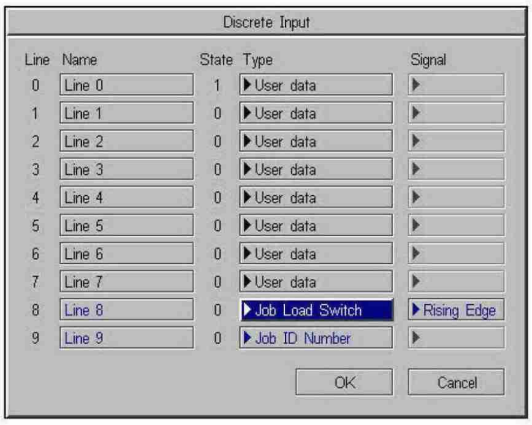
To configure the discrete input port to support Job selection:

1. Press △ from anywhere in the spreadsheet to open the **System** menu.
2. Highlight **Settings** and press ⊗ to open the **Settings** menu.
3. Highlight **Discrete Input** and press ⊗ to open the **Discrete Input** dialog.
4. Identify one unused bit to serve as the job load switch. Activate the bit's input type list box and highlight the item labeled **Job Load Switch**. Press ⊗ to confirm your selection.
5. Identify one or more unused bits to serve as the job identification number. Activate the associated input type list boxes and highlight the item labeled **Job ID Number.** Press ⊗ to confirm your selection.

Note	The Job identification number is constructed by combining the job ID number bits in order from least significant to most significant. For example: two job ID number bits will allow you to select a Job from memory slots 0 to 3; three Job ID number bits allow Job selection from slots 0 to 8; and four Job ID number bits allow Job selection from slots 0 to 16.

WEBER-0005235
DTX-0255_0115
Confidential
PR01925275
US1450_003692553

GS.book Page 108 Friday, April 14, 2000 4:04 PM



6. Highlight **OK** and press ⊗ to confirm changes.

The system will now be configured to load the job in the memory slot given by the job ID number bits whenever the bit associated with the job load switch is pulsed.

Responding to a Job Load

To configure the discrete output port to respond to a Job load:

1. Press △ from anywhere in the spreadsheet to open the **System** menu. Highlight **Settings** and press ⊗ to open the **Settings** menu.
2. Highlight **Discrete Output** and press ⊗ to open the **Discrete Output** dialog. Identify an unused bit to serve as the job load handshake.
3. Highlight the **Details** button for the selected output line and press ⊗ to open the **Discrete Output Details** dialog.

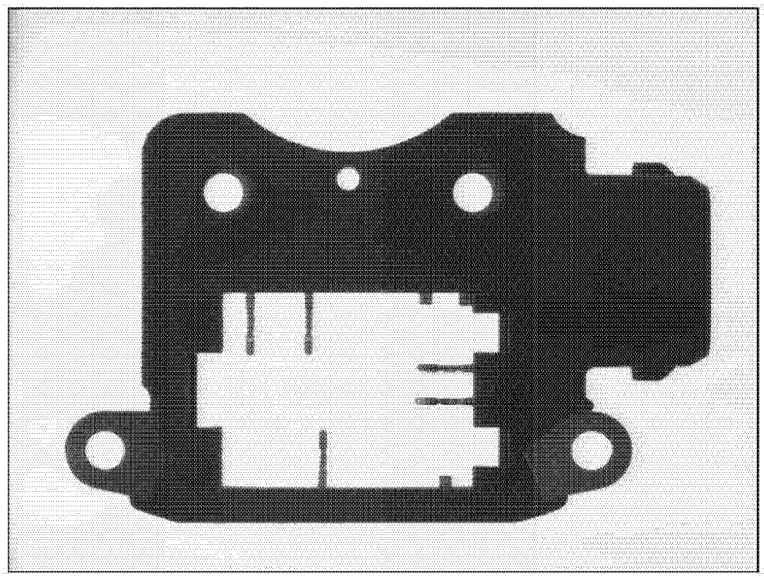
WEBER-0005236
DTX-0255_0116
Confidential
PR01925276
US1450_003692553

APPENDIX E: SAMPLE IMAGES

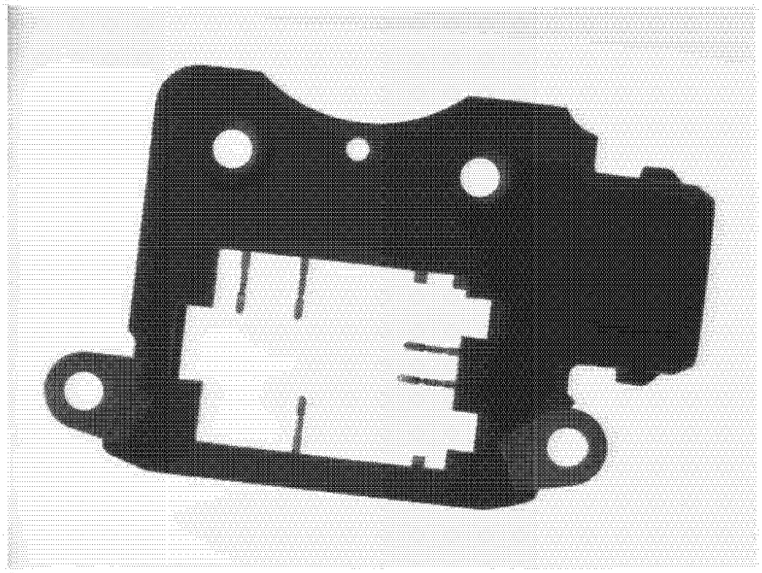
The images printed below are included to assist you in following along with the lessons in this book.

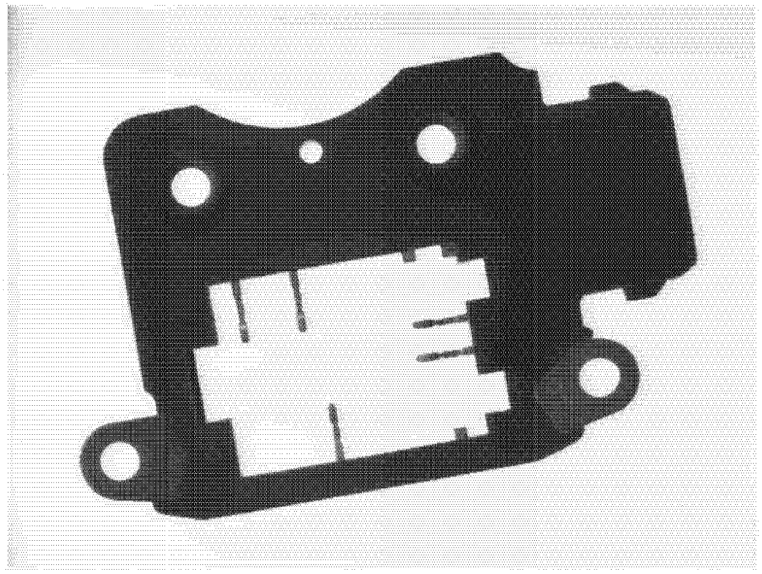
To use these images, setup the camera as described in *Chapter 1: Acquire a Working Image.* If you are using a 16-mm lens, the lens should be positioned approximately 10 to 11-inches (250 to 275 millimeters) above the page.

Note	These images are included on the CD-ROM that ships with each In-Sight system (image1_0.bmp, image1_1.bmp, and image1_2.bmp).



WEBER-0005237
DTX-0255_0117
Confidential
PR01925277
US1450_003692553





WEBER-0005238
DTX-0255_0118
Confidential
PR01925278
US1450_003692553