# FANUC Robot series

## R-30*i*B/R-30*i*B Mate CONTROLLER

## *i*RVision Inspection Application

## OPERATOR'S MANUAL

B-83304EN-3/02

**• Original Instructions**

Before using the Robot, be sure to read the "FANUC Robot Safety Manual (B-80687EN)" and understand the content.



The products in this manual are controlled based on Japan's "Foreign Exchange and Foreign Trade Law". The export from Japan may be subject to an export license by the government of Japan.
Further, re-export to another country may be subject to the license of the government of the country from where the product is re-exported. Furthermore, the product may also be controlled by re-export regulations of the United States government.
Should you wish to export or re-export these products, please contact FANUC for advice.

In this manual we have tried as much as possible to describe all the various matters.
However, we cannot describe all the matters which must not be done, or which cannot be done, because there are so many possibilities.
Therefore, matters which are not especially described as possible in this manual should be regarded as "impossible".

# SAFETY PRECAUTIONS

Thank you for purchasing FANUC Robot.
This chapter describes the precautions which must be observed to ensure the safe use of the robot.
Before attempting to use the robot, be sure to read this chapter thoroughly.

Before using the functions related to robot operation, read the relevant operator's manual to become familiar with those functions.

If any description in this chapter differs from that in the other part of this manual, the description given in this chapter shall take precedence.

For the safety of the operator and the system, follow all safety precautions when operating a robot and its peripheral devices installed in a work cell.
In addition, refer to the “FANUC Robot SAFETY HANDBOOK (B-80687EN)”.

# 1 WORKING PERSON

The personnel can be classified as follows.

Operator:
- Turns robot controller power ON/OFF
- Starts robot program from operator’s panel

Programmer or teaching operator:
- Operates the robot
- Teaches robot inside the safety fence

Maintenance engineer:
- Operates the robot
- Teaches robot inside the safety fence
- Maintenance (adjustment, replacement)

- An operator cannot work inside the safety fence.
- A programmer, teaching operator, and maintenance engineer can work inside the safety fence. The working activities inside the safety fence include lifting, setting, teaching, adjusting, maintenance, etc.
- To work inside the fence, the person must be trained on proper robot operation.

During the operation, programming, and maintenance of your robotic system, the programmer, teaching operator, and maintenance engineer should take additional care of their safety by using the following safety precautions.

- Use adequate clothing or uniforms during system operation
- Wear safety shoes
- Use helmet

# 2 DEFINITION OF WARNING, CAUTION AND NOTE

To ensure the safety of users and prevent damage to the machine, this manual indicates each precaution on safety with "Warning" or "Caution" according to its severity. Supplementary information is indicated by "Note". Read the contents of each "Warning", "Caution" and "Note" before attempting to use the robots.

> ⚠**WARNING**
> Applied when there is a danger of the user being injured or when there is a danger of both the user being injured and the equipment being damaged if the approved procedure is not observed.

> ⚠**CAUTION**
> Applied when there is a danger of the equipment being damaged, if the approved procedure is not observed.

> **NOTE**
> Notes are used to indicate supplementary information other than Warnings and Cautions.

- Read this manual carefully, and store it in a sales place.

# 3 WORKING PERSON SAFETY

Working person safety is the primary safety consideration. Because it is very dangerous to enter the operating space of the robot during automatic operation, adequate safety precautions must be observed.
The following lists the general safety precautions. Careful consideration must be made to ensure working person safety.

(1) Have the robot system working persons attend the training courses held by FANUC.

> FANUC provides various training courses. Contact our sales office for details.

(2) Even when the robot is stationary, it is possible that the robot is still in a ready to move state, and is waiting for a signal. In this state, the robot is regarded as still in motion. To ensure working person safety, provide the system with an alarm to indicate visually or aurally that the robot is in motion.
(3) Install a safety fence with a gate so that no working person can enter the work area without passing through the gate. Install an interlocking device, a safety plug, and so forth in the safety gate so that the robot is stopped as the safety gate is opened.

> The controller is designed to receive this interlocking signal of the door switch. When the gate is opened and this signal received, the controller stops the robot (Please refer to "STOP TYPE OF ROBOT" in SAFETY PRECAUTIONS for detail of stop type). For connection, see Fig.3 (a) and Fig.3 (b).

(4) Provide the peripheral devices with appropriate grounding (Class A, Class B, Class C, and Class D).

(5) Try to install the peripheral devices outside the work area.
(6) Draw an outline on the floor, clearly indicating the range of the robot motion, including the tools such as a hand.
(7) Install a mat switch or photoelectric switch on the floor with an interlock to a visual or aural alarm that stops the robot when a working person enters the work area.
(8) If necessary, install a safety lock so that no one except the working person in charge can turn on the power of the robot.

| |
|---|
| The circuit breaker installed in the controller is designed to disable anyone from turning it on when it is locked with a padlock. |

(9) When adjusting each peripheral device independently, be sure to turn off the power of the robot
(10) Operators should be ungloved while manipulating the operator’s panel or teach pendant. Operation with gloved fingers could cause an operation error.
(11) Programs, system variables, and other information can be saved on memory card or USB memories. Be sure to save the data periodically in case the data is lost in an accident.
(12) The robot should be transported and installed by accurately following the procedures recommended by FANUC. Wrong transportation or installation may cause the robot to fall, resulting in severe injury to workers.
(13) In the first operation of the robot after installation, the operation should be restricted to low speeds. Then, the speed should be gradually increased to check the operation of the robot.
(14) Before the robot is started, it should be checked that no one is in the area of the safety fence. At the same time, a check must be made to ensure that there is no risk of hazardous situations. If detected, such a situation should be eliminated before the operation.
(15) When the robot is used, the following precautions should be taken. Otherwise, the robot and peripheral equipment can be adversely affected, or workers can be severely injured.
- Avoid using the robot in a flammable environment.
- Avoid using the robot in an explosive environment.
- Avoid using the robot in an environment full of radiation.
- Avoid using the robot under water or at high humidity.
- Avoid using the robot to carry a person or animal.
- Avoid using the robot as a stepladder. (Never climb up on or hang from the robot.)
(16) When connecting the peripheral devices related to stop(safety fence etc.) and each signal (external emergency , fence etc.) of robot. be sure to confirm the stop movement and do not take the wrong connection.
(17) When preparing trestle, please consider security for installation and maintenance work in high place according to Fig.3 (c). Please consider footstep and safety bolt mounting position.

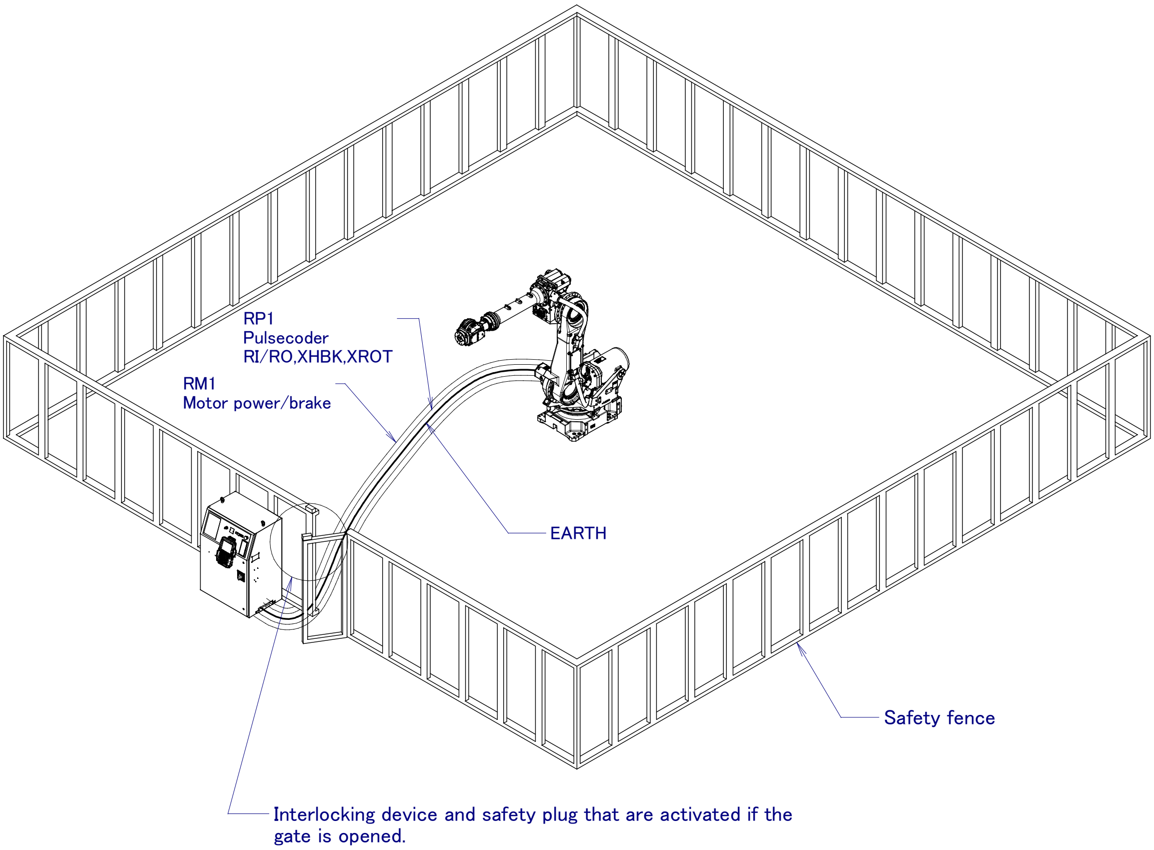



**Fig. 3 (a) Safety fence and safety gate**

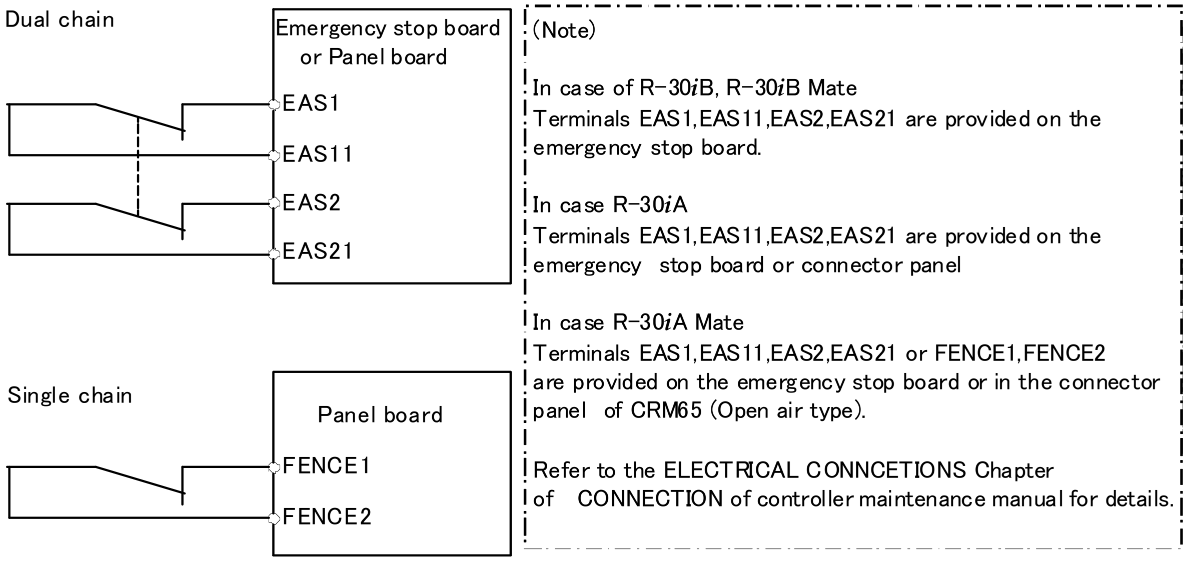



**Fig. 3 (b) Limit switch circuit diagram of the safety fence**

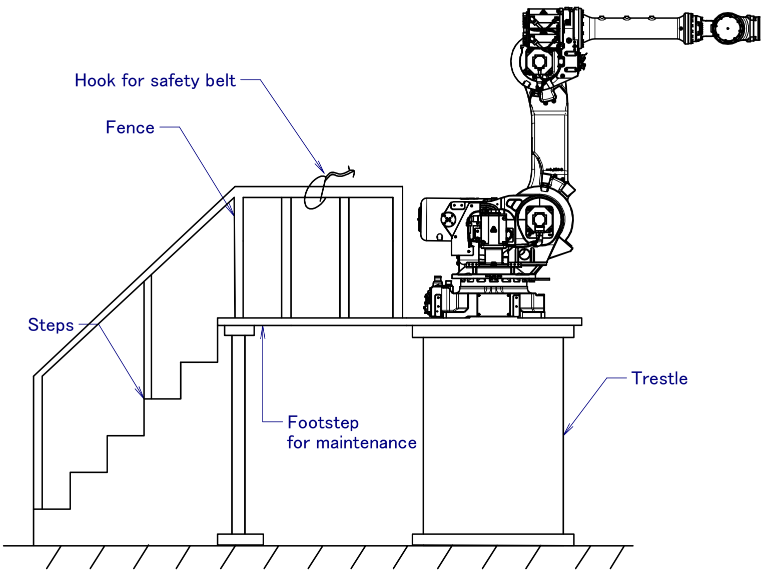



**Fig.3 (c) Footstep for maintenance**

# 3.1 OPERATOR SAFETY

The operator is a person who operates the robot system. In this sense, a worker who operates the teach pendant is also an operator. However, this section does not apply to teach pendant operators.

(1) If you do not have to operate the robot, turn off the power of the robot controller or press the EMERGENCY STOP button, and then proceed with necessary work.
(2) Operate the robot system at a location outside of the safety fence
(3) Install a safety fence with a safety gate to prevent any worker other than the operator from entering the work area unexpectedly and to prevent the worker from entering a dangerous area.
(4) Install an EMERGENCY STOP button within the operator's reach.

The robot controller is designed to be connected to an external EMERGENCY STOP button. With this connection, the controller stops the robot operation (Please refer to "STOP TYPE OF ROBOT" in SAFETY PRECAUTIONS for detail of stop type), when the external EMERGENCY STOP button is pressed. See the diagram below for connection.

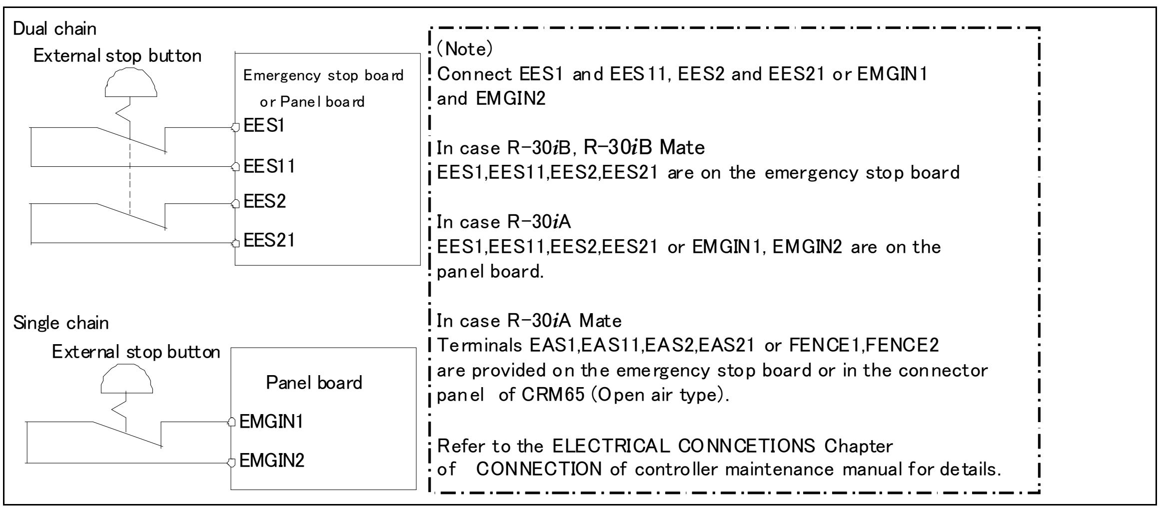



**Fig.3.1 Connection diagram for external emergency stop button**

# 3.2 SAFETY OF THE PROGRAMMER

While teaching the robot, the operator must enter the work area of the robot. The operator must ensure the safety of the teach pendant operator especially.

(1) Unless it is specifically necessary to enter the robot work area, carry out all tasks outside the area.
(2) Before teaching the robot, check that the robot and its peripheral devices are all in the normal operating condition.
(3) If it is inevitable to enter the robot work area to teach the robot, check the locations, settings, and other conditions of the safety devices (such as the EMERGENCY STOP button, the DEADMAN switch on the teach pendant) before entering the area.
(4) The programmer must be extremely careful not to let anyone else enter the robot work area.
(5) Programming should be done outside the area of the safety fence as far as possible. If programming needs to be done in the area of the safety fence, the programmer should take the following precautions:
   - Before entering the area of the safety fence, ensure that there is no risk of dangerous situations in the area.
   - Be prepared to press the emergency stop button whenever necessary.
   - Robot motions should be made at low speeds.
   - Before starting programming, check the entire system status to ensure that no remote instruction to the peripheral equipment or motion would be dangerous to the user.

Our operator panel is provided with an emergency stop button and a key switch (mode switch) for selecting the automatic operation mode (AUTO) and the teach modes (T1 and T2). Before entering the inside of the safety fence for the purpose of teaching, set the switch to a teach mode, remove the key from the mode switch to prevent other people from changing the operation mode carelessly, then open the safety gate. If the safety gate is opened with the automatic operation mode set, the robot stops (Please refer to "STOP TYPE OF ROBOT" in SAFETY PRECAUTIONS for detail of stop type). After the switch is set to a teach mode, the safety gate is disabled. The programmer should understand that the safety gate is disabled and is responsible for keeping other people from entering the inside of the safety fence. (In case of R-30*i*A Mate Controller standard specification, there is no mode switch. The automatic operation mode and the teach mode is selected by teach pendant enable switch.)

Our teach pendant is provided with a DEADMAN switch as well as an emergency stop button. These button and switch function as follows:
(1) Emergency stop button: Causes an emergency stop (Please refer to "STOP TYPE OF ROBOT" in SAFETY PRECAUTIONS for detail of stop type) when pressed.
(2) DEADMAN switch: Functions differently depending on the teach pendant enable/disable switch setting status.
   (a) Disable: The DEADMAN switch is disabled.
   (b) Enable: Servo power is turned off when the operator releases the DEADMAN switch or when the operator presses the switch strongly.
   Note) The DEADMAN switch is provided to stop the robot when the operator releases the teach pendant or presses the pendant strongly in case of emergency. The R-30*i*B/R-30*i*B Mate/R-30*i*A/ R-30*i*A Mate employs a 3-position DEADMAN switch, which allows the robot to operate when the 3-position DEADMAN switch is pressed to its intermediate point. When the operator releases the DEADMAN switch or presses the switch strongly, the robot stops immediately.

The operator's intention of starting teaching is determined by the controller through the dual operation of setting the teach pendant enable/disable switch to the enable position and pressing the DEADMAN switch. The operator should make sure that the robot could operate in such conditions and be responsible in carrying out tasks safely.

Based on the risk assessment by FANUC, number of operation of DEADMAN SW should not exceed about 10000 times per year.

The teach pendant, operator panel, and peripheral device interface send each robot start signal. However the validity of each signal changes as follows depending on the mode switch and the DEADMAN switch of the operator panel, the teach pendant enable switch and the remote condition on the software.

**In case of R-30*i*B/R-30*i*B Mate/R-30*i*A Controller or CE or RIA specification of R-30*i*A Mate Controller**

| Mode | Teach pendant enable switch | Software remote condition | Teach pendant | Operator panel | Peripheral device |
|---|---|---|---|---|---|
| AUTO mode | On | Local | Not allowed | Not allowed | Not allowed |
| | | Remote | Not allowed | Not allowed | Not allowed |
| | Off | Local | Not allowed | Allowed to start | Not allowed |
| | | Remote | Not allowed | Not allowed | Allowed to start |
| T1, T2 mode | On | Local | Allowed to start | Not allowed | Not allowed |
| | | Remote | Allowed to start | Not allowed | Not allowed |
| | Off | Local | Not allowed | Not allowed | Not allowed |
| | | Remote | Not allowed | Not allowed | Not allowed |

**T1,T2 mode: DEADMAN switch is effective.**

**In case of standard specification of R-30*i*A Mate Controller**

| Teach pendant enable switch | Software remote condition | Teach pendant | Peripheral device |
|---|---|---|---|
| On | Ignored | Allowed to start | Not allowed |
| Off | Local | Not allowed | Not allowed |
| | Remote | Not allowed | Allowed to start |

(6) (Only when R-30*i*B/R-30*i*B Mate /R-30*i*A Controller or CE or RIA specification of R-30*i*A Mate controller is selected.) To start the system using the operator's panel, make certain that nobody is the robot work area and that there are no abnormal conditions in the robot work area.

(7) When a program is completed, be sure to carry out a test operation according to the procedure below.
   (a) Run the program for at least one operation cycle in the single step mode at low speed.
   (b) Run the program for at least one operation cycle in the continuous operation mode at low speed.
   (c) Run the program for one operation cycle in the continuous operation mode at the intermediate speed and check that no abnormalities occur due to a delay in timing.
   (d) Run the program for one operation cycle in the continuous operation mode at the normal operating speed and check that the system operates automatically without trouble.
   (e) After checking the completeness of the program through the test operation above, execute it in the automatic operation mode.

(8) While operating the system in the automatic operation mode, the teach pendant operator should leave the robot work area.

## 3.3 SAFETY OF THE MAINTENANCE ENGINEER

For the safety of maintenance engineer personnel, pay utmost attention to the following.

(1) During operation, never enter the robot work area.

(2) A hazardous situation may arise when the robot or the system, are kept with their power-on during maintenance operations. Therefore, for any maintenance operation, the robot and the system should be put into the power-off state. If necessary, a lock should be in place in order to prevent any other person from turning on the robot and/or the system. In case maintenance needs to be executed in the power-on state, the emergency stop button must be pressed.

(3) If it becomes necessary to enter the robot operation range while the power is on, press the emergency stop button on the operator panel, or the teach pendant before entering the range. The

maintenance personnel must indicate that maintenance work is in progress and be careful not to allow other people to operate the robot carelessly.

(4) When entering the area enclosed by the safety fence, the maintenance worker must check the entire system in order to make sure no dangerous situations exist. In case the worker needs to enter the safety area whilst a dangerous situation exists, extreme care must be taken, and entire system status must be carefully monitored.

(5) Before the maintenance of the pneumatic system is started, the supply pressure should be shut off and the pressure in the piping should be reduced to zero.

(6) Before the start of teaching, check that the robot and its peripheral devices are all in the normal operating condition.

(7) Do not operate the robot in the automatic mode while anybody is in the robot work area.

(8) When you maintain the robot alongside a wall or instrument, or when multiple workers are working nearby, make certain that their escape path is not obstructed.

(9) When a tool is mounted on the robot, or when any moving device other than the robot is installed, such as belt conveyor, pay careful attention to its motion.

(10) If necessary, have a worker who is familiar with the robot system stand beside the operator panel and observe the work being performed. If any danger arises, the worker should be ready to press the EMERGENCY STOP button at any time.

(11) When replacing a part, please contact FANUC service center. If a wrong procedure is followed, an accident may occur, causing damage to the robot and injury to the worker.

(12) When replacing or reinstalling components, take care to prevent foreign material from entering the system.

(13) When handling each unit or printed circuit board in the controller during inspection, turn off the circuit breaker to protect against electric shock.
If there are two cabinets, turn off the both circuit breaker.

(14) A part should be replaced with a part recommended by FANUC. If other parts are used, malfunction or damage would occur. Especially, a fuse that is not recommended by FANUC should not be used. Such a fuse may cause a fire.

(15) When restarting the robot system after completing maintenance work, make sure in advance that there is no person in the work area and that the robot and the peripheral devices are not abnormal.

(16) When a motor or brake is removed, the robot arm should be supported with a crane or other equipment beforehand so that the arm would not fall during the removal.

(17) Whenever grease is spilled on the floor, it should be removed as quickly as possible to prevent dangerous falls.

(18) The following parts are heated. If a maintenance worker needs to touch such a part in the heated state, the worker should wear heat-resistant gloves or use other protective tools.

- Servo motor
- Inside the controller
- Reducer
- Gearbox
- Wrist unit

(19) Maintenance should be done under suitable light. Care must be taken that the light would not cause any danger.

(20) When a motor, reducer, or other heavy load is handled, a crane or other equipment should be used to protect maintenance workers from excessive load. Otherwise, the maintenance workers would be severely injured.

(21) The robot should not be stepped on or climbed up during maintenance. If it is attempted, the robot would be adversely affected. In addition, a misstep can cause injury to the worker.

(22)When performing maintenance work in high place, secure a footstep and wear safety belt.

(23) After the maintenance is completed, spilled oil or water and metal chips should be removed from the floor around the robot and within the safety fence.

(24) When a part is replaced, all bolts and other related components should put back into their original places. A careful check must be given to ensure that no components are missing or left not mounted.

(25) In case robot motion is required during maintenance, the following precautions should be taken :

- Foresee an escape route. And during the maintenance motion itself, monitor continuously the whole system so that your escape route will not become blocked by the robot, or by peripheral equipment.
- Always pay attention to potentially dangerous situations, and be prepared to press the emergency stop button whenever necessary.

(26) The robot should be periodically inspected. (Refer to the robot mechanical manual and controller maintenance manual.) A failure to do the periodical inspection can adversely affect the performance or service life of the robot and may cause an accident

(27) After a part is replaced, a test operation should be given for the robot according to a predetermined method. (See TESTING section of “Controller operator’s manual”.) During the test operation, the maintenance staff should work outside the safety fence.

# 4 SAFETY OF THE TOOLS AND PERIPHERAL DEVICES

## 4.1 PRECAUTIONS IN PROGRAMMING

(1) Use a limit switch or other sensor to detect a dangerous condition and, if necessary, design the program to stop the robot when the sensor signal is received.

(2) Design the program to stop the robot when an abnormal condition occurs in any other robots or peripheral devices, even though the robot itself is normal.

(3) For a system in which the robot and its peripheral devices are in synchronous motion, particular care must be taken in programming so that they do not interfere with each other.

(4) Provide a suitable interface between the robot and its peripheral devices so that the robot can detect the states of all devices in the system and can be stopped according to the states.

## 4.2 PRECAUTIONS FOR MECHANISM

(1) Keep the component cells of the robot system clean, and operate the robot in an environment free of grease, water, and dust.

(2) Don’t use unconfirmed liquid for cutting fluid and cleaning fluid.

(3) Employ a limit switch or mechanical stopper to limit the robot motion so that the robot or cable does not strike against its peripheral devices or tools.

(4) Observe the following precautions about the mechanical unit cables. Failure to follow precautions may cause mechanical troubles.
   - Use mechanical unit cable that have required user interface.
   - Don’t add user cable or hose to inside of mechanical unit.
   - Please do not obstruct the movement of the mechanical unit cable when cables are added to outside of mechanical unit.
   - In the case of the model that a cable is exposed, Please do not perform remodeling (Adding a protective cover and fix an outside cable more) obstructing the behavior of the outcrop of the cable.
   - When installing user peripheral equipment on the robot mechanical unit, please pay attention that equipment does not interfere with the robot itself.

(5) The frequent power-off stop for the robot during operation causes the trouble of the robot. Please avoid the system construction that power-off stop would be operated routinely. (Refer to bad case example.) Please execute power-off stop after reducing the speed of the robot and stopping it by hold stop or cycle stop when it is not urgent. (Please refer to "STOP TYPE OF ROBOT" in SAFETY PRECAUTIONS for detail of stop type.)
(Bad case example)

- Whenever poor product is generated, a line stops by emergency stop.
- When alteration was necessary, safety switch is operated by opening safety fence and power-off stop is executed for the robot during operation.
- An operator pushes the emergency stop button frequently, and a line stops.
- An area sensor or a mat switch connected to safety signal operate routinely and power-off stop is executed for the robot.

(6) Robot stops urgently when collision detection alarm (SRVO-050) etc. occurs. Please try to avoid unnecessary power-off stops. It may cause the trouble of the robot, too. So remove the causes of the alarm.

# 5 SAFETY OF THE ROBOT MECHANISM

## 5.1 PRECAUTIONS IN OPERATION

(1) When operating the robot in the jog mode, set it at an appropriate speed so that the operator can manage the robot in any eventuality.
(2) Before pressing the jog key, be sure you know in advance what motion the robot will perform in the jog mode.

## 5.2 PRECAUTIONS IN PROGRAMMING

(1) When the work areas of robots overlap, make certain that the motions of the robots do not interfere with each other.
(2) Be sure to specify the predetermined work origin in a motion program for the robot and program the motion so that it starts from the origin and terminates at the origin.
Make it possible for the operator to easily distinguish at a glance that the robot motion has terminated.

## 5.3 PRECAUTIONS FOR MECHANISMS

(1) Keep the work areas of the robot clean, and operate the robot in an environment free of grease, water, and dust.

## 5.4 PROCEDURE TO MOVE ARM WITHOUT DRIVE POWER IN EMERGENCY OR ABNORMAL SITUATIONS

For emergency or abnormal situations (e.g. persons trapped in or by the robot), brake release unit can be used to move the robot axes without drive power.

Please refer to controller maintenance manual and mechanical unit operator’s manual for using method of brake release unit and method of supporting robot.

# 6 SAFETY OF THE END EFFECTOR

## 6.1 PRECAUTIONS IN PROGRAMMING

(1) To control the pneumatic, hydraulic and electric actuators, carefully consider the necessary time delay after issuing each control command up to actual motion and ensure safe control.
(2) Provide the end effector with a limit switch, and control the robot system by monitoring the state of the end effector.

# 7 STOP TYPE OF ROBOT

The following three robot stop types exist:

### Power-Off Stop (Category 0 following IEC 60204-1)

Servo power is turned off and the robot stops immediately. Servo power is turned off when the robot is moving, and the motion path of the deceleration is uncontrolled.
The following processing is performed at Power-Off stop.

- An alarm is generated and servo power is turned off.
- The robot operation is stopped immediately. Execution of the program is paused.

### Controlled stop (Category 1 following IEC 60204-1)

The robot is decelerated until it stops, and servo power is turned off.
The following processing is performed at Controlled stop.

- The alarm "SRVO-199 Controlled stop" occurs along with a decelerated stop. Execution of the program is paused.
- An alarm is generated and servo power is turned off.

### Hold (Category 2 following IEC 60204-1)

The robot is decelerated until it stops, and servo power remains on.
The following processing is performed at Hold.

- The robot operation is decelerated until it stops. Execution of the program is paused.

> ⚠ **WARNING**
> The stopping distance and stopping time of Controlled stop are longer than the stopping distance and stopping time of Power-Off stop. A risk assessment for the whole robot system, which takes into consideration the increased stopping distance and stopping time, is necessary when Controlled stop is used.

When the emergency stop button is pressed or the FENCE is open, the stop type of robot is Power-Off stop or Controlled stop. The configuration of stop type for each situation is called *stop pattern*. The stop pattern is different according to the controller type or option configuration.

There are the following 3 Stop patterns.

| Stop pattern | Mode | Emergency stop button | External Emergency stop | FENCE open | SVOFF input | Servo disconnect |
|---|---|---|---|---|---|---|
| A | AUTO | P-Stop | P-Stop | C-Stop | C-Stop | P-Stop |
| | T1 | P-Stop | P-Stop | - | C-Stop | P-Stop |
| | T2 | P-Stop | P-Stop | - | C-Stop | P-Stop |
| B | AUTO | P-Stop | P-Stop | P-Stop | P-Stop | P-Stop |
| | T1 | P-Stop | P-Stop | - | P-Stop | P-Stop |
| | T2 | P-Stop | P-Stop | - | P-Stop | P-Stop |
| C | AUTO | C-Stop | C-Stop | C-Stop | C-Stop | C-Stop |
| | T1 | P-Stop | P-Stop | - | C-Stop | P-Stop |
| | T2 | P-Stop | P-Stop | - | C-Stop | P-Stop |

P-Stop: Power-Off stop
C-Stop: Controlled stop
-: Disable

The following table indicates the Stop pattern according to the controller type or option configuration.

| Option | R-30*i*B/R-30*i*B Mate |
|---|---|
| Standard | A (*) |
| Controlled stop by E-Stop (A05B-2600-J570) | C (*) |

(*) R-30*i*B/R-30*i*B Mate does not have servo disconnect. R-30*i*B Mate does not have SVOFF input.

| Option | R-30*i*A | | | | R-30*i*A Mate | | |
|---|---|---|---|---|---|---|---|
| | Standard (Single) | Standard (Dual) | RIA type | CE type | Standard | RIA type | CE type |
| Standard | B (*) | A | A | A | A (**) | A | A |
| Stop type set (Stop pattern C) (A05B-2500-J570) | N/A | N/A | C | C | N/A | C | C |

(*) R-30*i*A standard (single) does not have servo disconnect.
(**) R-30*i*A Mate Standard does not have servo disconnect, and the stop type of SVOFF input is Power-Off stop.

The stop pattern of the controller is displayed in "Stop pattern" line in software version screen. Please refer to "Software version" in operator's manual of controller for the detail of software version screen.

## "Controlled stop by E-Stop" option

When "Controlled stop by E-Stop" (A05B-2600-J570) option (In case of R-30*i*A/R-30*i*A Mate, it is Stop type set (Stop pattern C) (A05B-2500-J570)) is specified, the stop type of the following alarms becomes Controlled stop but only in AUTO mode. In T1 or T2 mode, the stop type is Power-Off stop which is the normal operation of the system.

| Alarm | Condition |
|---|---|
| SRVO-001 Operator panel E-stop | Operator panel emergency stop is pressed. |
| SRVO-002 Teach pendant E-stop | Teach pendant emergency stop is pressed. |
| SRVO-007 External emergency stops | External emergency stop input (EES1-EES11, EES2-EES21) is open. (R-30*i*A/R-30*i*B/R-30*i*B Mate controller) |
| SRVO-194 Servo disconnect | Servo disconnect input (SD4-SD41, SD5-SD51) is open. (R-30*i*A controller) |
| SRVO-218 Ext.E-stop/Servo Disconnect | External emergency stop input (EES1-EES11, EES2-EES21) is open. (R-30*i*A Mate controller) |
| SRVO-408 DCS SSO Ext Emergency Stop | In DCS Safe I/O connect function, SSO[3] is OFF. |
| SRVO-409 DCS SSO Servo Disconnect | In DCS Safe I/O connect function, SSO[4] is OFF. |

Controlled stop is different from Power-Off stop as follows:

- In Controlled stop, the robot is stopped on the program path. This function is effective for a system where the robot can interfere with other devices if it deviates from the program path.
- In Controlled stop, physical impact is less than Power-Off stop. This function is effective for systems where the physical impact to the mechanical unit or EOAT (End Of Arm Tool) should be minimized.
- The stopping distance and stopping time of Controlled stop is longer than the stopping distance and stopping time of Power-Off stop, depending on the robot model and axis. Please refer to the operator's manual of a particular robot model for the data of stopping distance and stopping time.

In case of R-30*i*A or R-30*i*A Mate, this function is available only in CE or RIA type hardware.

When this option is loaded, this function cannot be disabled.

The stop type of DCS Position and Speed Check functions is not affected by the loading of this option.

> ⚠ **WARNING**
> The stopping distance and stopping time of Controlled stop are longer than the stopping distance and stopping time of Power-Off stop. A risk assessment for the whole robot system, which takes into consideration the increased stopping distance and stopping time, is necessary when this option is loaded.

# TABLE OF CONTENTS

# 1 PREFACE

This chapter describes an overview of this manual, which should be read before operating the *i*RVision Inspection function.

## 1.1 OVERVIEW OF THE MANUAL

### 1.1.1 About This Manual

This manual describes how to operate *i*RVision controlled by the R-30*i*B/ R-30*i*B Mate controller.
In this manual, only the operation and the technique of programming for the dedicated sensor functions are explained, assuming that the installation and the setup of the robot are completed. Refer to the "HANDLING TOOL Operations Manual" about other operations of FANUC Robots.

> ⚠ **CAUTION**
> This manual is based on R-30*i*B/ R-30*i*B Mate system software of V8.20P/01. Note that the functions and settings not described in this manual may be available, and some notation differences are present, depending on the software version.

### 1.1.2 Organization of This Manual

| | |
|---|---|
| **Chapter 1** | Preface |
| **Chapter 2** | Overview |
| **Chapter 3** | Study for application |
| **Chapter 4** | Setup |
| **Chapter 5** | Sample applications |

### 1.1.3 Related Manuals

This section introduces related manual for *i*RVision.

**R-30*i*B/ R-30*i*B Mate CONTROLLER OPERATOR’S MANUAL (Basic Operation) B-83284EN**

This is the main manual of R-30*i*B/ R-30*i*B Mate Controller. This manual describes the following items for manipulating workpieces with the robot:

- Setting the system for manipulating workpieces
- Operating the robot
- Creating and changing a program
- Executing a program
- Status indications
- Backup and restore robot programs.

This manual is used on an applicable design, robot installation, robot teaching.

**R-30*i*B CONTROLLER MAINTENANCE MANUAL B-83195EN**

This manual describes the maintenance and connection of R-30*i*B Controller.

**R-30*i*B Mate CONTROLLER MAINTENANCE MANUAL B-83525EN**

This manual describes the maintenance and connection of R-30*i*B Mate Controller.

### R-30*i*B Mate CONTROLLER (Open Air) MAINTENANCE MANUAL B-83555EN

This manual describes the maintenance and connection of R-30*i*B Mate Controller (Open Air).

### R-30*i*B/ R-30*i*B Mate CONTROLLER OPERATOR'S MANUAL (Alarm Code List) B-83284EN-1

This manual describes the error code listings, causes, and remedies of R-30*i*B/ R-30*i*B Mate Controller.

### R-30*i*B CONTROLLER Sensor Mechanical / Control unit OPERATOR'S MANUAL B-83434EN

This manual describes the connection between sensors which is a camera or 3D Laser Sensor and R-30*i*B Controller, and maintenance of sensors.

### R-30*i*B/ R-30*i*B Mate CONTROLLER *i*RVision OPERATOR'S MANUAL (Reference) B-83304EN

This manual is the reference manual for *i*RVision on the R-30*i*B/ R-30*i*B Mate controller. This manual describes each
function which is provided by *i*RVision. When you would like to know the meanings (e.g. the items about the *i*RVision setup screen, the arguments of the instruction, and so on), please refer to this manual.

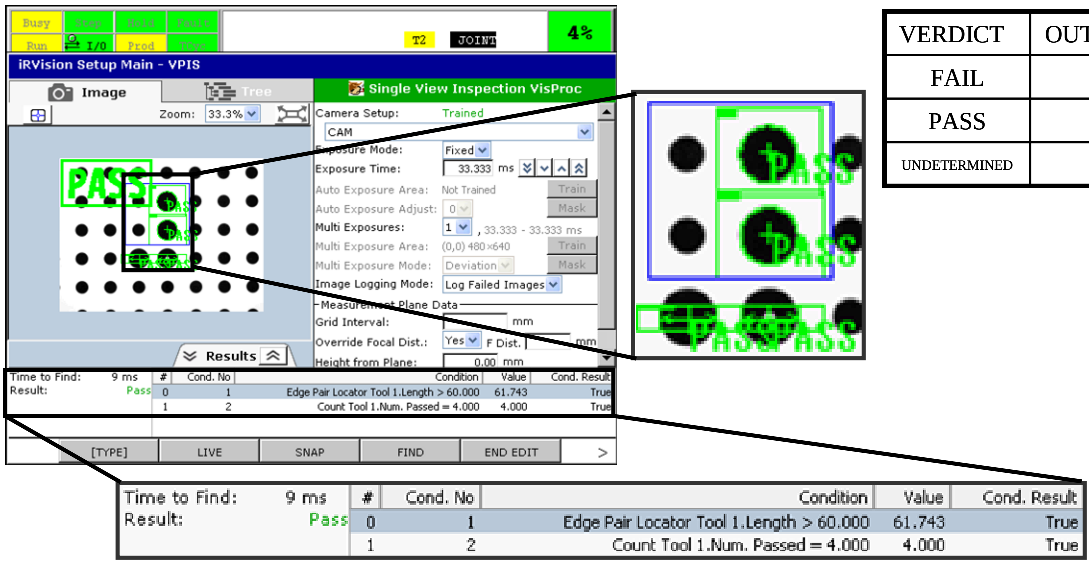
# 2 OVERVIEW

This chapter gives an overview of *i*RVision Inspection.

## 2.1 FEATURES

*i*RVision Inspection is a function which snaps an image of a target to be inspected, processes the image, and evaluates whether the target passes or fails inspection based on specified conditions. This function automates visual appearance inspections that have conventionally been performed manually, an example of which includes a post-assembly check for necessary parts. Even features that look the same to the human eye, differences become apparent when the features are quantified using image processing. Using *i*RVision for an inspection enables the detection of defects that may be missed by an operator during a visual appearance inspection, and can minimize defective products.

The Inspection function extracts various features of a target including brightness, position, length, number of parts, and area from an image, evaluates whether each feature meets specified conditions, and finally determines whether the target passes or fails inspection by logically combining the evaluation results. You can set various logical conditions according to your purposes. An inspection result is stored as a numeric value in a specified register in the robot controller. When an inspection result of a target is determined as failed, a value of 0 is stored in the register; when it is determined as passed, a value of 1 is stored. If the result is inconclusive because certain image inspections could not be performed, for example, a value of 2 is stored. You can then program a robot to perform an appropriate operation on the target, based on the evaluation result of the Inspection function. Unlike other *i*RVision functions designed to compensate robot motion, the Inspection function does not output offset data or measurement values.




| VERDICT | OUTPUT |
|---|---|
| FAIL | 0 |
| PASS | 1 |
| UNDETERMINED | 2 |

## 2.2 DEFINITIONS

When the setup screen for a vision process is opened, a tree view indicating the vision process structure is displayed at the upper right of the screen. Command tools to be added to the tree view differ depending on the details of inspection. This section explains terms related to the tree view and tree view basic rules using a vision process having the structure as shown in the figure below as an example.


## Vision Tool

As shown in the figure above, a vision process consists of several elements. Each element is called a vision tool. The top vision tool named “Single-View Inspection VisProc” is the main vision process and other vision tools are called command tools. An Inspection vision process requires at least two command tools. At least one command tool must be an evaluation tool. Other command tools are added according to the details of inspection. Command tools can be divided into the following four types:

Locator Tool

A locator tool performs image processing, detects a target in the image, and outputs the position where the target is detected. This tool also outputs the score, contrast, and other characteristics of the detected target as measurement values. In addition, the tool can output positional information as measurement values.

Measurement Tool

A measurement tool also performs image processing like a locator tool, but does not output a position. For example, this tool measures the brightness of the specified section in an image and outputs the result as a measurement value.

Evaluation Tool

An evaluation tool evaluates whether a target passes or fails inspection. This tool receives measurement values output by locator and measurement tools and outputs an evaluation result by checking whether or not the set conditions are met. An Inspection vision process requires at least one evaluation tool.

Others

Command tools other than the above. Various types of command tools are available: For instance, one tool performs arithmetic operation, another performs a geometric calculation, and another counts the number of targets found. Most tools output calculation results and other data as measurement values, and the calculation results can be evaluated using an evaluation tool.

## Parent tool, child tool, and sibling tool

If you look at the tree view carefully, you will find that the vision tools are not arranged in a row and some vision tools are indented. Relationships among vision tools are represented by thin dotted lines. For example, “Histogram Tool 1” is connected with “Edge-Pair Tool 1” with a thin dotted line and is indented one level to the right of “Edge-Pair Tool 1”. In this case, “Histogram Tool 1” is considered as a child tool of “Edge-Pair Tool 1” and “Edge-Pair Tool 1” is the parent tool of “Histogram Tool 1”. Similarly, “Evaluation Tool 2” is a child tool of “Edge-Pair Tool 1” and “Edge-Pair Tool 1” is the parent tool of “Evaluation Tool 2”. “Histogram Tool 1” and “Evaluation Tool 2” are considered as sibling tools since they share the same parent tool. “Single-View Inspection VisProc”, which is placed at the top, has four child tools, “Window Shift Tool 1”, “Edge-Pair Tool 1”, “Count Tool 1”, and “Evaluation Tool 1”.

## Tree view rules

Basic rules on the tree view for Single-View Inspection are explained below.

- The tree view indicates the execution sequence. The command tools are executed sequentially from top to bottom.
- Single-View Inspection is always the first vision tool.
- An evaluation tool is always placed as the last sibling tool.
- The result of the evaluation tool set as a child tool of the vision process is used as the final evaluation result of the vision process.
- A child tool cannot be moved or copied as a child tool of a vision tool which is not its parent tool.
- Child tools that can be added differ depending on the vision tool and the tree structure.

# 2.3 COMMAND TOOLS FOR INSPECTION

When a new Single View Inspection vision process is created, only one evaluation tool is set. Add command tools to the vision process according to the details of inspection. This section simply introduces what part each command tool plays in inspection. For details of each command tool, refer to the *iRVision Operator's Manual(Reference)*.

## 2.3.1 Locator Tools

A locator tool performs image processing for a snapped image, detects a target specified in advance, and outputs the position of the target in the image as a measurement value. At the same time, the tool outputs the values of location parameters when the target is detected.

### 2.3.1.1 GPM locator tool

A GPM locator tool detects a geometry taught as a model pattern in an image and outputs the position of the geometry detected in the image. Since the contour line of a target in an image is used for detection, the position of the target can be obtained accurately. For inspection, a GPM locator tool can be used as follows:

Locating parts :
Shifts the measurement window of a command tool performing inspection, in combination with a window shift tool or dynamic window function described later. A sample application of a window shift tool is introduced in Subsection 5.1.1, "Inspecting Whether Soldering Is Applied", and a sample application of dynamic window function is in Subsection 5.3.3, "Inspecting Whether Products with Different Specifications Are Mixed".

Counting number of parts :
Counts the number of targets in combination with a count tool described later. Use a GPM locator tool for this purpose when targets appear almost the same in geometry, since matching is performed based on the geometry. A sample application is introduced in Subsection 5.3.1, "Counting the Number of Targets in an Image".

Checking presence/absence :
Basically the same as counting number of parts. This function can evaluate whether the number of found parts is 0 or 1 to check whether the expected target is present or absent.

Measuring length :
Measures the length and angle in combination with a geometric calculation tool described later. For example, you can use a GPM locator tool to detect two holes and a geometric calculation tool to calculate the distance between the holes.

The following measurement values are available for inspection:
- Position

- Angle
- Size
- Spect ratio
- Skew angle
- Score
- Fit error
- Contrast

### 2.3.1.2 Curved surface locator tool

A curved surface locator tool detects the same pattern as a model pattern taught in advance from an image using gradation (change from light to dark or vice versa) and outputs the position of the pattern in the image. The positional accuracy is lower compared to a GPM locator tool. This tool can detect a target with an ambiguous contour line which is difficult to detect using a GPM locator tool. For inspection, a curved surface locator tool can be used as follows:

Locating parts :
Shifts the measurement window of a command tool performing inspection in combination with a window shift tool or dynamic window function described later.

Counting number of parts :
Counts the number of targets in combination with a count tool described later. Use this function when the target has an ambiguous contour line since matching is performed based on gradation.

Checking presence/absence :
Basically the same as counting number of parts. This function can evaluate whether the number of found parts is 0 or 1 to check whether the expected target is present or absent.

The following measurement values are available for inspection:

- Position
- Angle
- Size
- Score

### 2.3.1.3 Blob locator tool

A blob locator tool extracts a region which has the same amount of characteristic as the model taught in advance from a binarized image and outputs the position of the region in the image. This tool is suitable for detecting a target when its geometry is variable or for measuring an area. Unlike a GPM locator tool, this tool does not search for the same geometry, but measures the amount of characteristics such as an area and perimeter length of a target. When using a blob locator tool, be careful about the lighting environment because the tool binarizes an image before image processing. For inspection, a blob locator tool can be used as follows:

Locating parts :
Shifts the measurement window of a command tool performing inspection in combination with a window shift tool or dynamic window function described later.

Counting number of parts :
Counts the number of targets in combination with a count tool described later. Use this function to count the number of targets that are variable in geometry.

Checking presence/absence :

Basically the same as counting number of parts. This function can evaluate whether the number of found parts is 0 or 1 to check whether the expected target is present or absent.

Measuring length :

Measures the length and angle in combination with a geometric calculation tool described later. For example, you can use a blob locator tool to detect two holes and a geometric calculation tool to calculate the distance between the holes.

Measuring area :

A blob locator tool can measure an area. The area output by the tool itself is the area of each detected area, but you can use this tool in combination with a statistic calculation tool to measure a total area. A sample application is introduced in Subsection 5.6.1, “Inspecting the Amount of Coated Chemicals”.

The following measurement values are available for inspection:

- Position
- Score
- Angle
- Area
- Perimeter
- Circularity
- Semi-major axis length
- Semi-minor axis length
- Elongation

#### 2.3.1.4 Edge pair locator tool

An edge pair locator tool detects a parallel edge pair from an image and outputs the center position of the edge pair and the distance between the edges. You can set a measurement plane in an inspection vision process to evaluate whether the target passes or fails in millimeters, by converting the length between edges to millimeters. For inspection, an edge pair locator tool can be used as follows:

Measuring length :

Use this function for a dimensional inspection for molded or machined items and for a clearance inspection after an assembly process. A sample application is introduced in Chapter 4, “SETUP” and Subsection 5.4.1, “Inspecting the Width”.

Detecting position :

The length from a reference line can be measured to check whether a label is attached in position. A sample application is introduced in Subsection 5.2.1, “Inspecting the Location Where a Label Is Attached”.

Measuring interval :

For items having the same dimension that are equally spaced, such as IC leads and connector holes, the interval can be inspected. Use this function in combination with a statistic calculation tool. A sample application is introduced in Subsection 5.4.2, “Inspecting the Interval”.

Counting number of parts :

For items having the same dimension that are equally spaced, such as IC leads and connector holes, the number of intervals can be counted. Use this function in combination with a count tool.

The following measurement values are available for inspection:

- Position

- Angle
- Contrast
- Distance between edges

## 2.3.2 Measurement Tools

Measurement tools perform image processing like locator tools, but do not output positions. For example, these tools measure the brightness of the specified section in an image and output the result as a measurement value.

### 2.3.2.1 Histogram tool

A histogram tool measures the brightness of an image. This tool can be used for a wide variety of applications including type identification and an evaluation of whether there is a target or impurity, using the brightness of the measurement area. A sample application is introduced in Subsection 5.1.1, “Inspecting Whether Soldering Is Applied”.

The following measurement values are available for inspection:

- Number of pixels
- Brightness of brightest pixel
- Brightness of darkest pixel
- Median of brightness
- Mode of brightness
- Mean of brightness
- Standard deviation of brightness
- Ratio of pixels within the range
- Ratio of pixels outside the range

### 2.3.2.2 Edge histogram tool

An edge histogram tool measures the changes (gradients) in the brightness of an image. A sample application is introduced in Subsection 5.1.2, “Inspecting Whether There Is a Thread”.

The following measurement values are available for inspection:

- Number of pixels
- Maximum
- Minimum
- Median
- Mode
- Mean
- Std. Dev.
- Ratio of pixels within the range
- Ratio of pixels outside the range

### 2.3.2.3 Surface flaw inspection tool

Surface flaw inspection tool finds defects on the surface of a target object. First, regions that seem suspicious are extracted by searching within the specified search window. Then, the measurements of potential flaws such as individual flaw area and length, as well as the number of found flaws and the ratio of total flaw area to the search window area, are evaluated. For inspection, a surface flaw inspection tool can be used as follows:

Counting number of defects:
Counts the number of defects on the surface of a target object.

Finding defects on molded plastics and the like:
Find smudges, discoloration and contamination on molded plastics and the like.

Finding defects on metal surfaces and the like:
Find dents and scratches on metal surfaces and the like.

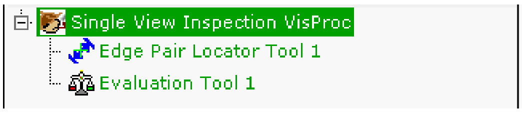
The following measurement values are available for inspection:

- Number of found defects
- Total area
- Flaw ratio
- Inspected ratio
- Max area
- Max perimeter
- Max magnitude

### 2.3.3 Evaluation Tool

An evaluation tool determines whether an inspection passes or fails by evaluating specified conditional expressions. You can define multiple conditional expressions and have the tool finally determine whether an inspection passes or fails by evaluating those multiple conditional expressions in a comprehensive fashion.

Normally, a Single View Inspection vision process has only one evaluation tool as shown in the figure below:




In the above example, "Edge-Pair Tool 1" measures the length, and "Evaluation Tool 1" evaluates the measured length and evaluates whether the inspection passes or fails.
An evaluation tool can also be set as a child tool of a locator tool as shown in the figure below:


In this case, "Evaluation Tool 2" evaluates whether each target detected by "GPM Locator Tool 1" passes or fails. Then, "Count Tool 1" counts the number of passed and failed targets determined by "Evaluation Tool 2", and finally, "Evaluation Tool 1" makes such a final determination that the inspection is passed when all targets detected by "GPM Locator Tool 1" are determined as pass. For how evaluation tools are used in this example, see also Section 2.4, "DYNAMIC WINDOW".

Various sample applications of evaluation tools are introduced in Chapter 5, "SAMPLE APPLICATIONS".

## 2.3.4 Other Tools

Some command tools are not classified as any of locator, measurement, or evaluation tools. These command tools include a tool which performs basic arithmetic operations, a tool which performs a geometric calculation, and a tool which counts the number of targets found. Most tools output calculation results and other data as measurement values, and the calculation results can be evaluated using an evaluation tool.

### 2.3.4.1 Count tool

A count tool counts the number of targets found by locator tools. As well as simply counting the number of targets detected by a locator tool, a count tool can count the number of targets with a specific model ID, the number of passed targets, and the number of failed targets. A sample application is introduced in Subsection 5.3.1, “Counting the Number of Targets in an Image” and Subsection 5.3.3, “Inspecting Whether Products with Different Specifications Are Mixed”.

The following measurement values are available for inspection:

- Num. Found
- Num. Passed
- Num. Failed

### 2.3.4.2 Arithmetic calculation tool

An arithmetic calculation tool performs basic arithmetic operations for measured values. For example, it can calculate the difference between the mean brightness values measured by two histogram tools. A sample application is introduced in Subsection 5.2.1, “Inspecting the Location Where a Label Is Attached”.

The following measurement value is available for inspection:

- Calculation result

### 2.3.4.3 Geometric calculation tool

A geometric calculation tool performs a geometric calculation using the positions found by locator tools. For example, it can calculate the distance between the holes found by two locator tools.

The following measurement value is available for inspection:

- Calculation result

### 2.3.4.4 Statistics calculation tool

A statistics calculation tool performs a statistics calculation for the measured values of targets found by locator tools. For example, when a blob locator tool has found five blobs, a statistics calculation tool can calculate an average area or a standard deviation of the five blobs. A sample application is introduced in Subsection 5.4.2, “Inspecting the Interval” and Subsection 5.6.1 “Inspecting the Amount of Coated Chemicals”.

The following measurement values are available for inspection:

- Maximum
- Minimum
- Mean
- Std. Dev.
- Total

- Found

### 2.3.4.5 Position calculation tool

The position calculation tool calculates a new position from other found positions. For example, it can calculate the intersection of two lines found by two line locator tools, the foot of perpendicular from a hole found by a GPM locator tool to a line found by a line locator tool, and so on.

The following measurement value is available for inspection:

- Calculation result

### 2.3.4.6 Image preprocess tool

The image preprocess tool has the color extraction tool and the image filter tool. You can treat the image which is processed by the color extraction tool and the image filter tool which are the child tool of the image preprocess tool as the input image of the command tool which is used for a detection and an inspection. Only one image preprocess tool can be inserted just below the vision process.

### 2.3.4.7 Conditional execution tool

A conditional execution tool evaluates the result of a histogram tool or other tools based on specified conditions and, only if the conditions are met, executes the specified processing. A conditional execution tool does not output any measurement values, but operates on the detection result of its parent tool. Use this tool in a case where you want to exclude targets detected by a parent locator tool that meet a specific condition from targets to be evaluated for inspection.

### 2.3.4.8 Window shift tool

A window shift tool dynamically shifts the search windows of the sibling tools that are placed below it in the tree view, using the position detected by its child locator tool. For example, for a vision process having the structure as shown in the figure below, "Window Shift Tool 1" shifts the search windows of its sibling tools below it, that is, "Edge-Pair Tool 1" and "Histogram Tool 1". Use a window shift tool when targets to be inspected are not fixed and where they appear in an image is not consistent.


When a child locator tool of a window shift tool finds multiple targets, the execution result with the highest score is used to shift the search windows of the sibling tools.

### 2.3.4.9 Multi-locator tool

The Multi-locator tool changes the locator tool to be executed, according to the value set in a robot register.

### 2.3.4.10 Multi-window tool

The multi-window tool changes the search window to be used, according to the value set in a robot register.

# 2.4 DYNAMIC WINDOW

A command tool placed as a child tool of a locator tool is executed repeatedly as many times as the number of targets detected by the parent locator tool. For example, for a vision process having the structure as shown in the figure below, when “GPM Locator Tool 1” finds four targets, “Histogram Tool 1” and “Evaluation Tool 2” are executed for each found target, for a total of four times each. During this execution, the measurement window of the child tool which performs image processing (“Histogram Tool 1” in the figure below) is shifted according to the position detected by the parent locator tool (“GPM Locator Tool 1” in the figure below). This effect is called a dynamic window.


When there are multiple targets to be inspected within the camera view, the use of a dynamic window function is effective. For example, for a vision process having the structure as shown in the figure above, “GPM Locator Tool 1” finds multiple targets to be inspected in an image, “Histogram Tool 1” performs measurement for each found target, and “Evaluation Tool 2” evaluates whether each found target passes or fails inspection. After the inspection result of individual targets have been determined as pass or fail, “Evaluation Tool 1” finally evaluates whether the inspection is passed in a comprehensive fashion based on the inspection result of individual targets (for example, the inspection comprehensively passes when the inspections of all targets within the camera field of view pass).

# 2.5 RESTRICTIONS

The following restrictions are imposed on the *i*RVision Inspection function:

The accuracy of inspection depends on the size of the camera field of view, apparent size of a target in an image, lighting environment, and other factors. Carefully consider the camera and lens to be used, where the camera is installed, and the layout of lighting and other devices. In particular, in a place where direct sunlight is received or where the ambient light changes greatly, measurements could become unstable. Install the system in another place or take measures to reduce the effect of ambient light.

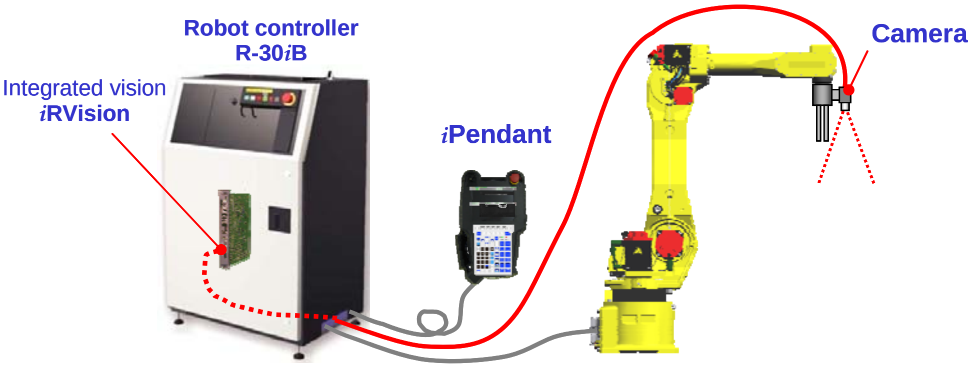
Depending on a TP program or vision process structure, the runtime display may not show the exact cause of a failed inspection. In such a case, enable the vision log function and run a test from the *i*RVision setup screen, with the saved image of the defective target, to reproduce the inspection result.

# 3 STUDY FOR APPLICATION

Thorough considerations and preparations are recommended before setting up an inspection system. Be careful about the differences between the use of a vision system for inspection and for robot compensation. This chapter explains some items to consider when preparing for an application of an inspection system.

## 3.1 SAMPLE SYSTEM CONFIGURATIONS

The following figure shows the configuration of a basic inspection system using *i*RVision.




In addition to components of an ordinary robot system, the following components are required:

- Camera
- Lens
- Lighting
- Camera cable

## 3.2 DIFFERENCES BETWEEN POSITION COMPENSATION AND INSPECTION

The main difference between position compensation and inspection is as follows: While a vision system for position compensation must “detect a target even when its snapped image is apparently a little different from the model pattern”, a vision system for inspection must “find any slight difference between a target and the model pattern and evaluate whether the target passes or fails”. For this reason, when performing an inspection, be extra careful about the layout of the camera and the lighting, and the change in the brightness of the ambient environment, so that any slight difference between a target and the model pattern clearly appears in the camera image. To perform both position compensation and inspection in one system, using a camera for inspection in addition to a camera for detecting the location of a target may ensure more reliable inspection.

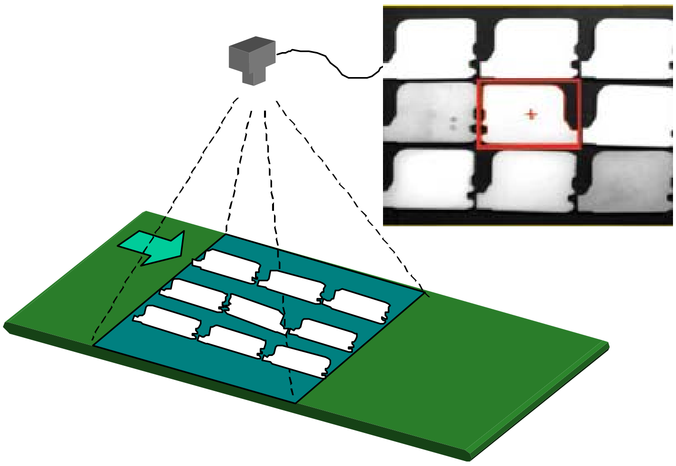
In a system for inspecting the length or area as well as the presence/absence of each feature, make sure that the relative relationship between the camera and the target is fixed. To inspect the size or the area of a target, it is desirable to place a camera so that the optical axis of the camera is perpendicular to the measurement plane. If the optical axis of the camera is not perpendicular to the measurement plane, an accurate inspection is difficult because the apparent size of a target varies depending on where the target appears in the image. To evaluate the relative relationship between multiple features, it is important that these features should be on the same plane. Note that if these requirements are not satisfied, the measured values may not be evaluated correctly.

# 3.3 FIXED CAMERA AND ROBOT-MOUNTED CAMERA

Consider the method for installing a camera. A camera can be installed as a fixed camera or a robot-mounted camera, and these methods have their own advantages. Select an appropriate camera installation method according to the inspection system you want to configure.

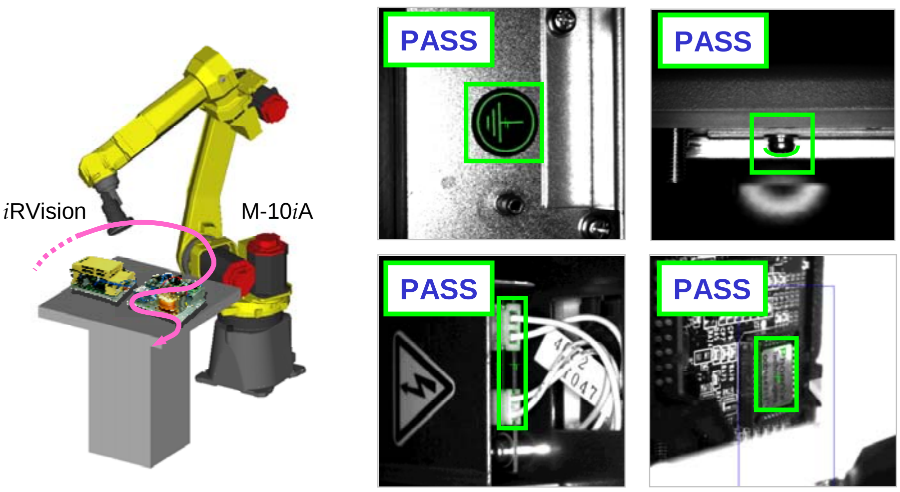
## Fixed camera

A fixed camera is attached to a pedestal or on another fixed structure. In this method, the camera always sees the same area from the same distance. One advantage of a fixed camera is that the robot cycle time can be reduced because *i*RVision can capture and process an image while the robot performs another task. With this method, you can inspect the target held by a robot or inspect targets set on a table or on a conveyor.



## Robot-mounted camera

A robot-mounted camera is usually mounted on the wrist unit of the robot, or on a robot hand. This method gives the capability of inspecting various locations with one camera by moving the robot. Using a robot-mounted camera, inspection which conventionally required multiple cameras can be performed with one camera. Also, a location that is difficult to view through a fixed camera can be inspected by moving the robot to view the location through the robot-mounted camera. This method is useful for viewing several locations of a target for inspection such as visual appearance inspection after an assembly process.



# 3.4 SELECTING THE LENS

Select the lens to be used.

## 3.4.1 Size of the Camera Field of View

First, determine the size of the camera field of view based on the size of a target or the location to be inspected. The larger the location to be inspected appears in the image, the more accurately the inspection can be performed. If the position of the target shifts during an inspection because the target is not fixed, determine the size of the camera field of view to include the amount of the shift.

## 3.4.2 Focal Distance of the Lens

After determining the size of the camera field of view, determine the focal distance of the lens to be used. When the distance between the camera and target is the same, the target through a lens with a short focal distance looks smaller because the camera field of view covers a wider area. On the contrary, the target through a lens with a long focal distance looks larger because the camera field of view covers a narrower area. For this reason, when the size of the camera field of view is determined, the distance between the camera and target becomes shorter if a lens with a short focal distance is used and it becomes longer if a lens with a long focal distance is used. The relationship between the focal distance of the lens and the distance between the camera and target is calculated by the following relational expression:

$L = f \times H \div h$

L : Distance between the camera and detection target
f : Focal distance
H : Size of the camera field of view in the vertical direction of the screen
h : Size of the light receiving element in the vertical direction of the screen

For the Kowa digital camera, the cell size of the light receiving element is 6.7 μm and the number of valid pixels in the vertical direction of the screen is 480, so h = 6.7 μm × 480 = 3.216 mm. If you want to set the size of the camera field of view in the vertical direction of the screen (H) to 100 mm, the following distances between the camera and target can be estimated using the formula:

| | |
|---|---|
| When the focal distance of the lens f is 8 mm | $L = 8 \times 100 \div 3.216 = 248$ mm |
| When the focal distance of the lens f is 12 mm | $L = 12 \times 100 \div 3.216 = 373$ mm |

The Sony XC-56 Analog camera’s cell size is 7.4 μm.

Consider where the camera should be installed according to the system configuration and select a lens with the focal distance according to the camera installation position.

## 3.4.3 Other Criteria for Selecting the Lens

Consider the following items in addition to the focal distance when selecting the lens. Determine the values according to the inspection system you want to configure.

### Depth of focus

The aperture of a lens is effective in adjusting the depth of focus as well as the brightness of the camera image. The depth of focus means a distance range from the camera in which the objects appear to be in focus. Closing the aperture makes the depth of focus deeper and opening the aperture makes it shallower. That is, when the aperture is closed, objects at different distances appear to be in focus; when it is opened, the closer or further appear blurred.

Generally, the aperture is closed (intense light is required) to make the depth of focus deeper. To avoid capturing the background clearly in an image, the diaphragm may be intentionally opened so that only the section of interest is in focus.

#### Shortest range

The minimum distance at which objects appear to be in focus is called the shortest range. The shortest range in which objects appear to be in focus differs depending on the lens used. If the camera is moved closer to the target than the shortest range of the lens, the target appears blurred. If you want to make the camera closer to the target than the shortest range, use a macro lens or an extension tube. When an extension tube is used, the depth of field tends to become shallower.

#### Distortion

An image snapped through a lens is usually distorted. This symptom is called distortion. Generally, lenses with shorter focal distances have greater distortion. High-precision lenses having less distortion with the same focal distance are also available. Distortion does not often become a problem when inspecting for the presence/absence of a target. To measure the length or the area, select a lens having less distortion whenever possible.

## 3.5 SELECTION OF THE LIGHT

Select the type of lighting and lighting method to be used.

For inspection, lighting is not only used for simply lighting up the inspection area, but it is also an important measure to clearly capture the features of the location to be inspected in an image. For example, an environment in which enough intense light is applied to the target, compared to the brightness of the surrounding area, the effect of the changes in the ambient brightness is minimal for the brightness of the image. A camera generates an image by receiving the light thrown from a lighting device and reflected by the target on the CCD light receiving surface. In an inappropriate lighting environment, images allowing stable inspection cannot be obtained. Appropriate lighting with proper type and color of lighting and lighting method allows images in which the features to be inspected are emphasized to be snapped even for the same target.

### 3.5.1 Types of Light Sources

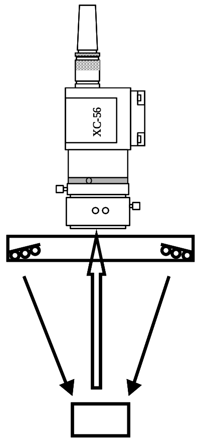
Generally, fluorescent and mercury lamps are used in many manufacturing premises. These lamps are not very suitable for inspection, however. For inspection, the use of LED lighting separately is recommended, which has a long life, high brightness, a resistance to vibration, and stability.

### 3.5.2 Lighting Method

The following lighting methods are available for inspection. Select an appropriate lighting method according to the targets to be inspected.

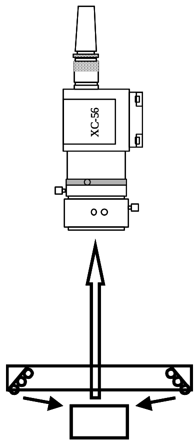
#### Ring light

The target is illuminated from a ring light source. A ring light is used for a wide variety of applications including general surface inspection because it evenly illuminates the entire target. Since the light is relatively highly directional, take measures to prevent the light from being reflected when a target has a shiny surface. These measures may include combining a diffusion filter, adjusting the distance between the light and target, or using a ring light with larger diameter. Generally, install a right light at the end of the lens of the camera.



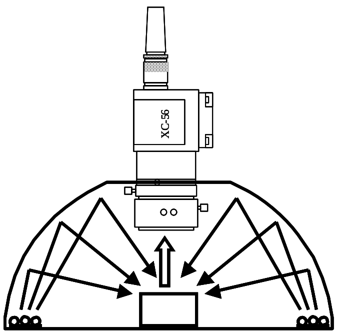

## Low-angle light

A ring light source is used. The light source is installed at a position nearer to the target as compared with a general ring light and the target is illuminated at a low angle. Since any unevenness on the target is illuminated, this method is suitable for inspecting flaws and stamps on the surface and broken contour line. Generally, install a low-angle light just above the target.



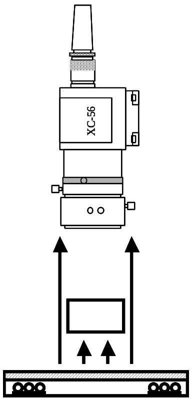

## Dome light

Light is thrown from bottom up on a dome reflector and the reflected light is thrown on the target. The target is illuminated evenly in all directions via the diffuser, which prevents the light source from being reflected on the target and provides light on an uneven or curved surface with even brightness. Since a dome light is generally installed so that it covers the target, the change in ambient brightness has little affect on the captured image, but a relatively large installation space is required around the inspection target.



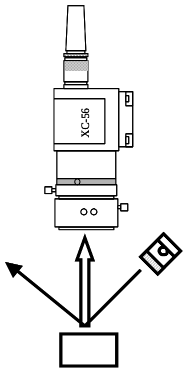
## Backlight

An inspection target is placed between the lens and the light, and light is thrown from the back of the target. This is called backlighting. This method is suitable for measuring the shape and dimensions, and inspecting foreign matters using the silhouette of the target. It is necessary to select a plane surface light having an emitting surface larger than the target, and a light installation space is required on the back of the target.



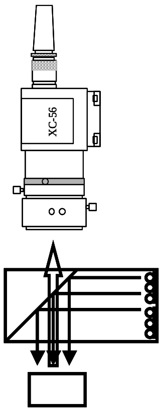

## Bar light

Light is thrown on the target in an oblique direction using a bar light source. Shiny planes on the target appear dark and flaws and stamps appear bright. With a ring light and other methods, the following problems may occur since the irradiation angle of the light is fixed: Light is partially reflected strongly depending on the shape of the target. The light is reflected on a shiny surface, regardless of curved or plane, of the target. With a bar light, an appropriate irradiation angle can be set to avoid these problems.




## Coaxial vertical light

By aligning the optical path of the light with the optical axis of the lens using a half mirror, light is thrown vertically on the target. A shiny plane perpendicular to the optical axis of the camera appears bright by halation and fine foreign particles and flaws on the surface appear dark. This method is suitable for an application which requires differences between uneven and even surfaces to be made clear. Install a coaxial vertical light between the camera and target.




## 3.5.3 Lighting Color

Changing the lighting color may make features easy to extract depending on the details of inspection. For example, blue light having a shorter wavelength is more suitable for inspecting fine flaws on the surface of the target than red light having a longer wavelength. If printed characters and flaws both appear dark in the image and the flaws cannot be extracted, throwing light in a color similar to the color of the printed characters can lighten the color of the printed characters, which would enable the inspection of the printed material.

# 3.6 FILTER

Attach a filter to the lens as required.

### Color filter

A color filter can be used to limit the wavelength of light entering the camera. For example, when a red-transmitting filter is used, the red sections of a target appear bright and other sections appear dark since only the red light is transmitted through the filter and enters the camera. A filter in the same color as the lighting can be used to reduce the influence of the ambient light. For example, a red LED light and red-transmitting filter can be used together to snap an image with stable brightness less susceptible to change in the ambient light since the reflecting light thrown from the red LED mainly enters the camera.

### Polarizing filter

A polarizing filter can prevent unnecessary light from being reflected in the camera image. Normal natural light is vibrating in various directions (called non-polarized light), but light reflected from the surface of water, glass, or metal is polarized in a specific direction. A polarizing filter can be used to reduce the reflection of the light from the surface of metal. A polarizing filter is also called a PL filter.

### Protecting filter

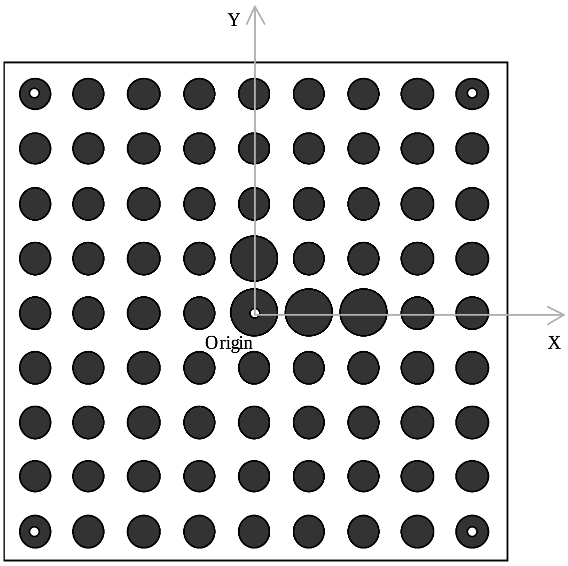
When a camera is used in an environment with dust or oil mist in the air the lens must be cleaned periodically. Dust and dirt attached on the lens may cause a detection error or non-detected error, which prevents correct inspection. Attaching a protecting filter at the end of the lens can greatly reduce cleaning man-hours. If the protecting filter becomes dirty, replace it with a spare protecting filter and clean the dirty protecting filter to use it as a spare, which can minimize production down time.

> ⚠ **CAUTION**
> Carefully attach and detach a filter so as not to make the lens loose or cause the aperture and focus to change.

# 3.7 CALIBRATION GRID

In an ordinary image inspection, a target is evaluated using the length in the image. The *i*RVision Inspection function allows evaluation after converting the length to millimeters. To convert the length to millimeters, a measurement plane must be set correctly. To set a measurement plane, view the calibration grid pattern as shown in the figure below with the camera used for inspection. Setting a measurement plane can reduce the influence of an alignment error between the camera and the measurement plane, and can compensate lens distortion when detecting positions and calculating measurements.

For inspection for which you want to evaluate the length in millimeters, a calibration grid appropriate for the size of the camera field of view is required. A FANUC standard calibration grid is available.



# 4 SETUP

This chapter explains the basic setup of Single View Inspection Vision Process. In an actual application, the configuration and taught content of a vision process are determined depending on the inspection.

## 4.1 ENTIRE FLOW

This section outlines the flow of setting up a vision process using the Single View Inspection Vision Process.

- Camera Installation and Connection
- Create a Camera Setup
- Teach the Camera Setup
- Create an Inspection Vision Process
- Teach the Vision Process
- Add Command Tools
- Teach Command Tools
- Teach the Evaluation Tool
- Test the Vision Process
- Create Robot Program

## 4.2 CAMERA INSTALLATION AND CONNECTION

Install a camera and connect the camera to the robot controller.

### Checking the Camera Setting

Change the setting of the camera unit to match *i*RVision.
For details, refer to the *iRVision 2D Vision application Operator's Manual*.

### Connecting the Camera

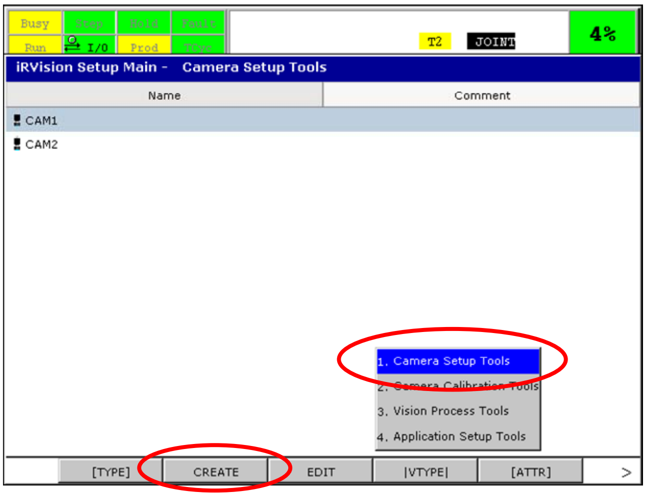
Connect the camera to the robot controller.
For details, refer to the *iRVision 2D Vision application Operator's Manual*.

### Installing the Camera

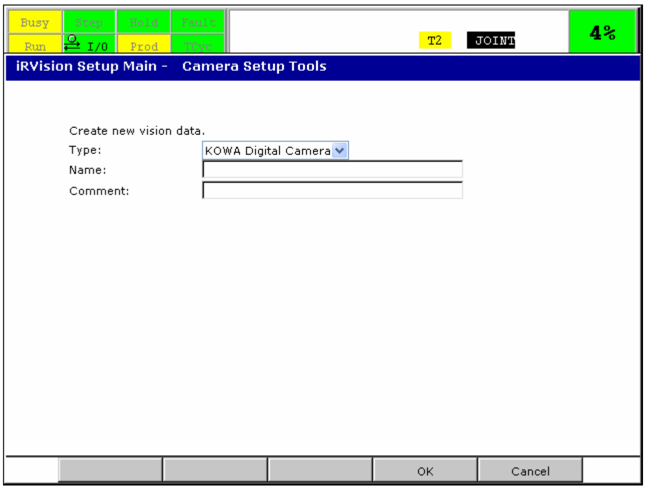
Attach a lens to the camera and install the camera on the robot hand or on a camera stand. If the inspection target is already positioned, adjust the robot measurement position or the camera position so that the feature to be inspected is located at the center of the camera field of view. The image may be distorted due to the distortion of the lens used, so the length measured near the edges of the image is slightly different from the length measured at the center. Accordingly, the area close to the center of the image should be used for inspection to minimize the effect of lens distortion whenever possible. The camera needs to be fixed securely to prevent it from being loosened and moved during operation of the system. When the camera is installed, it is convenient to make the camera position adjustable relative to the camera base. For example, if the bolt holes of the mounting bracket are formed as slit holes, the camera position can be adjusted so that the target is located at the center of the camera field of view when the position of the inspection target changes slightly.

## Creating a Camera Setup

The method of creating camera setup is described below.
Click [Camera Setup Tools] on the *i*RVision Setup Main screen.



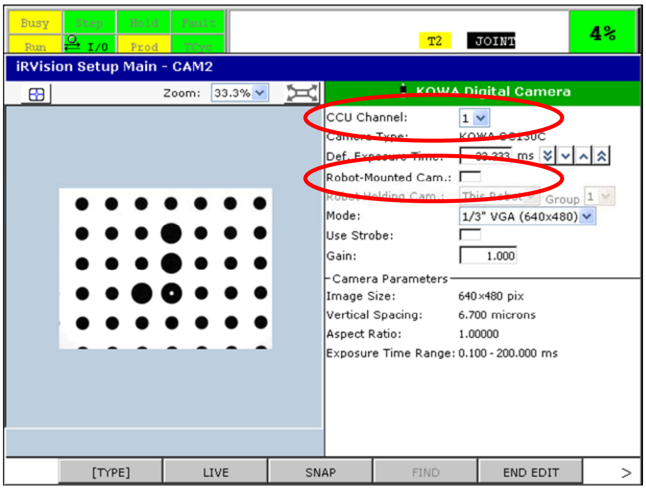

Click the “CREATE” button to display a screen as shown in the figure below.




Select the proper camera in the [Type] pulldown, enter a camera name in [Name], and click the [OK] button.

### Teaching a Camera Setup

Opening the setup screen of the created camera setup displays the following screen.




Specify the port number to which the camera is connected for [CCU Channel].
Check [Robot-Mounted Cam.] when the camera is mounted on the robot hand.

### Adjusting Lens Focus

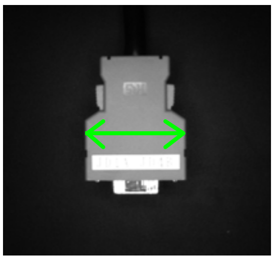
Adjust the focus of the lens on the target. Before adjusting the focus, open the aperture. The depth of focus becomes shallow when the aperture is open, so it is easy to exactly adjust the focus. Follow the procedure below to adjust the focus.

1 Set the aperture to a value little larger than what is actually used at production.
2 Adjust the exposure time so that the target captured in the image has an appropriate brightness.
3 Adjust the focus. If an object having text is placed at the same height as the target, adjustment of the focus becomes easier.
4 Adjust the aperture to achieve an appropriate depth of focus.
5 Adjust the exposure time so that the target captured in the image has an appropriate brightness.

## 4.3 VISION PROCESS TEACHING AND TEST

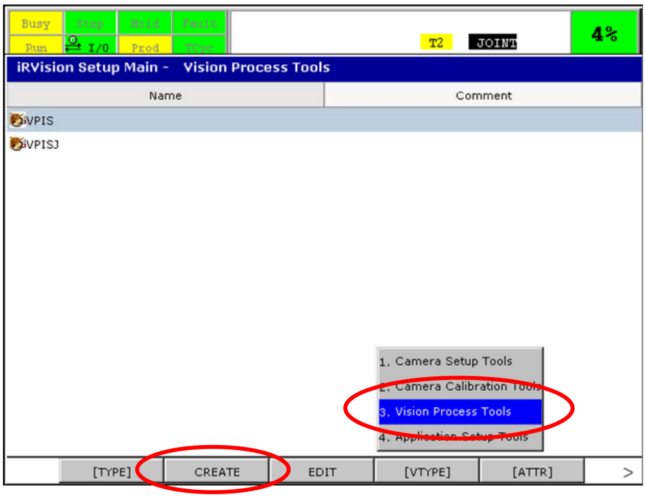
This section is written as a tutorial for creating a vision program that performs the following inspection, in an easy-to-understand manner, so that you can easily understand a flow of teaching and testing a vision program.

Description of the inspection:
The width of the connector shown in the following figure is measured to determine whether the width falls within the range from 15 mm to 16 mm. The connector is properly positioned and a fixed camera is used.

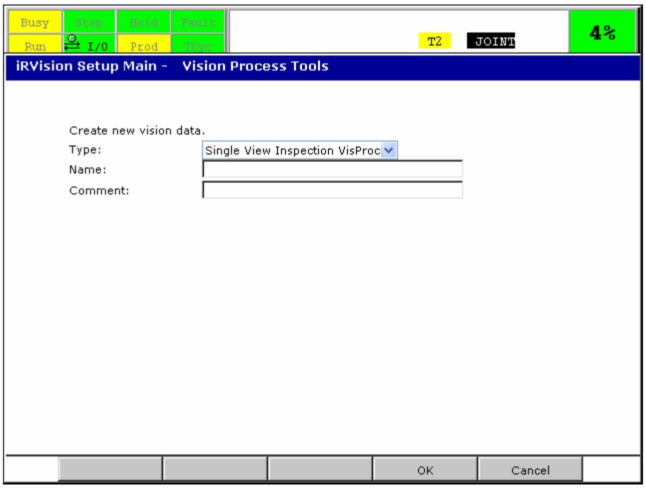


## 4.3.1 Creating Single View Inspection Vision Process

The method of creating a Single View Inspection Vision Process is described below.
Click [Vision Process Tools] on the *i*RVision Setup Main screen.



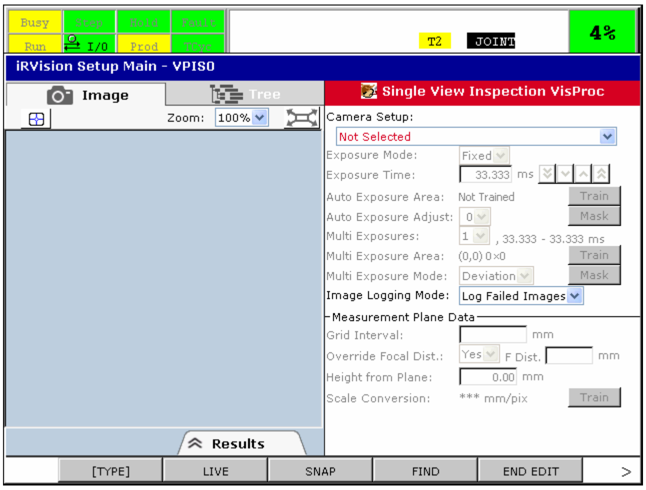

Click the ”CREATE” button to display a screen shown in the figure below.




Select [Single View Inspection VisProc] as [Type], enter the program name in [Name], and click the [OK] button.

# 4.3.2 Teaching of Single-View Inspection

Opening the created single view inspection vision process displays the following screen.




## Camera Setup

Select a camera setup to be used for inspection.
When a camera setup is selected, [Camera Setup] is set to [Trained], and [Status] is set to [Trained].

The status of the camera setup can be one of [Not Selected], [Trained], and [Invalid], which are described below.

- [Not Selected] – Camera setup is not selected. "Not selected" appears in red in the [Camera Setup] drop-down list. Select a camera setup from the drop-down list.
- [Trained] – A camera setup that can be used for image snapping is selected.
- [Invalid] – The selected camera setup is not present. The camera setup currently selected is indicated in red. The camera setup name may be changed. Select a camera setup again from the drop-down list.

If the camera setup is changed when measurement plane information is already set, a message appears asking whether the measurement plane information should be cleared. Generally, click [OK] to clear the measurement plane information. If [Cancel] is clicked, the camera setup is switched to the selected one, but the measurement plane information is kept. If the camera setup used for setting the measurement plane is changed, correct measurement cannot be obtained thereafter. Cancellation is used only when physically the same camera is selected again with only the camera setup name is changed.

## Exposure Time

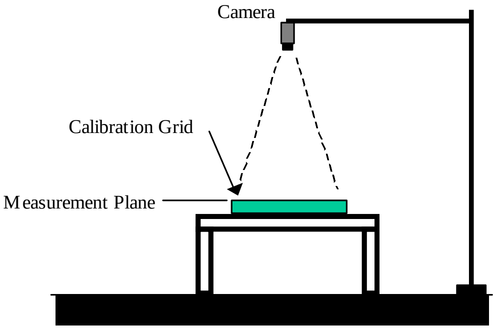
Enter the exposure time for capturing an image during the execution of the vision process. The longer the exposure time, the longer it takes for the entire processing to complete. If the lighting environment is improved so that an image can be snapped in a short exposure time, the cycle time is reduced. It is important to set the appropriate lighting environment and lens aperture before setting an exposure time suited for proper inspection.

## Multi Exposures

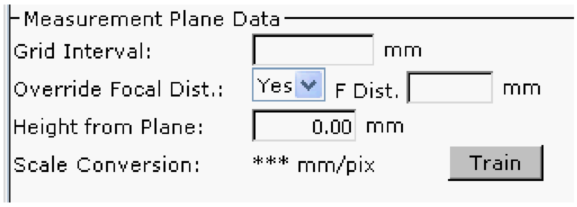
Multi exposures are used to reduce variations in brightness in an image caused by the changes in the ambient light, by synthesizing an image from multiple images snapped with different exposure time values. Since multiple images are snapped, the entire processing time is extended. In addition, multi exposures may reduce the contrast of features. Take notice when performing measurements requiring high contrast edge detection, such as measurement of lengths or areas.

### 4.3.2.1 Setting a measurement plane

In a single view inspection vision process, length is either evaluated as the number of pixels on the image or as a value converted into millimeters. To convert length into millimeters, mount the calibration grid at the same height as the measurement plane, snap the grid pattern with the camera, and set the measurement plane information. In this tutorial, the measurement plane is set because the width of the connector needs to be evaluated in millimeters.

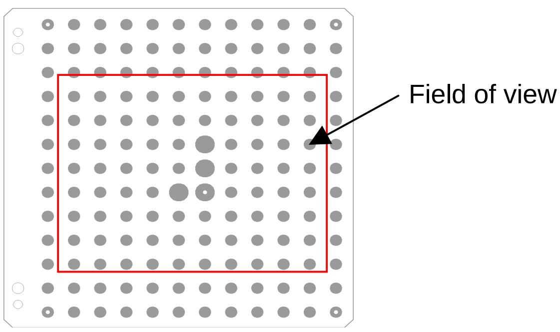



Follow the procedures below to set the measurement plane.




## Placing the calibration grid

Place the calibration grid within the camera field of view so that the grid is displayed over the entire camera field of view. It is not necessary to display the entire calibration grid within the camera field of view. If the grid is displayed only partially in the camera field of view, the measurement plane information may not be set accurately. The calibration grid needs to be placed so that the grid pattern plane is at the same height as and in parallel with the target to be measured. The height may differ between the grid pattern plane and the target if the target is in focus, but it is necessary to specify the difference in height with the [Height Offset from Plane] parameter described later. The surface plane of the target to be measured must be parallel with the calibration grid.




## Grid Interval

Input the grid interval of the calibration grid in millimeters.

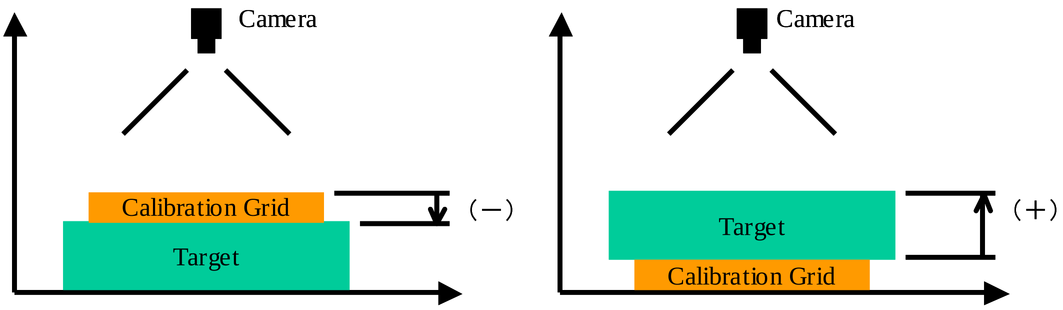
## Override Focal Distance

Select [Yes] and input the nominal focal distance of the lens in millimeters.

It is possible to select [No] to calculate the focal distance automatically. However, when the camera optical axis is perpendicular with the calibration grid, the focal distance cannot be calculated correctly. If the focal distance calculated automatically deviates from the nominal focal distance by ±10% or more, select [Yes] and input the nominal focal distance of the lens used.

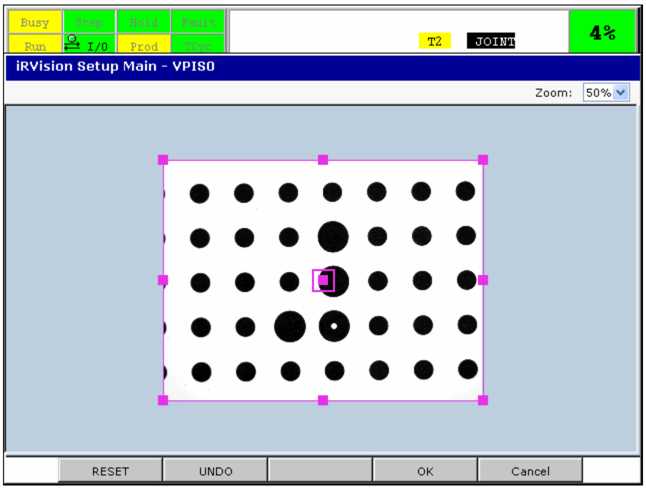
## Height Offset from Plane

The measurement plane offset value needs to be specified when the height of the measurement plane to be set (height of the target to be measured) differs from the height of the grid pattern plane of the placed calibration grid. Input a positive value in millimeters when the measurement plane to be set is closer to the camera than the grid pattern plane or a negative value when the measurement plane is farther from the camera than the grid pattern plane.




## Teaching a measurement plane

Snap an image of the calibration grid and set the measurement plane. Click the “SNAP” button to capture the image and click the [Train] button. The following screen is displayed.




Adjust the red rectangle so that the entire grid is included in the frame. Since the calibration grid is generally placed so that the grid is displayed over the entire field of view, the red rectangle is set to the full screen mode. When the [OK] button is clicked, the calibration grid is detected and the measurement plane is set. When the measurement plane is set, the average scale on the set measurement plane is displayed in [Scale Conversion]. The scale indicates the conversion factor between one pixel on the image and the length in millimeters on the measurement plane, and its unit is mm/pix.

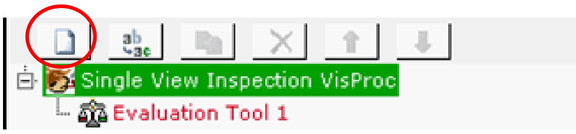
When the [Train] button is clicked and the measurement plane has already been taught, a message would appear asking whether the measurement plane information should be changed or the change should be canceled. If there are command tools that had already been taught, a change in the measurement plane may change their measurement results. If the measurement plane is re-taught, these command tools may need to be taught again as necessary.

**⚠ CAUTION**
When the measurement plane is taught after command tools had been taught, it is recommended to re-train those command tools that have measurement values subjected to millimeter conversion.

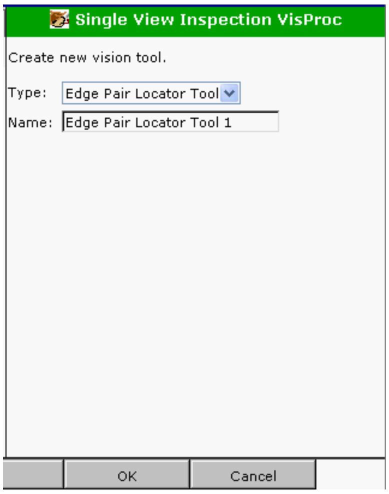
## 4.3.3 Command Tool Setup

Insert command tools for inspection into the vision process and teach the command tools.
In this tutorial, the width of a target is measured with an edge pair locator tool.

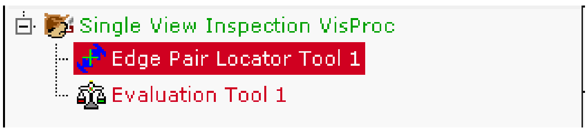
Click on [Single View Inspection VisProc] in the tree view.



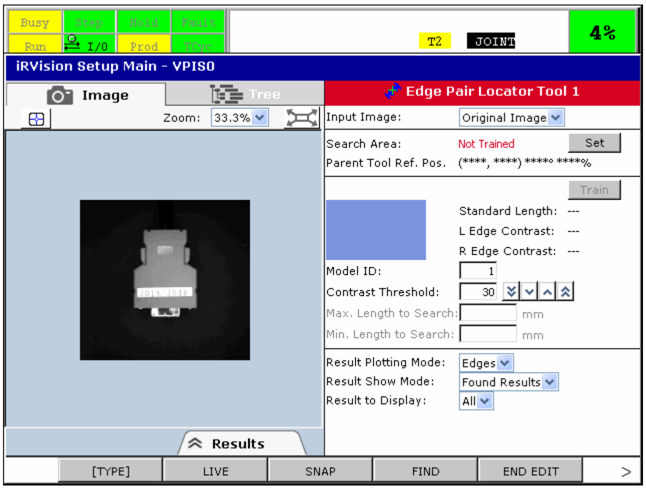

Click the  button to display a screen as shown in the figure below.




Select [Edge Pair Locator Tool] in the [Type] drop-down list and enter an appropriate tool name in the [Name] field.
In this tutorial, use the initial value [Edge Pair Locator Tool 1] in [Name].
When the [OK] button is clicked, [Edge Pair Locator Tool 1] is inserted into a position before [Evaluation Tool 1].

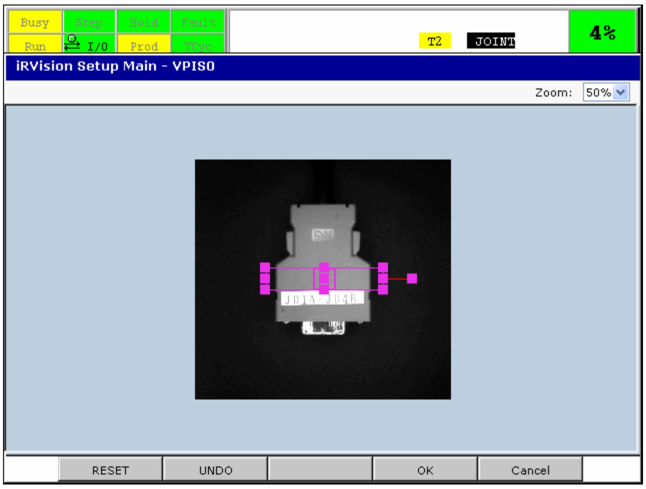


Teach the inserted edge pair locator tool.
When [Edge Pair Locator Tool 1] is clicked in the tree view, a screen as shown below appears.



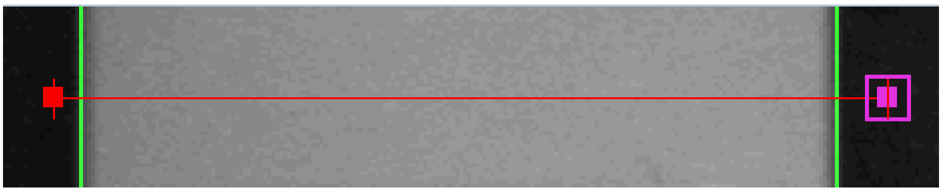

## Set Search Area

Set the search area in a position where the length is to be measured.
Click the [Set Search Area] button to display the area-setting screen as shown in the figure below. When a red rectangle appears on the screen, enclose the area to measure by the red rectangle and click the [OK] button.

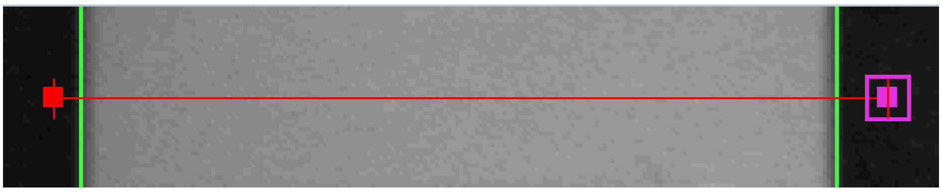



## Train Model

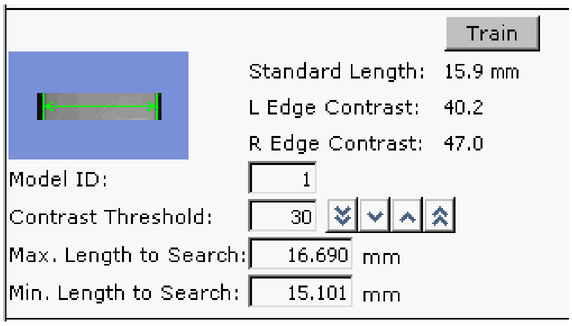
Teach the model pattern of an edge pair.
Click the [Train] button to display the edge selection screen. On the edge selection screen, select two edge lines in the search area according to the message.



First, select the starting edge. To select it, use the mouse pointer to drag the red cross-hair marker to the vicinity of the edge indicated by a green line. If the [OK] button is clicked, the starting edge is selected and the processing proceeds to the selection of the ending edge. The selected starting edge is displayed in blue.



After the model is taught, model information is displayed on the setting screen of the edge pair detection tool.




If no edges are found in the search area, a model cannot be taught. Before teaching a model, change the snapping condition or the search area or, adjust [Contrast Threshold] so that edges can be detected within the search area.

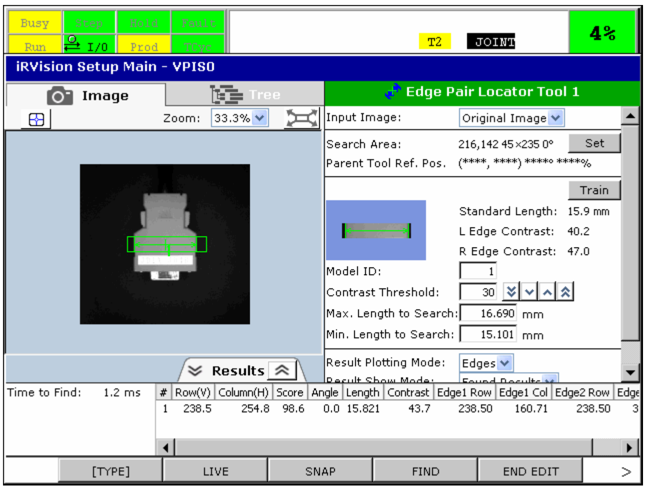
## Maximum Length to Search, Minimum Length to Search

Set the minimum length and maximum length of the edge pair to be detected. The initial values are set to 95% to 105% of the trained model.

In this tutorial, lengths of 15 mm and 16 mm are satisfactory. However, the range is slightly extended so that the length can be measured even if the width of the target is wider or narrower than expected.

## Running a test

Click the [Snap and Find] button to confirm that the detection is performed as expected.



## 4.3.4 Evaluation Tool Setup

Teach the evaluation tool.
Click on [Evaluation Tool 1] to open the setup screen of the evaluation tool.

Click the [Condition] tab or the [Variable] tab on the setup screen of the evaluation tool to switch between screens.

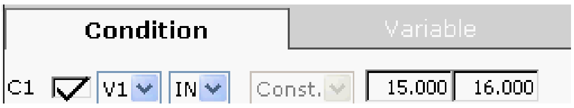
### Selecting a measurement value to evaluate

Click the [Variable] tab and select a measurement value to be evaluated.


1 Select [Edge Pair Locator Tool 1] in the left drop-down list of [V1].
2 Select [Length] in the left drop-down list of [V1].

### Setting conditions

Click the [Condition] tab and set the conditions for evaluating measurement values.

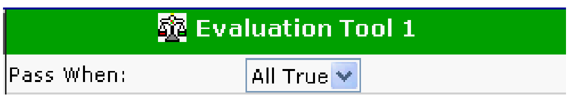



1 Check the [C1] checkbox.
2 Select [V1] in the leftmost drop-down list of [Condition].
3 Select [IN] in the second drop-down list of [Condition].
4 Select [Const.] in the third drop-down list of [Condition].
5 Enter [15.0] and [16.0] in the second rightmost and the rightmost text boxes of [Condition], respectively.

> ⚠ **CAUTION**
> 1 Even when a condition is set, if the corresponding check box is unchecked, the condition is not used for logical evaluation.
> 2 Even when a check box is checked, if the corresponding conditional expression is not set, the result will be [Not Set].

### Setting “Pass When”

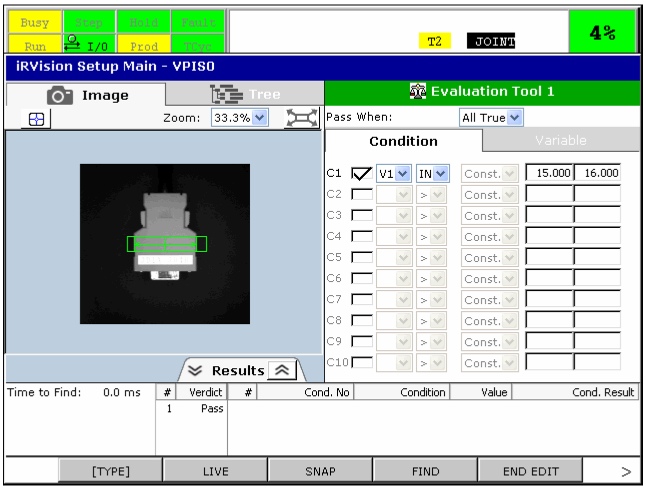
Select the condition for determining Pass or Fail.




In this tutorial, only one condition is set. Therefore, set [Pass When] to [All True]. Now, an evaluation condition where a length from 15 mm to 16 mm measured in [Edge Pair Locator Tool 1] is determined as pass has been set.

### Running a test

Confirm that evaluation is performed as expected.



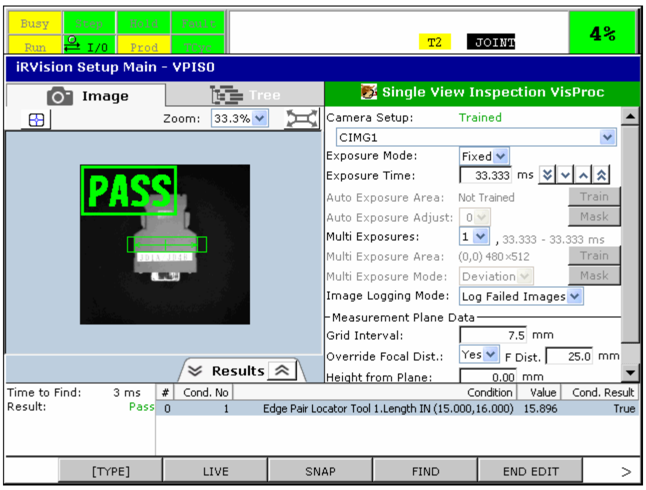

When the detection is performed, only the evaluation result is indicated on the result display screen. To display the details of the evaluation, click on the evaluation result.


One of the following results is indicated as the evaluation result of each conditional expression.

- TRUE – The condition is satisfied.
- FALSE – The condition is not satisfied.
- Not evaluated – The condition is not evaluated because the specified command tool failed to find or measure the target.
- Not set – The check box of a condition is checked, but the condition is not set.

## 4.3.5 Testing Vision Process

The single view inspection vision process is executed from the setup screen.
Click on [Single-View Inspection VisProc] in the tree view to open the vision process setup screen and click the [Snap] and then [Find] button.




The inspection result appears in the lower left corner of the image display screen. Repeat using multiple connectors and confirm that those with the width within the specified values should pass, and others fail the inspection.

# 4.4 CREATING A ROBOT PROGRAM

A sample robot program for executing the created Inspection Vision Process is shown below.

```
 1:   UFRAME_NUM=1
 2:   UTOOL_NUM=1
 3: J P[1] 100% FINE
 4:
 5:   VISION RUN_FIND  ‘VISION1’
 6:   VISION GET_PASSFAIL  ‘VISION1’  R[1]
 7:
 8:   SELECT R[1]=1, JMP LBL[20]
 9:             =0, JMP LBL[30]
10:            ELSE, JMP LBL[40]
11:
12:   LBL[20:PASS]
13:   CALL GOOD_PARTS
14:   END
15:
16:   LBL[30:FAIL]
17:   CALL BAD_PARTS
18:   END
19:
20:   LBL[40:ERROR]
21:   UALM[1]
```

In lines 1 to 3, the program moves the robot to a position where the field of view of the camera is not blocked.
In lines 5 and 6, the program executes inspection vision process VISION1 and stores the evaluation result in R[1].
In lines 8 to 10, the program branches to an appropriate process based on the evaluation result.

# 5 SAMPLE APPLICATIONS

This chapter demonstrates how different command tools can be combined to achieve various types of inspection, through some sample applications. There are multiple vision process configurations available to achieve the same inspection result. The sample applications in this chapter are provided as a guide to conceptualize a method to construct a vision process for an inspection. For your specific application, create an appropriate vision process according to the details of inspection.

## 5.1 PRESENCE/ABSENCE INSPECTIONS

This section introduces sample applications for inspecting whether an item is present. The following methods are available as main techniques to determine presence or absence:

- Measure the brightness of an image using a histogram tool.
- Measure the gradient distribution in an image using an edge histogram tool.
- Detect a taught model using a GPM locator tool.
- Detect a taught model using a blob locator tool.

### 5.1.1 Inspecting Whether Solder Is Applied

This application inspects whether the eight through holes on a printed circuit board are soldered, as shown within a circle in the figures below. Fig. 5.1.1 (a) shows a printed circuit board on which solder is applied properly to all eight through holes, thus should pass the inspection. Fig. 5.1.1 (b) shows a printed circuit board on which solder is not applied to the rightmost through hole, thus should fail the inspection. Assume that these printed circuit boards are roughly positioned and their locations slightly vary.

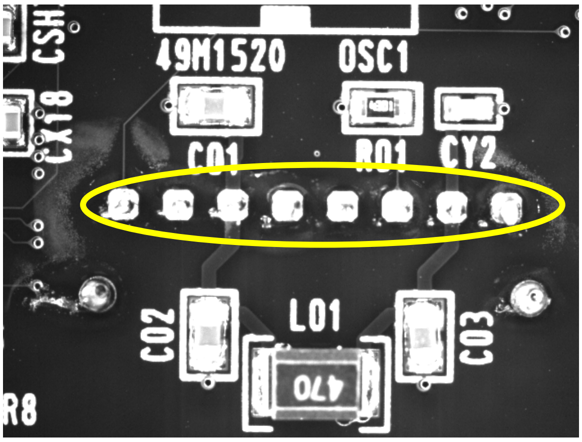


Fig. 5.1.1 (a)

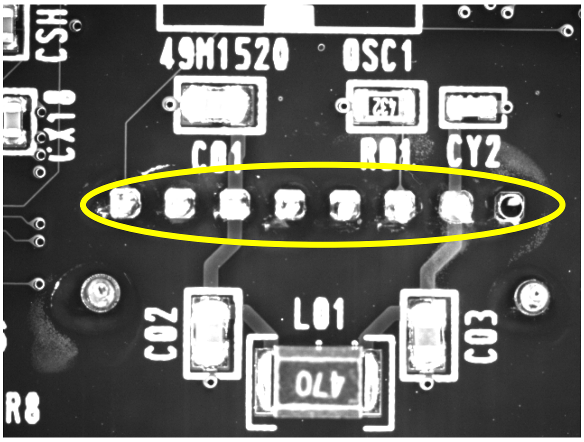
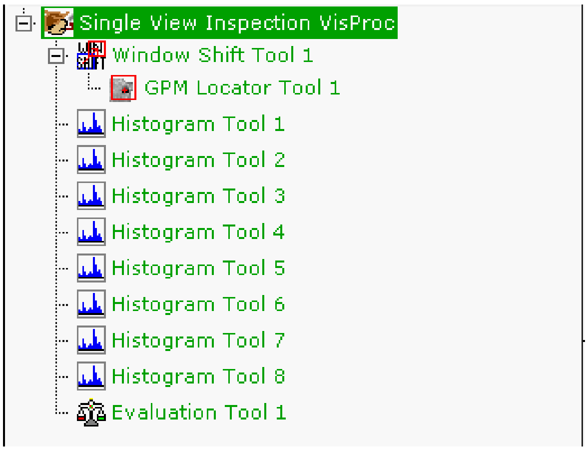


Fig. 5.1.1 (b)

**Study for application**

Consider which command tools are applicable for this application. In this example, using the characteristic that a through hole looks mostly white when soldered and relatively black when not soldered, histogram tools will be used to measure the brightness and an evaluation tool to evaluate the final PASS/FAIL. Since the printed circuit boards are not fixed, it is necessary to shift the measurement area of each histogram tool to the proper position during the execution of image processing. To carry out this inspection, a vision process having the following structure is created:



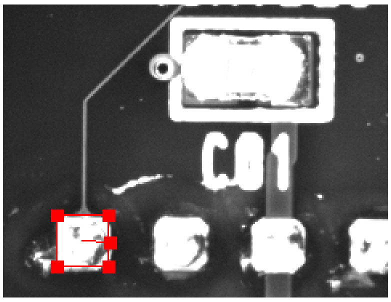

Use “Histogram Tool 1” to “Histogram Tool 8” to inspect the brightness of each of the eight through holes. Since the printed circuit boards are not fixed in position, use “Window Shift Tool 1” to detect the location of each printed circuit board. The measurement windows of “Histogram Tool 1” to “Histogram Tool 8”, which are placed under “Window Shift Tool 1” on the tree view, are shifted according to the detection result of “GPM Locator Tool 1”, which is a child tool of “Window Shift Tool 1”. Thus, “Histogram Tool 1” to “Histogram Tool 8” can correctly measure the brightness of the through holes even when the location of the printed circuit board is shifted. Use the last evaluation tool to evaluate the brightness measured by “Histogram Tool 1” to “Histogram Tool 8” for the PASS/FAIL evaluation.

## Train GPM Locator Tool 1

A feature on the printed circuit board of which the relative position from each through hole to be inspected is fixed is appropriate as the model of “GPM Locator Tool 1”. For example, teach the silk printing of the printed circuit board as the model, as shown in the figure below.


## Train Histogram 1 ~ 8

Set the measurement area of “Histogram Tool 1” on the leftmost through hole where solder is to be applied, as shown in the figure below. Likewise, set the measurement areas of “Histogram Tool 2” to “Histogram Tool 8”on the through holes sequentially from left to right in the same way.




## Train Evaluation Tool 1

In “Evaluation Tool 1”, the PASS/FAIL evaluation conditions will be specified. In inspection, it is important to set appropriate criteria for the PASS/FAIL evaluation. In this example, the histogram tool measurement results of through holes with and without solder are compared to determine the criterion.

**Fig. 5.1.1 (c)**

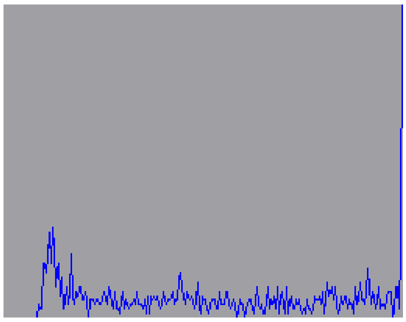
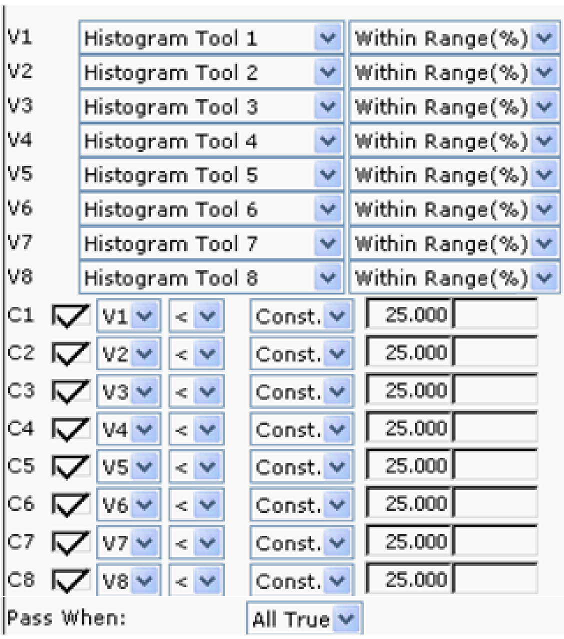


**Fig. 5.1.1 (d)**

Fig. 5.1.1 (c) shows the distribution of the brightness of a soldered through hole. Fig. 5.1.1 (d) shows the distribution of the brightness of a through hole with insufficient solder. These figures show that the results are concentrated in bright values in the distribution chart for the soldered through hole, and they are distributed in a wide range in the distribution chart for the through hole with insufficient solder. From these distribution charts, you will find that the percentage of dark pixels can probably be compared to determine whether the solder is applied. With *i*RVision, the brightness of a pixel is indicated with a numeric value between 0 and 255, and the value at the center of the histogram chart is 128. Therefore, a condition is set so that the target passes inspection when the percentage of pixels with brightness values between 0 and 128 does not exceed a certain threshold. For the eight histogram tools, set 0 to 128 for [Range of Interest] to calculate the percentage of pixels with brightness values within a range between 0 and 128.

Run a test for multiple printed circuit boards using the histogram tools, to measure the percentage of pixels whose brightness values are between 0 and 128, for both good and bad soldered through holes. In this example, it was found that the percentage of pixels within the range is 22% or less for soldered through holes and 40% or more for through holes with insufficient solder. From this data, around 31%, which is the middle value of 22% and 40%, seemed to be a decent threshold. However, a value of 25% was chosen to add a little margin so that through holes with questionable amount of solder would fail the inspection.

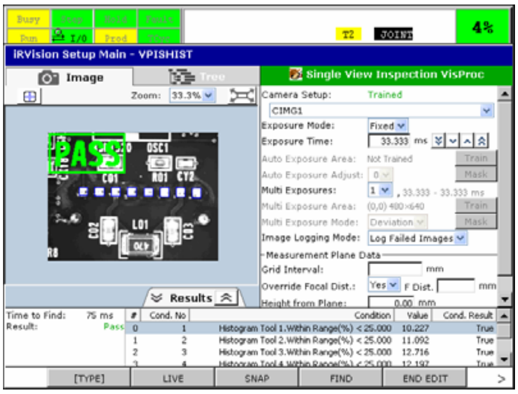
Set the conditions for “Evaluation Tool 1” to output [True] when the percentage of pixels within the range is 25 or less for each histogram tool. Select [All True] for the final evaluation condition of the evaluation tool because the inspection should output PASS when the measurements for all eight through holes meet the set conditions.



## Running a test

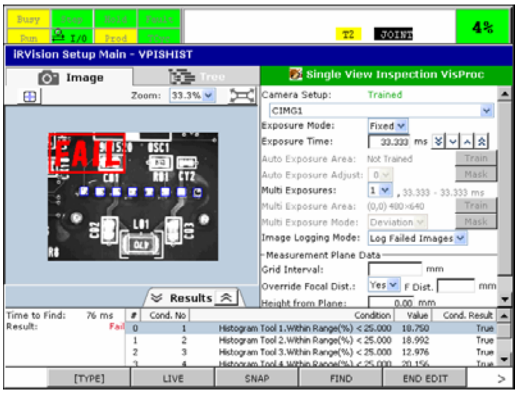
Run the vision process and check that the result is correct.

For the printed circuit board on which solder is applied properly to all through holes, the eight conditions are all [True]. Since [All True] is selected for [Pass When], the target passed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Histogram Tool 1.Within Range(%) < 25.000 | 10.227 | True |
| 1 | 2 | Histogram Tool 2.Within Range(%) < 25.000 | 11.092 | True |
| 2 | 3 | Histogram Tool 3.Within Range(%) < 25.000 | 12.716 | True |
| 3 | 4 | Histogram Tool 4.Within Range(%) < 25.000 | 12.197 | True |
| 4 | 5 | Histogram Tool 5.Within Range(%) < 25.000 | 9.269 | True |
| 5 | 6 | Histogram Tool 6.Within Range(%) < 25.000 | 8.739 | True |
| 6 | 7 | Histogram Tool 7.Within Range(%) < 25.000 | 13.939 | True |
| 7 | 8 | Histogram Tool 8.Within Range(%) < 25.000 | 11.837 | True |

For the printed circuit board on which solder is not applied properly to the rightmost through hole, [False] is output for [Cond. 8] because “Histogram Tool 8” measures the percentage of pixels within the range to be 52.571. Since [All True] is selected for [Pass When], the target failed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Histogram Tool 1.Within Range(%) < 25.000 | 18.750 | True |
| 1 | 2 | Histogram Tool 2.Within Range(%) < 25.000 | 18.992 | True |
| 2 | 3 | Histogram Tool 3.Within Range(%) < 25.000 | 12.976 | True |
| 3 | 4 | Histogram Tool 4.Within Range(%) < 25.000 | 20.156 | True |
| 4 | 5 | Histogram Tool 5.Within Range(%) < 25.000 | 23.173 | True |
| 5 | 6 | Histogram Tool 6.Within Range(%) < 25.000 | 15.126 | True |
| 6 | 7 | Histogram Tool 7.Within Range(%) < 25.000 | 13.074 | True |
| 7 | 8 | Histogram Tool 8.Within Range(%) < 25.000 | 52.571 | False |

If a non-defective solder fails the inspection or a defective solder passes the inspection, adjust the evaluation condition set in the evaluation tool. In this example, the percentage of values within a range that is obtained using each histogram tool can probably be evaluated to perform appropriate inspection. If it is difficult to make a PASS/FAIL evaluation with only one measurement value, set a condition that logically evaluates multiple measurement values. The following table lists various types of measurement results of the histogram tools for through holes with and without solder. According to the data in this table, Median and Mean values can probably be used as evaluation criteria because there are clear differences in these values.

| Result | Num. Pixels | Maximum | Minimum | Median | Mode | Mean | Std. Dev. | Within Range | Out of Range |
|---|---|---|---|---|---|---|---|---|---|
| PASS | 1089 | 255 | 34 | 255 | 255 | 222.3 | 57.5 | 11.3 | 88.7 |
| FAIL | 1089 | 255 | 22 | 127 | 255 | 135.4 | 78.3 | 51.8 | 48.2 |

If measurement values with only one command tool do not allow stable evaluation, it may be necessary to add other command tools.

## 5.1.2 Inspecting Whether There Is a Thread

This application inspects whether a hole on a plate (shown in the figures below) is threaded. Fig. 5.1.2 (a) shows a threaded hole, which should pass the inspection. Fig. 5.1.2 (b) shows an unthreaded hole, which should fail the inspection.

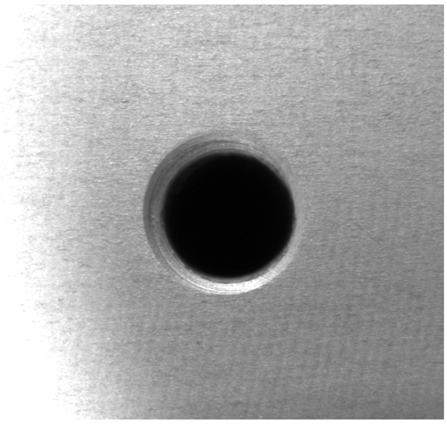


**Fig. 5.1.2 (a)**

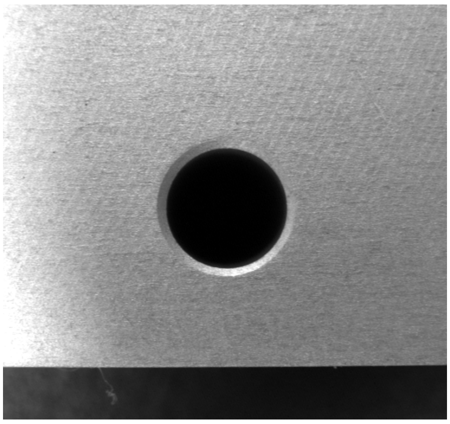


**Fig. 5.1.2 (b)**

### Study for application

Consider which command tools are applicable for this application. In this example, it is difficult to determine whether each hole is threaded from the images in Figs. 5.1.2 (a) and (b). To solve this problem, the camera is mounted such that the optical axis is at an angle with respect to the plate plane, so that an image as shown in the figure below is captured. The holes enclosed with a circle and a square in the figure below are to be inspected. For this application, the target would pass the inspection if both holes were threaded.

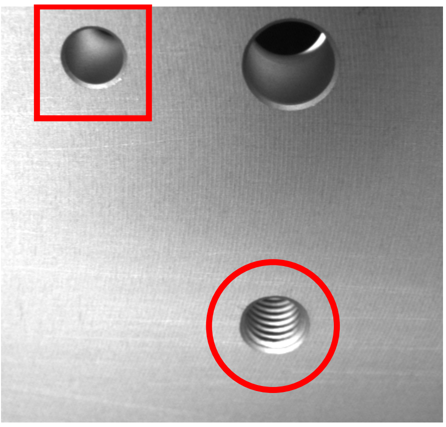
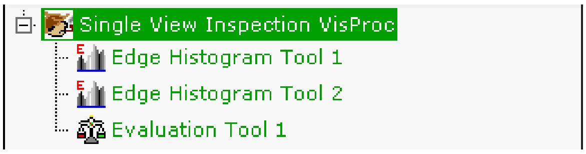


**Fig. 5.1.2 (c)**

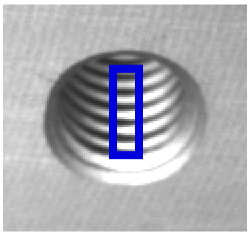
From Fig. 5.1.2 (c), you will see that horizontal stripes appear inside the circled hole, which is threaded, and a relatively even surface appears inside the boxed hole, which is not threaded. An inspection seems possible by measuring the directions of the gradient with an edge histogram tool. To carry out this inspection, a vision process having the following structure is created:

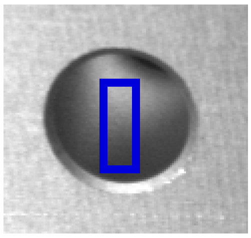


Use “Edge Histogram Tool 1” to measure the circled hole in Fig. 5.1.2 (c) and “Edge Histogram Tool 2” to measure the boxed hole in Fig. 5.1.2 (c). Use “Evaluation Tool 1” to evaluate these measurement results.

## Train Edge Histogram 1 ~ 2

Set the measurement areas of “Edge Histogram Tool 1” and “Edge Histogram Tool 2” as shown in the figures below. The measurement areas do not enclose the entire hole because the directions of the gradient tend to be constant in a narrower measurement area since the face of a hole is curved.





Before training the evaluation tool, the PASS/FAIL evaluation conditions need to be determined. For this example, the directions of gradient are expected to concentrate in upward and downward directions since the thread in the snapped image looks like a horizontal stripe pattern. Select [Direction] for [Mode], and [Upward] for [Center of histogram means] in the edge histogram tools setup screen. The results of measurements with these settings display the distribution charts shown below.

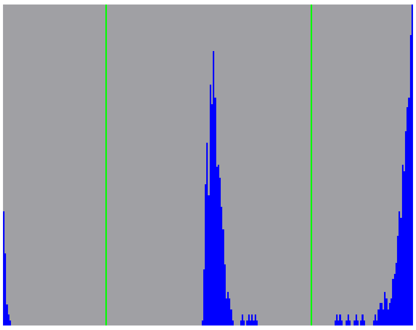


**Fig. 5.1.2 (d)**

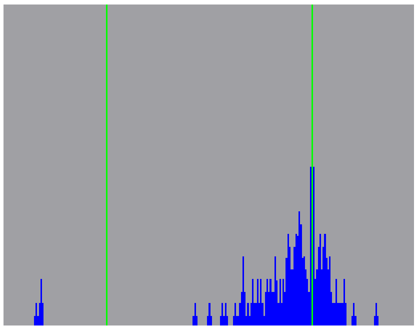
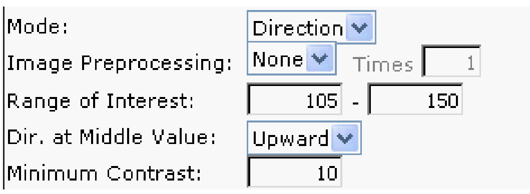


**Fig. 5.1.2 (e)**

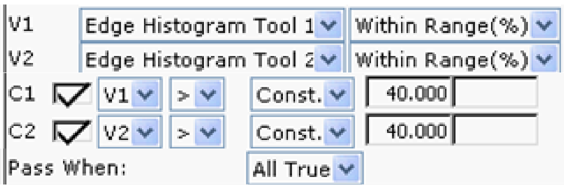
Fig. 5.1.2 (d) shows the distribution of the gradient directions for the threaded hole and Fig. 5.1.2 (e) shows the distribution for the non-threaded hole. As expected, a peak appears around the center of the distribution chart for the threaded hole. Since the center value of the distribution chart indicating a peak in Fig. 5.1.2 (d) is 128, set a range including this peak for [Range of Interest]. In this example, a range of 105 to 150 for [Range of Interest] is specified. It is necessary to run a test for many targets later to check whether these numeric values are appropriate, and to adjust them if they are inappropriate.




## Train Evaluation Tool 1

Set conditions for “Evaluation Tool 1” to output [True] when the percentage of pixels within the range that is checked by each edge histogram tool is above a specified value. The results of running tests on multiple targets show that the percentage of pixels within the range is 46% or higher for threaded holes and 10% or lower for non-threaded holes. From this data, a value between 10 and 46 seems appropriate as the threshold used to determine whether a target passes or fails. In this example, a value of 40% was chosen to add a little margin so that a potentially faulty target would fail the inspection. Finally, select

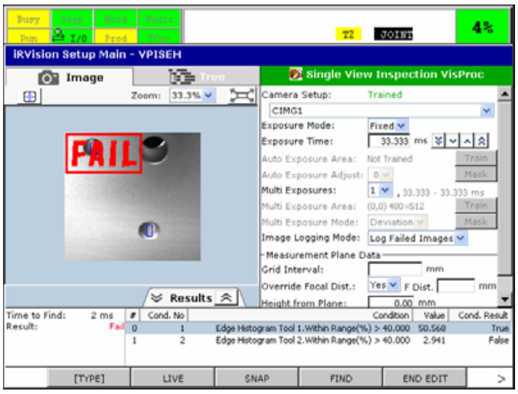
[All True] for [Pass When] since the condition required to determine that the target passes the inspection is that both holes are threaded.




## Running a test

Run the vision process and check that the result is correct.

In the following figure, the hole at the lower right is threaded, but the hole at the upper left is not threaded. The target failed inspection because [False] is output for [Cond. 2] which is the evaluation of the measurement value of “Edge Histogram Tool 2”, and [All True] is selected for [Pass When].




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Edge Histogram Tool 1.Within Range(%) > 40.000 | 50.568 | True |
| 1 | 2 | Edge Histogram Tool 2.Within Range(%) > 40.000 | 2.941 | False |

# 5.2 LOCATION INSPECTION

This section introduces sample applications for inspecting target locations. The following methods are available as main techniques to determine locations:

- Detect a taught model using a GPM locator tool.
- Detect a taught feature using a blob locator tool
- Measure a length using an edge pair locator tool.

## 5.2.1 Inspecting the Location Where a Label Is Attached

This application inspects whether a label is attached in place on a connector. Fig. 5.2.1 (a) shows a connector on which a label is attached in place, which should pass the inspection. Fig. 5.2.1 (b) shows a connector on which a label is not attached in place, which should fail the inspection. Assume that the connectors are fixed in position. When the lengths of the four-circled locations in Fig. 5.2.1 (a) satisfy the following conditions, the label is considered to be attached in place:

- The lengths of measurement sections <1> and <2> are 1 ±0.2 mm or the difference between the lengths of measurement sections <1> and <2> is within 0.3 mm.
- The lengths of measurement sections <3> and <4> are 2 mm or less.

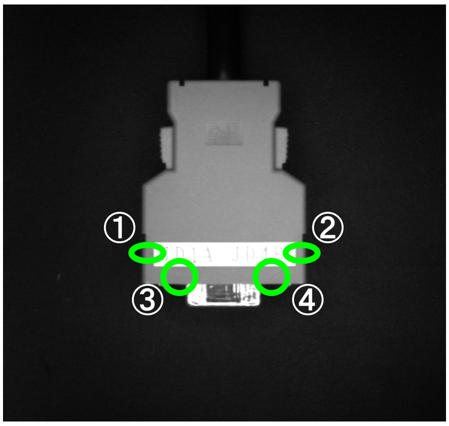



Fig. 5.2.1 (a)

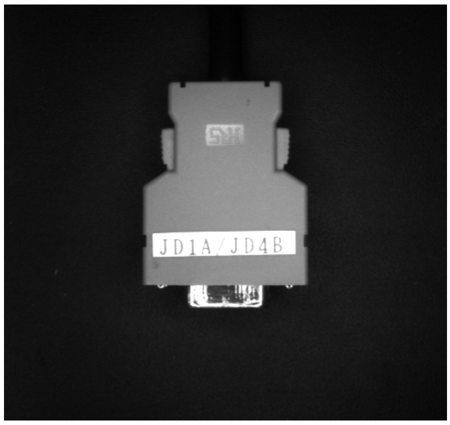
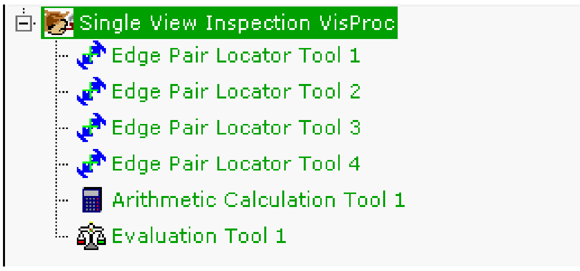



Fig. 5.2.1 (b)

## Study for application

Consider which command tools are applicable for this application. This application inspects whether a label is attached in place by checking the lengths from the perimeter of the connector. First, edge pair locator tools can be used to measure the length of each section. Then, arithmetic calculation tools can be used to compare the lengths of measurement edge pairs. To carry out this inspection, a vision process having the following structure is created:




Use "Edge Pair Locator Tool 1" to "Edge Pair Locator Tool 4" to measure the lengths of four sections. Use "Arithmetic Calculation Tool 1" and "Arithmetic Calculation Tool 2" to calculate the difference between the results of edge pair locator tools. Since it is necessary to evaluate the lengths in millimeters, measurement plane information is set in "Single View Inspection VisProc".

## Train Edge Pair Locator Tool 1 ~ 4

Teach "Edge Pair Locator Tool 1" to "Edge Pair Locator Tool 4" to measure the length of each of measurement sections <1> to <4> in Fig. 5.2.1 (a). Set the measurement area for each edge pair locator tool a little wider so that the length can be measured even if the label is slightly shifted. In addition, specify edge-pair lengths that can be measured in this measurement area for [Maximum Length to Search] and [Minimum Length to Search]. For a connector on which no edge pair is detected within this threshold, UNDETERMINED is output as the inspection result, and should be handled as a case where a label itself is not attached.

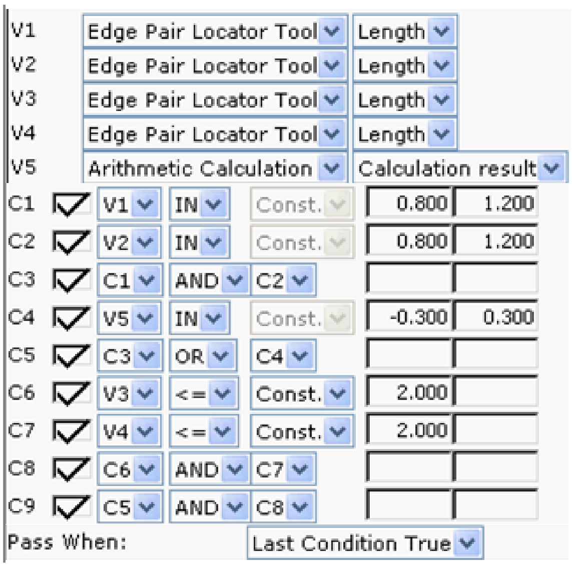
## Train Arithmetic Calculation Tool 1

Train "Arithmetic Calculation Tool 1" to calculate the difference between the lengths measured by "Edge Pair Locator Tool 1" and "Edge Pair Locator Tool 2".

## Train Evaluation Tool 1

Since the conditions for the inspection to output PASS are specified for this application, teach "Evaluation Tool 1" to output PASS when the specified conditions are satisfied. To evaluate the measurement values for this inspection, a logical operation must be performed. This is because when the four evaluation conditions are defined as follows and if "(A OR B) AND (C)" is satisfied, the target passes the inspection:

A: The lengths of measurement sections <1> and <2> are 1 ±0.2 mm. ([Cond. 1] to [Cond. 3], See below)
B: The difference between the lengths of measurement sections <1> and <2> is within 0.3 mm. ([Cond. 4])
C: The lengths of measurement sections <3> and <4> are 2 mm or less. ([Cond. 6] to [Cond. 8])

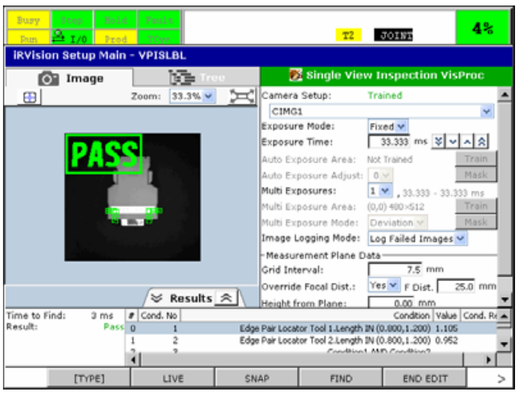



[Cond. 5] evaluates “A OR B”, [Cond. 8] evaluates “C”, and [Cond. 9] finally evaluates “(A OR B) AND (C)”. The inspection should output PASS when the result of the logical operation for [Cond. 9] is [True]. Thus, select [Last Condition True] for [Pass When] in the evaluation tool.

## Running a test

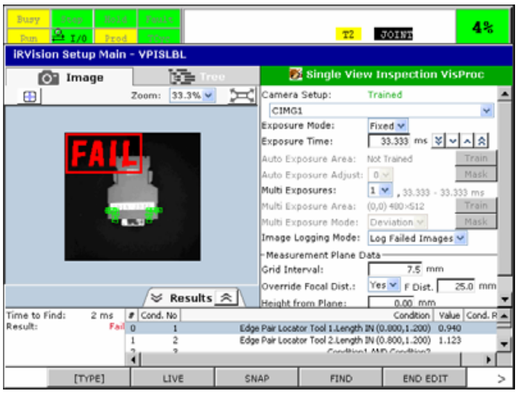
Run the vision process and check that the result is correct.

For the connector in Fig. 5.2.1 (a), condition A, B and C are [True]. Since “A OR B” is [True], and “C” is also [True], “(A OR B) AND (C)” is also [True]. Since [Last Condition True] is selected for [Pass When], the target passed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Edge Pair Locator Tool 1.Length IN (0.800,1.200) | 1.105 | True |
| 1 | 2 | Edge Pair Locator Tool 2.Length IN (0.800,1.200) | 0.952 | True |
| 2 | 3 | Condition1 AND Condition2 | | True |
| 3 | 4 | Arithmetic Calculation Tool 1.Calculation result IN (-0.300,0.300) | 0.152 | True |
| 4 | 5 | Condition3 OR Condition4 | | True |
| 5 | 6 | Edge Pair Locator Tool 3.Length <= 2.000 | 1.796 | True |
| 6 | 7 | Edge Pair Locator Tool 4.Length <= 2.000 | 1.723 | True |
| 7 | 8 | Condition6 AND Condition7 | | True |
| 8 | 9 | Condition5 AND Condition8 | | True |

For the connector in Fig. 5.2.1 (b), all measurement values other than 1 to 5 do not satisfy their respective conditions. “A OR B” is [True] but “C” is [False], therefore “(A OR B) AND (C)” is [False]. Since [Last Condition True] is selected for [Pass When], the target failed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Edge Pair Locator Tool 1.Length IN (0.800,1.200) | 0.940 | True |
| 1 | 2 | Edge Pair Locator Tool 2.Length IN (0.800,1.200) | 1.123 | True |
| 2 | 3 | Condition1 AND Condition2 | | True |
| 3 | 4 | Arithmetic Calculation Tool 1.Calculation result IN (-0.300,0.300) | -0.182 | True |
| 4 | 5 | Condition3 OR Condition4 | | True |
| 5 | 6 | Edge Pair Locator Tool 3.Length <= 2.000 | 2.748 | False |
| 6 | 7 | Edge Pair Locator Tool 4.Length <= 2.000 | 2.987 | False |
| 7 | 8 | Condition6 AND Condition7 | | False |
| 8 | 9 | Condition5 AND Condition8 | | False |

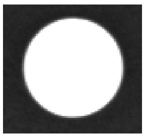
# 5.3 NUMBER INSPECTIONS

This section introduces sample applications for inspecting a number of targets found. The following method is available as a technique to determine a number of targets found:

- Count the number of targets detected with locator tools using a count tool.

## 5.3.1 Counting the Number of Targets in an Image

This application checks that there are five tablets as shown below in the image. Assume that these tablets all have the same shape and size.



Consider which command tools are applicable for this application. In this example, the number of tablets having a specified shape is counted, thus a GPM locator tool can be used to detect the tablets. Then, a count tool can be used to count the number of detected tablets. To carry out this inspection, a vision process having the following structure is created:


### Train GPM Locator Tool 1

Teach a “GPM Locator Tool 1” model with a reference target and adjust the parameters so that the command tool detects the target correctly. Since all tablets have a circular shape and the same size, uncheck the [Enabled] check boxes in [Degree of Freedom].

| DOF | Enabled | Nom. | Min. | Max. | |
|---|---|---|---|---|---|
| Orientation: | ☐ | 0.0 | -180.0 | 180.0 | ° |
| Scale: | ☐ | 100.0 | 90.0 | 110.0 | % |
| Aspect: | ☐ | 100.0 | 90.0 | 100.0 | % |

### Train Count 1

No setup is required for “Count Tool 1” because a count tool by default counts the total number of items detected by all sibling locator tools that are listed before it on the tree view.

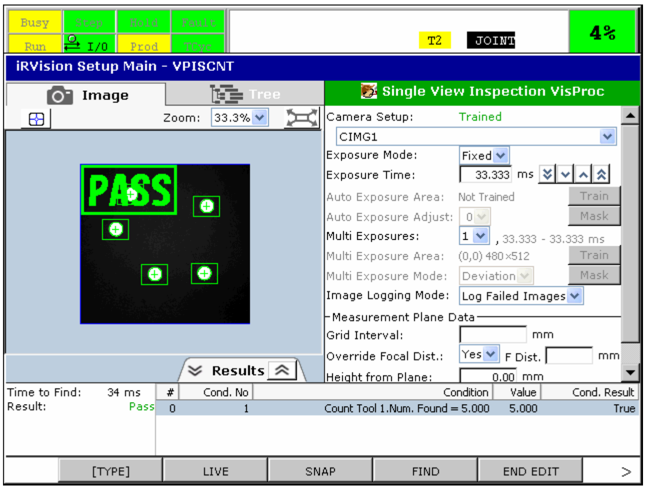
## Train Evaluation Tool 1

Set a condition for “Evaluation Tool 1” to output [True] when [Num. Found] of “Count Tool 1” is 5. Select [All True] for [Pass When] since only one condition is set.

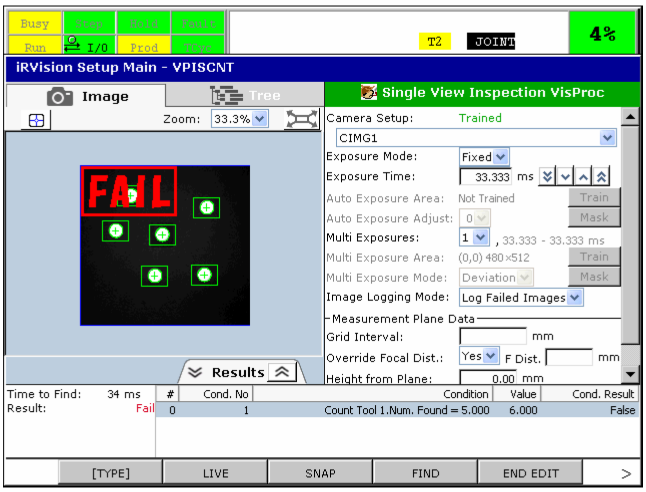

## Running a test

Run the vision process and check that the result is correct.

For the following image, five features matching the model taught for the GPM locator tool are detected, and the condition is evaluated as [True]. Since [All True] is selected for [Pass When], the target passed the inspection.

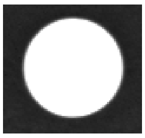



For the following image, six features matching the model taught for the GPM locator tool are detected, and the condition is evaluated as [False]. Since [All True] is selected for [Pass When], the target failed the inspection.



## 5.3.2 Detecting Defects and Foreign Objects and Counting the Number of Targets

This application checks that there are five tablets as shown below in the image. This application inspects not only whether there are five tablets, but also whether any tablet is defective and whether any foreign object is contained.



### Study for application

Consider which command tools are applicable for this application. When the vision process created in Subsection 5.3.1, "Counting the Number of Targets in an Image" was used to inspect images which contain a defect or a foreign object as shown in Figs. 5.3.2 (a) and (b) both images passed the inspection.

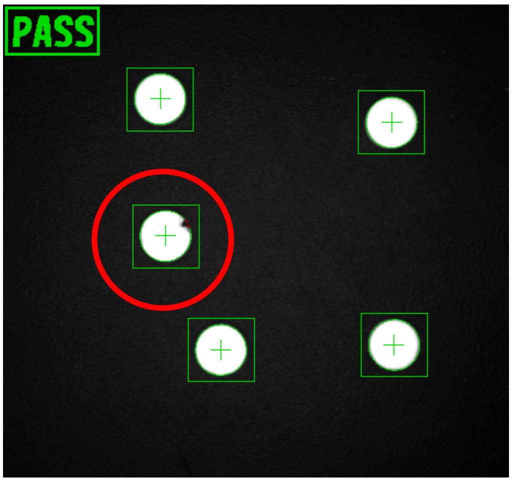



**Fig. 5.3.2 (a)**

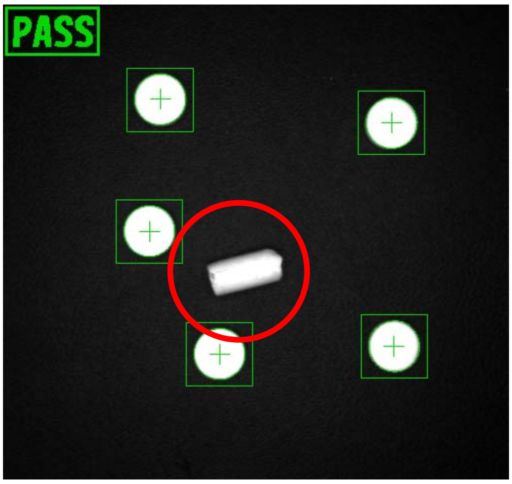



**Fig. 5.3.2 (b)**

In this example, it is necessary to check whether the detected targets satisfy evaluation conditions and whether foreign objects are present. To carry out this inspection, a vision process having the following structure is created:


A blob locator tool can detect objects with different geometric features. For this reason, a blob locator tool is appropriate for detecting objects in the image in this case. Use "Blob Locator Tool 1" to detect the locations of objects and use "GPM Locator Tool 1" to verify that each detected object matches with the model. By checking the shape using "GPM Locator Tool 1", defective targets and foreign object can be omitted. Use "Evaluation Tool 2" to evaluate whether the objects found with "Blob Locator Tool 1" should be counted toward the number of targets passed. Then, use "Count Tool 1" to count the total number of objects and targets passed in the image. Only when the total number of objects and the number of targets passed are both 5, the inspection should output PASS.

## Train Blob Locator Tool 1

Teach a blob locator tool to detect all blobs having at least a certain area. When a tablet is taught as the model and is detected, its area is about 2800 pixels. In this example, since there is little noise in the background of the image, only the blobs having an area of 5 pixels or less are excluded as noise, and all the other blobs are detected. The [Minimum] of [Area] is set to 5, and [Maximum] of [Area] is set to the area of the entire image (307200 = 640 × 480).

| | |
|---|---|
| Calculate the angle: | ☐ |
| Angle Calc. Method: | Axes of Inertia |
| Find if Touching Win.: | ☐ |

| DOF | Enabled | Min. | Max. | |
|---|---|---|---|---|
| Area: | ☑ | 5 | 307200 | pix |
| Perimeter: | ☐ | 107 | 161 | pix |
| Circularity: | ☐ | 0.760 | 1.140 | |
| Semi Major: | ☐ | 16.7 | 25.1 | pix |
| Semi Minor: | ☐ | 16.6 | 24.8 | pix |
| Elongation: | ☐ | 0.807 | 1.211 | |

## Train GPM Locator Tool 1

Teach a "GPM Locator Tool 1" model with a reference target and adjust the parameters so that the command tool detects the target correctly. Since all tablets have a circular shape and the same size, uncheck the [Enabled] check boxes in [Degree of Freedom].

| DOF | Enabled | Nom. | Min. | Max. | |
|---|---|---|---|---|---|
| Orientation: | ☐ | 0.0 | -5.0 | 5.0 | ° |
| Scale: | ☐ | 100.0 | 95.0 | 130.0 | % |
| Aspect: | ☐ | 100.0 | 95.0 | 100.0 | % |

## Train Evaluation Tool 2

Set a condition to evaluate whether the objects found with "Blob Locator Tool 1" match the "GPM Locator Tool 1" model. When a test is run for the sample shown in Fig. 5.3.2 (a) using "GPM Locator Tool 1", it is found that the score value for a defective target is lower than that of the other targets as shown in the figure below:

| # | Row(V) | Column(H) | Score | Contrast | Fit Error |
|---|---|---|---|---|---|
| 1 | 345.5 | 393.4 | 100.0 | 143.1 | 0.117 |
| 2 | 119.9 | 390.7 | 99.9 | 147.5 | 0.275 |
| 3 | 96.1 | 156.5 | 99.9 | 155.6 | 0.267 |
| 4 | 350.7 | 218.1 | 99.8 | 160.5 | 0.284 |
| 5 | 235.2 | 161.9 | 95.1 | 168.3 | 0.374 |

Therefore, the score output by "GPM Locator Tool 1" seems like a good choice as the criterion for eliminating a defective tablet. In this test, while the detected scores of features with no defect were 99.8 or more, the detected score of a feature with a defect was 95.1. Accordingly, the threshold for distinguishing between non-defective and defective targets should be set to a value between 95.1 and 99.8. The threshold for distinguishing between non-defective and defective targets is determined by increasing the number of test samples. In this example, the test was conducted for 20 defective samples. Then, the threshold score was set to 96.5, which was obtained by adding a margin to the score of the highest defective sample score.

> **TIP**
> The score may vary depending on the lighting environment, location of the target in the image, and other factors even when the same target is detected. Set a value with which to clearly distinguish between defective and non-defective targets for the score threshold.

Train "Evaluation Tool 2" to output PASS when the [Score] value of "GPM Locator Tool 1" is above a threshold. For this example, the PASS/FAIL threshold is set to 96.5 based on the result of the tests. Select [All True] for [Pass When] since only one condition is set.


With this setting, each object detected by "Blob Locator Tool 1" that has a shape matching the tablet model taught for "GPM Locator Tool 1" by at least 96.5% is considered as a non-defective target.

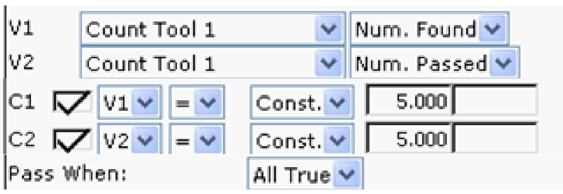
When a foreign object is contained as shown in Fig. 5.3.2 (b), the result of Cond.1 is not evaluated because GPM Locator Tool 1 outputs Not Found for the foreign object. Since [All True] is selected for [Pass When], the judgment of "Evaluation Tool 2" is UNDETERMINED if any condition is not evaluated. Accordingly, the foreign object is not counted as a non-defective tablet (it is also not counted as a defective tablet).

## Train Count Tool 1

"Count Tool 1" needs to count the number of objects detected by "Blob Locator Tool 1". Leave "Count Tool 1" at a default setting since a count tool counts the total number of items detected by all sibling locator tools that appear before it on the tree view.

## Train Evaluation Tool 1

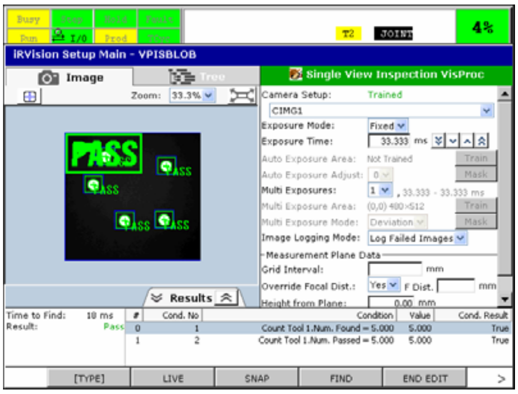
Set conditions for "Evaluation Tool 1" to output [True] when the [Num. Passed] and [Num. Found] values are both 5. Select [All True] for [Pass When] since both conditions must be [True] for the inspection to pass.




## Running a test

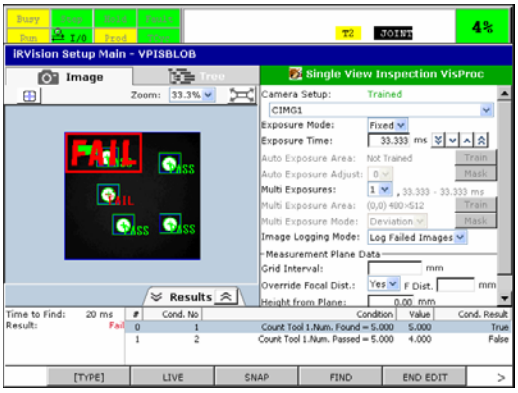
Run the vision process and check that the result is correct.

For the following image, the [Num. Found] and [Num. Passed] conditions are both [True]. Since [All True] is selected for [Pass When], the target passed the inspection.



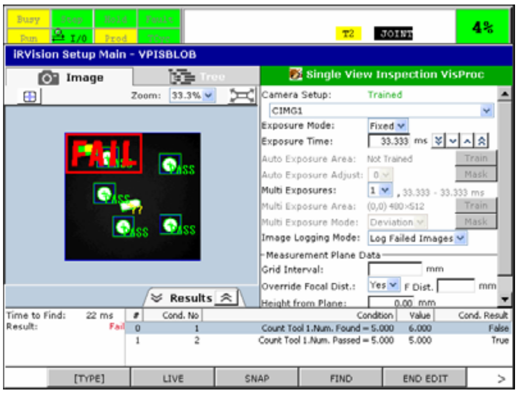

| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Count Tool 1.Num. Found = 5.000 | 5.000 | True |
| 1 | 2 | Count Tool 1.Num. Passed = 5.000 | 5.000 | True |

For the image in Fig. 5.3.2 (a), condition 1 is [True], but condition 2 is [False] because one of the detected targets is defective. Since [All True] is selected for [Pass When], the target failed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Count Tool 1.Num. Found = 5.000 | 5.000 | True |
| 1 | 2 | Count Tool 1.Num. Passed = 5.000 | 4.000 | False |

For the image in Fig. 5.3.2 (b), condition 2 is [True], but condition 1 is [False] because one foreign object is detected and the [Num. Found] value is 6. Since [All True] is selected for [Pass When], the target failed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Count Tool 1.Num. Found = 5.000 | 6.000 | False |
| 1 | 2 | Count Tool 1.Num. Passed = 5.000 | 5.000 | True |

**⚠ CAUTION**

To detect any foreign object correctly, the difference between reference targets and foreign objects to be excluded must clearly appear in the image. Whether a defective product can be detected depends on the size of the defective section and resolution of the image.

## 5.3.3 Inspecting Whether Products with Different Specifications Are Mixed

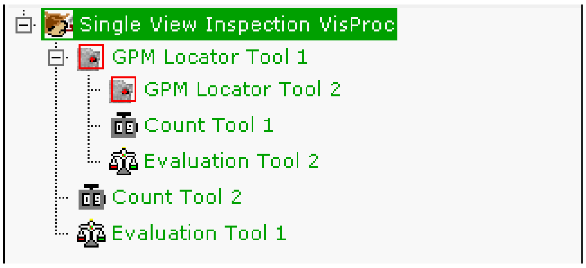
This application inspects whether different models of compact flash cards are mixed. Fig. 5.3.3 (a) shows an image of two compact flash cards of the same model, which should pass the inspection. Fig. 5.3.3 (b) shows an image including one compact flash card with a different specification, which should fail the inspection. Assume that these compact flash cards are not fixed in position and the number of cards in each snapped image is unknown.


**Fig. 5.3.3 (a)**


**Fig. 5.3.3 (b)**

### Study for application

Consider which command tools are applicable for this application. In this example, the targets can be detected using a GPM locator tool because the targets having the same outline are to be detected. Also, it is necessary to check the model of each detected target. To check the model, a GPM locator tool can be used to check whether there is a logo "128MB" in place. To carry out this inspection, a vision process having the following structure is created:




### Train GPM Locator Tool 1

A model for "GPM Locator Tool 1" should be taught such that the command tool can detect different models of the same shape. In this example, the outline of a target is taught as a model, as shown in Fig. 5.3.3 (c).

### Train GPM Locator Tool 2

Teach the feature specific to the expected model as the model pattern for "GPM Locator Tool 2". For this example, teach the tool logo "128MB" as shown in Fig. 5.3.3 (d). Set the area enclosed by a rectangle in Fig. 5.3.3 (c) as the search window. The search window of "GPM Locator Tool 2" is dynamically shifted according to the detection result of "GPM Locator Tool 1". That is, "GPM Locator Tool 2" always detects the target in the same area with respect to the outline of each target detected by "GPM Locator Tool 1".

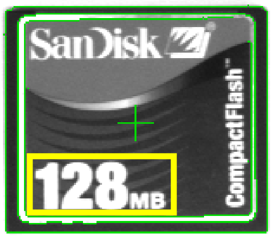



Fig. 5.3.3 (c)


Fig. 5.3.3 (d)

## Train Count 1

"Count Tool 1" should count the number of targets detected by "GPM Locator Tool 2". Leave "Count Tool 1" at a default setting since a count tool counts the total number of items detected by all sibling locator tools that appear before it on the tree view.

## Train Evaluation Tool 2

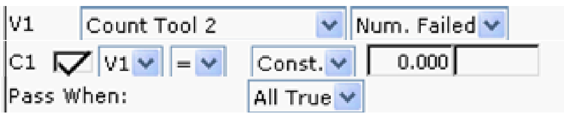
Set a condition for "Evaluation Tool 2" to output [True] when [Num. Found] of "Count Tool 1" is 1. Select [All True] for [Pass When] since only one condition is set.


## Train Count 2

"Count Tool 2" should count the number of targets detected by "GPM Locator Tool 1". Leave "Count Tool 2" at a default setting since a count tool counts the total number of items detected by all sibling locator tools that appear before it on the tree view.

## Train Evaluation Tool 1

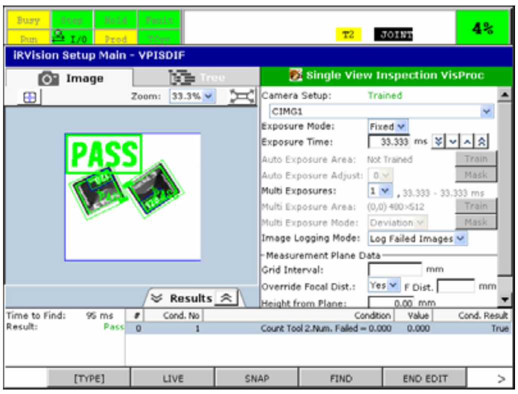
Set a condition for "Evaluation Tool 1" to output [True] when [Num. Failed] of "Count Tool 2" is 0. Select [All True] for [Pass When] since only one condition is set.




## Running a test

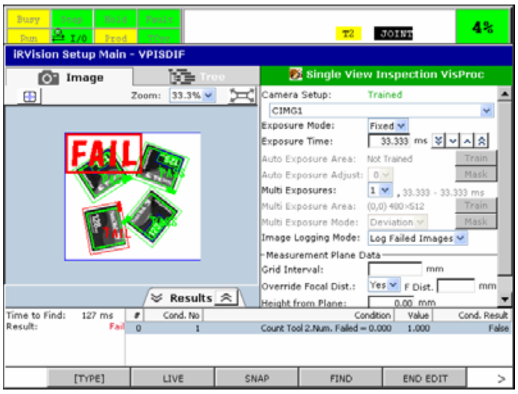
Run the vision process and check that the result is correct.

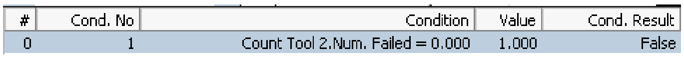
For the following image, the condition is [True] because none of the detected feature is evaluated as [Fail] with "Count Tool 2". Since [All True] is selected for [Pass When], the target passed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Count Tool 2.Num. Failed = 0.000 | 0.000 | True |

For the following image, the condition is [False] because one detected feature is evaluated as [Fail] with “Count Tool 2”. Since [All True] is selected for [Pass When], the target failed the inspection.



| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Count Tool 2.Num. Failed = 0.000 | 1.000 | False |

For this example, only one child tool is used to determine the model. Multiple command tools can be added as child tools of “GPM Locator Tool 1” as required to comprehensively evaluate the measurement results of them, which would allow a more detailed inspection.

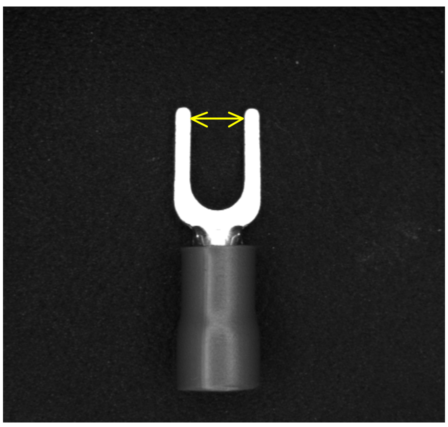
# 5.4 LENGTH/WIDTH INSPECTIONS

This section introduces sample applications for inspecting lengths and widths. The following methods are available as main techniques to determine lengths:

- Measure a length using an edge pair locator tool.
- Measure an interval by combining an edge pair locator tool and a statistics calculation tool.
- Calculate the distance between locations detected by multiple locator tools, using a geometric calculation tool.
- Calculate the horizontal/vertical distance between locations detected by multiple locator tools using an arithmetic calculation tool.

## 5.4.1 Inspecting the Width

In this example, the width at the tip of the crimp terminal, as shown in the figure below, is inspected. The width to be inspected is indicated with an arrow. When the width is within the specified range, the inspection vision process outputs PASS; when it is outside the range, the vision process outputs FAIL. Assume that the crimp terminal is not fixed, but the cylindrical section does not rotate about its axis.



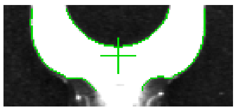
## Study for application

Consider which command tools are applicable for this application. In this example, the width to be inspected is the length between two straight lines. Therefore, the width can be measured using an edge pair locator tool and evaluated using an evaluation tool for the final PASS/FAIL result. Since the crimp terminal is not fixed in position, the search area of the edge pair locator tool must be shifted to the proper position during the execution of image processing. To carry out this inspection, a vision process having the following structure is created:


"Edge Pair Locator Tool 1" is used to inspect the width at the tip of the crimp terminal. Since the crimp terminal is not positioned, use "Window Shift Tool 1" to detect the location of the crimp terminal first. The search area of "Edge Pair Locator Tool 1", placed below "Window Shift Tool 1" on the tree view, is shifted based on the detection result of "GPM Locator Tool 1", which is a child tool of "Window Shift Tool 1". With this structure, "Edge Pair Locator Tool 1" can correctly measure the length at the tip even when the location of the crimp terminal is shifted. Use the evaluation tool to evaluate the length measured by "Edge Pair Locator Tool 1" to output the final PASS/FAIL result.

## Train GPM Locator Tool 1

A feature having a fixed shape is appropriate as the model of "GPM Locator Tool 1". For example, teach the branch section of the crimp terminal as the model as shown in the figure below.



## Train Edge Pair Locator Tool 1

Determine the PASS/FAIL evaluation conditions. To determine a range of the width of good crimp terminals, prepare many non-defective crimp terminals, measure the width of each crimp terminal with a test run of "Edge Pair Locator Tool 1". Based on the data collected above, set the upper and lower limits of the width at the tip for good crimp terminals. In this example, a crimp terminal with the width at the tip between 40 and 45 pixels should pass the inspection. To also detect defective crimp terminals and measure their width, however, the search area for "Edge Pair Locator Tool 1" is set to be able to measure the crimp terminals with wider tip width. Then, [Maximum Length to Search] and [Minimum Length to Search] for "Edge Pair Locator Tool 1" are set to 120 and 20, respectively. These numeric values are determined based on the edge pair length of a crimp terminal deformed to the maximum degree that can be detected by "GPM Locator Tool 1".

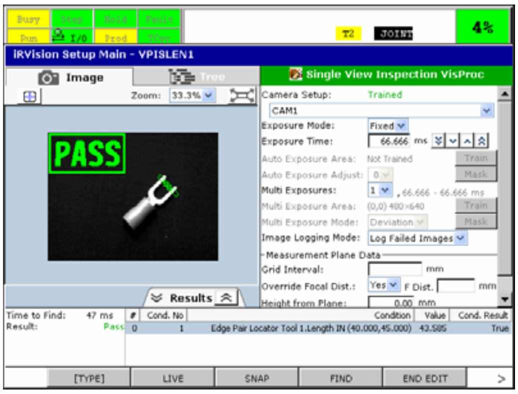
## Train Evaluation Tool 1

A condition for "Evaluation Tool 1" is set to evaluate the [Length] result of "Edge Pair Locator Tool 1" and to output [True] when the edge pair length is between 40 and 45. Select [All True] for [Pass When] since only one condition is set.


## Running a test

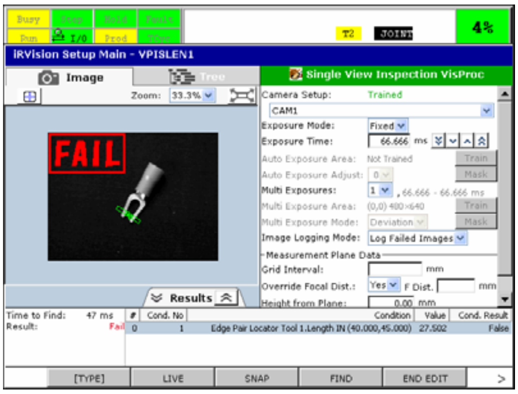
Run the vision process and check that the result is correct.

For the following image, the condition 1 is [True] because the measurement value of the length is 43.585. Since [All True] is selected for [Pass When], the target passed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Edge Pair Locator Tool 1.Length IN (40.000,45.000) | 43.585 | True |

For the following image, the condition 1 is [False] because the measurement value of the length is 27.502. Since [All True] is selected for [Pass When], the target failed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Edge Pair Locator Tool 1.Length IN (40.000,45.000) | 27.502 | False |

If a non-defective crimp terminal is evaluated as FAIL or a defective crimp terminal is evaluated as PASS, adjust the condition set in the evaluation tool.

## 5.4.2 Inspecting the Interval

This application inspects whether the pins of an IC chip as shown in the figure below are arranged at regular intervals. The intervals to be inspected are indicated with arrows. Fig. 5.4.2 (a) shows a chip whose pin intervals are within the allowable range, which should pass the inspection. Fig. 5.4.2 (b) shows a chip having a bent pin, which should fail the inspection. Assume that the IC chip is fixed in position. A non-defective IC chip must satisfy the following conditions:

- The maximum interval is within 2 pixels of the mean interval.
- The minimum interval is within 2 pixels of the mean interval.
- The number of pins is 8.

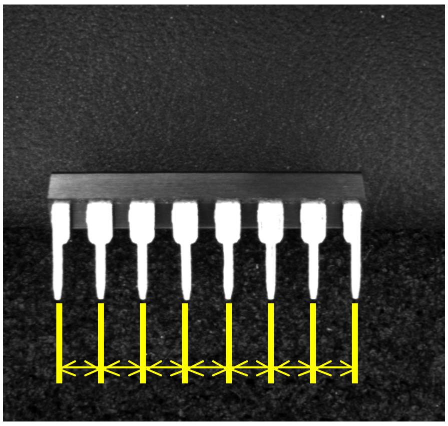


Fig. 5.4.2 (a)

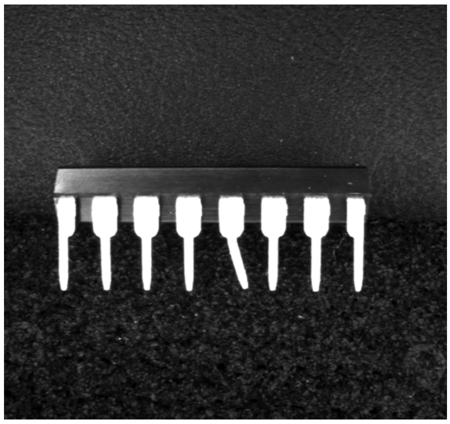
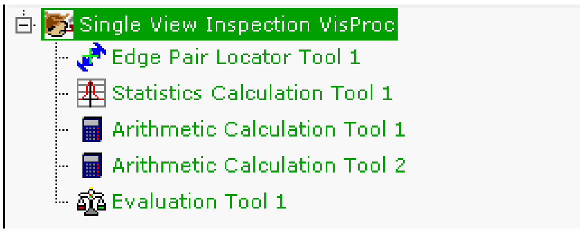


Fig. 5.4.2 (b)

## Study for application

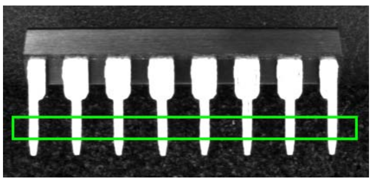
Consider which command tools are applicable for this application. In this example, an edge pair locator tool is used to detect the location of the center of each pin and a statistics calculation tool is used to calculate the interval between pin centers. If a pin is missing or bent, the maximum and minimum intervals would exceed a certain range. To count the number of pins, also set the number of detected intervals as an evaluation criterion. To carry out this inspection, a vision process having the following structure is created:




Use “Edge Pair Locator Tool 1” to detect the location of the center of each pin. Use “Statistics Calculation Tool 1” to calculate the interval using the detection result of “Edge Pair Locator Tool 1”, the maximum, minimum, and mean intervals, and the number of detected pins. Using the measurement values calculated by “Statistics Calculation Tool 1”, calculate the difference between the maximum and mean intervals using “Arithmetic Calculation Tool 1” and the difference between the minimum and mean intervals using “Arithmetic Calculation Tool 2”. Then, set conditions for “Evaluation Tool 1” to output [True] when the calculation results of “Arithmetic Calculation 1” and “Arithmetic Calculation 2” are 2 or less and the number of intervals detected by “Statistics Calculation 1” is 7. This is because seven intervals are detected when the number of detected pins is 8.

## Train Edge Pair Locator Tool 1

Set the search area of “Edge-Pair Tool 1” so that the area encloses all pins and is perpendicular to the pins. Set the search area as shown below because measuring the straight section of each pin increases accuracy. Teach the right and left edges of a pin as the model of “Edge Pair Locator Tool 1”.

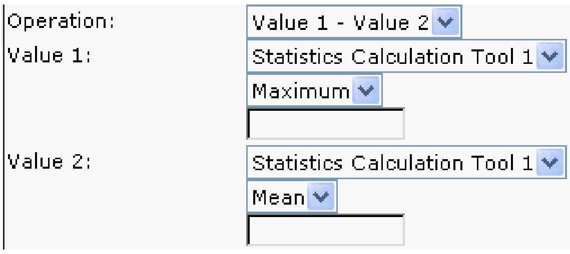


## Train Statistics Calculation Tool 1

For “Statistics Calculation Tool 1”, select [Edge Pair Locator Tool 1] for [Locator Tool], and [Interval] for [Measurement].

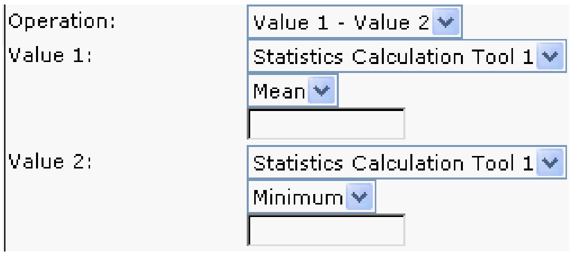
## Train Arithmetic Calculation Tool 1

Use “Arithmetic Calculation Tool 1” to calculate the difference between the maximum and mean intervals calculated by “Statistics Calculation Tool 1”.   On the setup screen, set the items as follows:



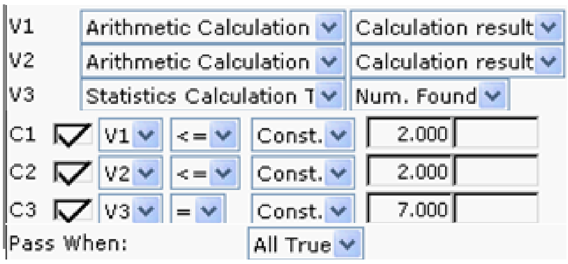

## Train Arithmetic Calculation Tool 2

Use “Arithmetic Calculation Tool 2” to calculate the difference between the minimum and mean intervals calculated by “Statistics Calculation Tool 1”.   On the setup screen, set the items as follows:




## Train Evaluation Tool 1

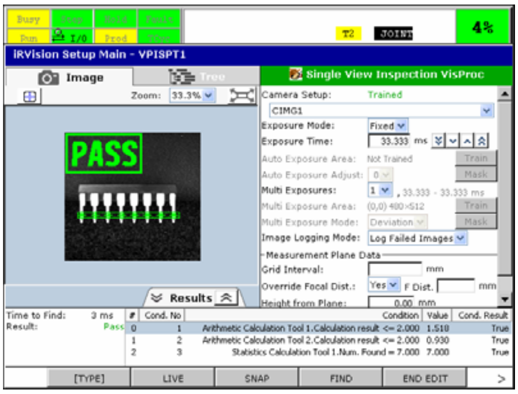
Use “Evaluation Tool 1” to evaluate whether the ID chip meet the criteria. On the setup screen set the items as follows.




## Running a test

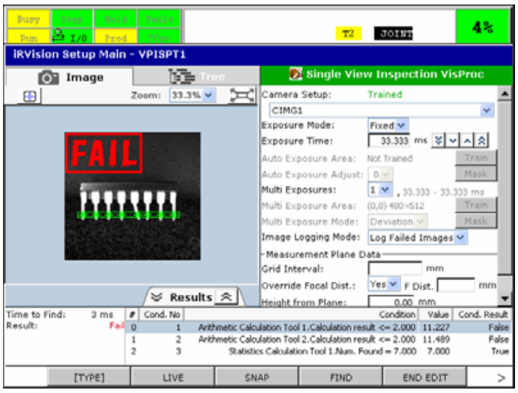
Run the vision process and check that the result is correct.

For the image in Fig. 5.4.2 (a), all conditions are [True] because all measured pin intervals are within the threshold and the number of detected intervals is correct.   Since [All True] is selected for [Pass When], the target passed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Arithmetic Calculation Tool 1.Calculation result <= 2.000 | 1.518 | True |
| 1 | 2 | Arithmetic Calculation Tool 2.Calculation result <= 2.000 | 0.930 | True |
| 2 | 3 | Statistics Calculation Tool 1.Num. Found = 7.000 | 7.000 | True |

For the image in Fig. 5.4.2 (b), the condition 1 and the condition 2 are [False] though the number of detected intervals is correct. Since [All True] is selected for [Pass When], the target failed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Arithmetic Calculation Tool 1.Calculation result <= 2.000 | 11.227 | False |
| 1 | 2 | Arithmetic Calculation Tool 2.Calculation result <= 2.000 | 11.489 | False |
| 2 | 3 | Statistics Calculation Tool 1.Num. Found = 7.000 | 7.000 | True |

# 5.5 ANGLE INSPECTION

This section gives an example of inspecting an angle of an object. The following methods are available as main techniques to determine an angle:

- Detect a taught model using a GPM locator tool.
- Calculate an angle using a geometric calculation tool based on the results of multiple locator tools.

## 5.5.1 Inspecting Whether Targets Have the Same Orientation

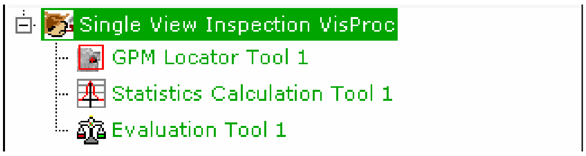
This application inspects whether targets have the same orientation. The inspection should output PASS when the orientation of each detected target is within ±5 degrees of the reference target. Fig. 5.5.1 (a) shows an image of targets within the specified angle, which should pass the inspection. Fig. 5.5.1 (b) shows an image including the center target rotating by more than 5 degrees, which should fail the inspection. A defect inspection is not considered in this example. This application should output FAIL if at least one target is over the orientation threshold.

**Fig. 5.5.1 (a)**

**Fig. 5.5.1 (b)**

## Study for application

Consider which command tools are applicable for this application. Since the orientation of identical objects are of interest in this example, a GPM locator tool will be used to detect each target and the PASS/FAIL evaluation will be made using the detected angle. To evaluate the targets based on the maximum and minimum angles of multiple targets, a statistics calculation tool can be used. To carry out this inspection, a vision process having the following structure is created:




The inspection method would be to use “GPM Locator Tool 1” to detect targets and use “Statistics Calculation 1” to calculate the statistics value of [Angle] of “GPM Locator Tool 1”.

## Train GPM Locator Tool 1

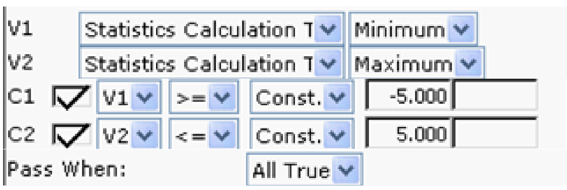
Teach the entire target as shown below as the model for “GPM Locator Tool 1” since the tool needs to detect the orientation correctly. Set –180 to 180 for [Degree of Freedom] for [Angle].


## Train Statistics Calculation Tool 1

For “Statistics Calculation Tool 1”, select [GPM Locator Tool 1] for [Locator Tool] and [Angle] for [Measurement].

## Train Evaluation Tool 1

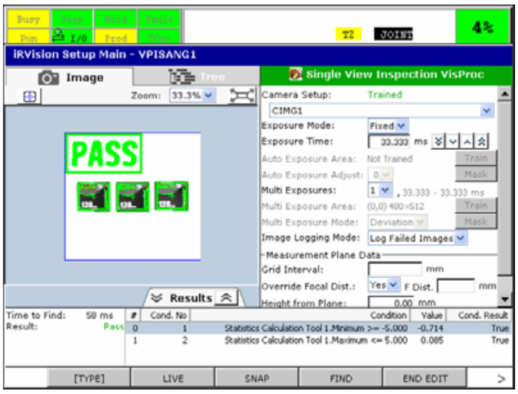
Conditions for “Evaluation Tool 1” are set to evaluate the [Minimum] and [Maximum] values of “Statistics Calculation 1” and to output [True] when the found angles of the targets are all within ±5 degrees. Select [All True] for [Pass When] since only one condition is set.




## Running a test

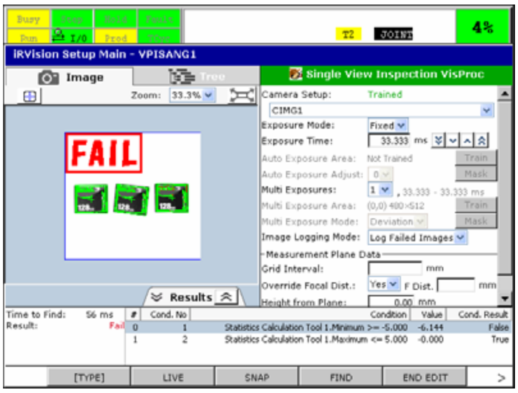
Run the vision process and check that the result is correct.

For the following image, both conditions are [True]. Since [All True] is selected for [Pass When], the target passed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Statistics Calculation Tool 1.Minimum >= -5.000 | -0.714 | True |
| 1 | 2 | Statistics Calculation Tool 1.Maximum <= 5.000 | 0.085 | True |

For the following image, the [Minimum] value of “Statistics Calculation 1” is below the threshold, and the condition 1 is [False]. Since [All True] is selected for [Pass When], the target failed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Statistics Calculation Tool 1.Minimum >= -5.000 | -6.144 | False |
| 1 | 2 | Statistics Calculation Tool 1.Maximum <= 5.000 | -0.000 | True |

# 5.6 AREA INSPECTION

This section gives an example of inspecting an area in an image. The following methods are available as main techniques to inspect an area:

- Inspect the areas of individual features detected by a blob locator tool.
- Calculate the sum of the areas measured by a blob locator tool using a statistics calculation tool.

## 5.6.1 Inspecting the Amount of Coated Chemicals

In this example, the amount of silicone applied to a designated position on a fixed printed circuit board is inspected. The target printed circuit board would pass this inspection if the area of the applied silicone covers at least 7000 pixels in the image. Fig. 5.6.1 (a) shows an image of a printed circuit board with a sufficient amount of silicone, which should pass the inspection. Fig. 5.6.1 (b) shows an image of a printed circuit board with insufficient amount of silicone, which should fail the inspection.

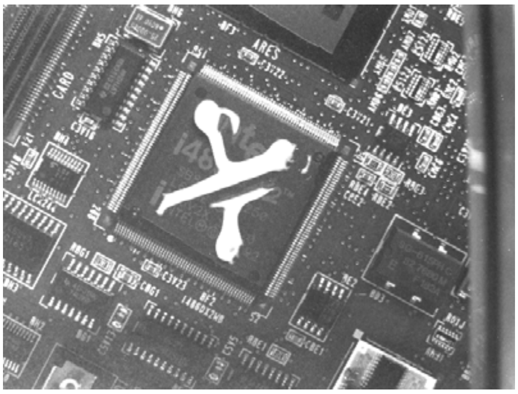


Fig. 5.6.1 (a)

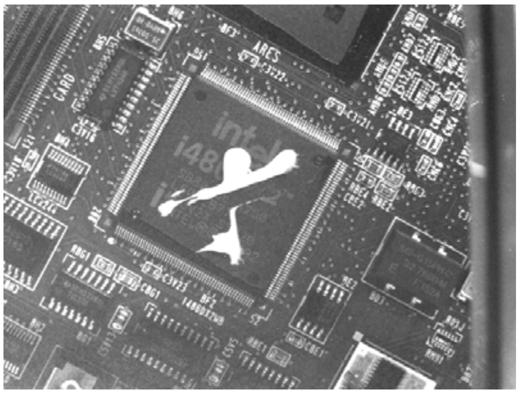
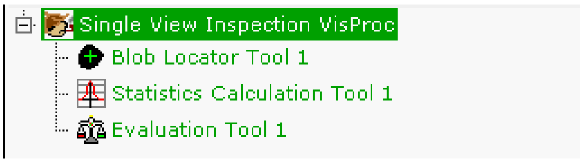


Fig. 5.6.1 (b)

## Study for application

Consider which command tools are applicable for this application. For this example, a blob locator tool will be used to measure the area of applied silicone. Due to the chemical application procedure, the applied silicone is divided into multiple blobs. Therefore, it is necessary to calculate the sum of the areas of the blobs using a statistics calculation tool for the finale PASS/FAIL evaluation. To carry out this inspection, a vision process having the following structure is created:




The inspection method would be to detect the region where silicone is applied using “Blob Locator Tool 1”, then calculate the sum of the areas using “Statistics Calculation Tool 1”, and finally evaluate the calculated sum of the areas using “Evaluation Tool 1” for the final PASS/FAIL evaluation.

## Train Blob Locator Tool 1

For an accurate inspection, it is necessary to generate an image for teaching a “Blob Locator Tool 1” model so that the area of silk printing or solder on the printed circuit board is distinguished from the area of silicone. However, because the shape of the blobs that can be found varies, a constraint specific to geometric features such as a perimeter or longitudinal length cannot be set as the characteristic parameter for the blob locator tool. From the image, an observation is made that the applied silicone appears bright in particular. Therefore, the [Threshold value] is adjusted so that only pixels having a specified brightness are extracted when the image is binarized.

Fig. 5.6.1 (c)

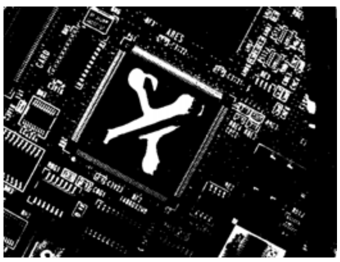


Fig. 5.6.1 (d)

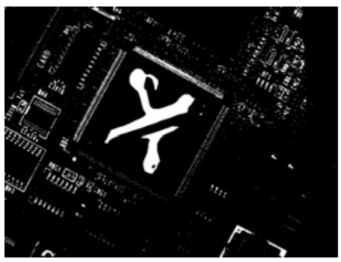
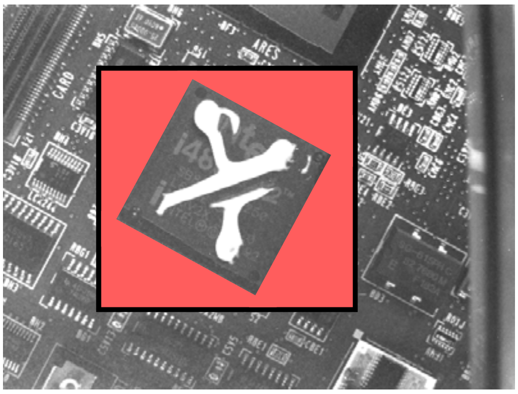
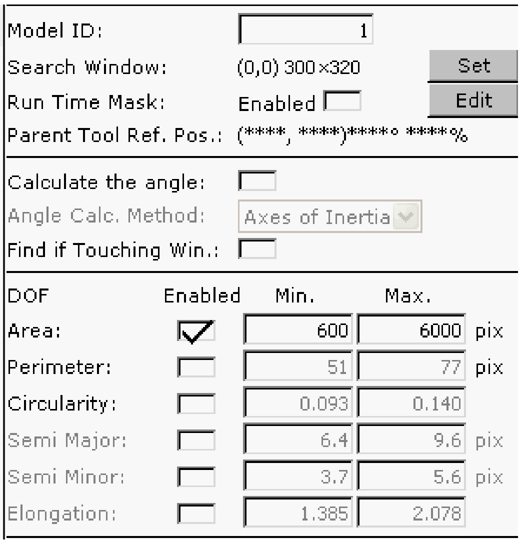


Fig. 5.6.1 (e)

Fig. 5.6.1 (c) shows the binarized image of Fig. 5.6.1 (a) when [Threshold value] is set to 100 (default value), Fig. 5.6.1 (d) shows the binarized image when it is set to 180, and Fig. 5.6.1 (e) shows the binarized image when it is set to 255. From these images, the desirable contrast threshold value seems to be a value between 180 and 255. In this example, the value is set to 230. Even after this adjustment, solder on the IC leads on the printed circuit board still appears bright. To prevent incorrect detection, and to exclude blobs with less than a certain area as noise, the search area and detection parameters are set as shown in the diagrams below, to narrow the search window to the minimum required for the inspection. Run a test for multiple images using the blob locator tool to determine the minimum and maximum values for [Area]. In this example, a feature whose area is less than 600 pixels is considered as a noise and a feature whose area is more than 6000 pixels is considered as a foreign object.



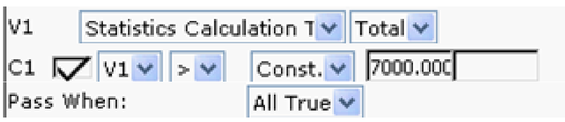



## Train Statistics Calculation Tool 1

Train "Statistics Calculation Tool 1" to calculate [Area] of "Blob Locator Tool 1".

## Train Evaluation Tool 1

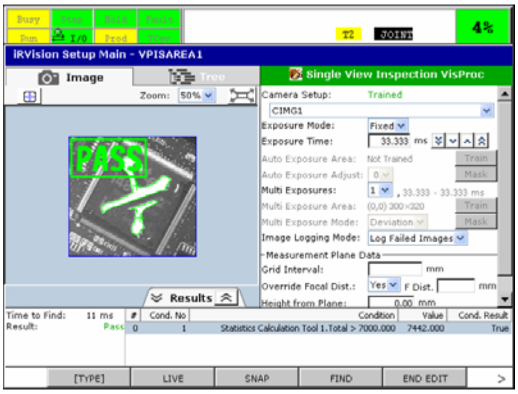
A condition for "Evaluation Tool 1" is set to evaluate the [Total] value of "Statistics Calculation Tool 1" and to output [True] when the total area is at least 7000 pixels. Select [All True] for [Pass When] since only one condition is set.




## Running a test

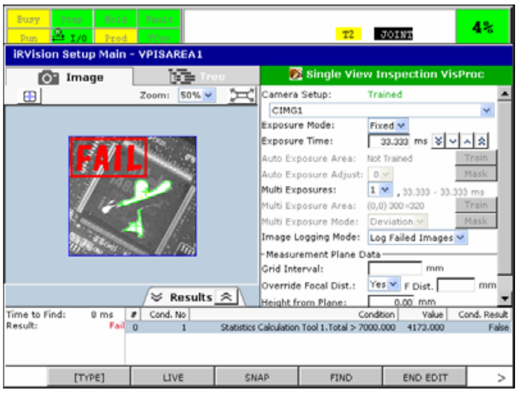
Run the vision process and check that the result is correct.

For the following image, the measurement value satisfied the specified condition for the printed circuit board with sufficient amount of silicone applied. Since [All True] is selected for [Pass When], the target passed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Statistics Calculation Tool 1.Total > 7000.000 | 7442.000 | True |

For the following image, the measurement value did not satisfy the specified condition for the printed circuit board with insufficient amount of silicone applied. Since [All True] is selected for [Pass When], the target failed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Statistics Calculation Tool 1.Total > 7000.000 | 4173.000 | False |

# 5.7 SURFACE FLAW INSPECTION

This section gives an example of inspecting defects like scratches and dents on the surface of target objects. In this application, the lighting methods described in "3.5 Selection of the light" are very important to emphasize the defects. The following methods are available as main techniques to inspect defects like scratches and dents:

- Use Surface flaw inspection tool to inspect defects like scratches and dents.
- Use a coaxial vertical light to illuminate the inspection surface. This lighting method is very effective to emphasize defects like scratches and dents. Generally, a fluorescent light is not suitable for these inspections.
- Use a low-angle light to illuminate the inspection surface. Printed material and the desired scratches on the surface of the target are difficult to distinguish. However, this lighting method can erases the prints and emphasizes scratches in the captured image.

## 5.7.1 Inspecting Scratches and Dents

In this example, scratches and dents on the surface of a button-type battery are inspected. Fig. 5.7.1 (a) shows an image without scratches and dents, which should pass the inspection. Fig. 5.7.1 (b) shows an image with three scratches, which should fail the inspection. Fig. 5.7.1 (c) shows an image with two dents, which should fail the inspection. All of the images are acquired using a fluorescent light.

Fig. 5.7.1 (a)

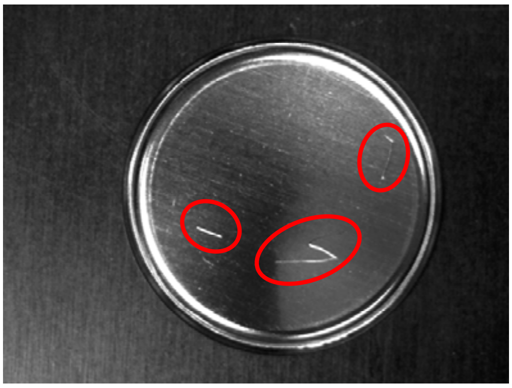


Fig. 5.7.1 (b)

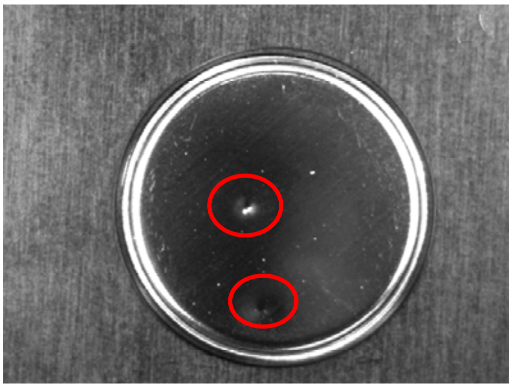


Fig. 5.7.1 (c)

Fig. 5.7.1 (b) shows three scratches which are circled in red. There is only a small difference between the brightness of the upper right scratch and the surface of the battery. The upper right scratch needs to be emphasized for a stable inspection. Fig. 5.7.1 (c) shows two dents which are circled in red. There is only a small difference between the brightness of the lower dent and the surface of the battery. The lower dent needs to be emphasized for a stable inspection.

## Study for application

Under the condition using a fluorescent light, some of the scratches and dents are not clear in the acquired images. Consider which lighting methods are applicable for this application to emphasize all of the scratches and dents. The surface of the button-type battery has a flat plane but the areas of the scratches and dents have an uneven plane. When a coaxial vertical lighting method is used for this kind of target, the surface of the target appears bright (in white) and the scratches and dents appear dark (in black). The following three figures show the acquired images using a coaxial vertical light instead of a fluorescent light. Fig. 5.7.1 (d) shows an image without scratches and dents. Fig. 5.7.1 (e) shows an image with three scratches and the upper right scratch appears in the acquired image. Fig. 5.7.1 (f) shows an image with two dents and the lower dent appears in the acquired image:

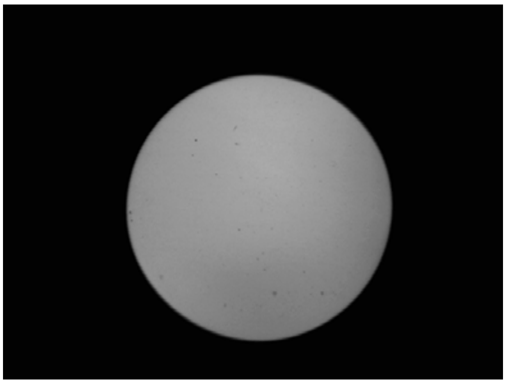


Fig. 5.7.1 (d)

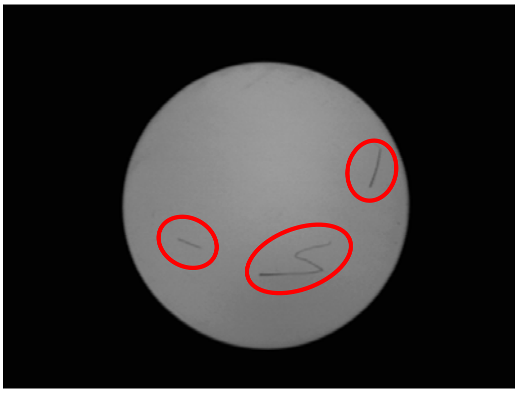


Fig. 5.7.1(e)

Fig. 5.7.1 (f)

After the appropriate images for the inspection are prepared, proceed to teaching the vision process. To carry out this inspection, create two vision processes, one for scratches and the other for dents. Both vision processes should have the following structure.

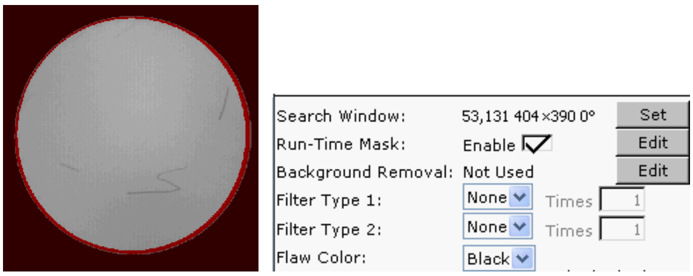

Use "Surface Flaw Inspection Tool 1" to inspect the defects on the surface of the button-type battery. Since the target is not fixed in position, use "Window Shift Tool 1" to detect the location of each target. Use "Evaluation Tool 1" to evaluate the defects detected by "Surface Flaw Inspection Tool 1" for the PASS/FAIL evaluation.

## Train GPM Locator Tool 1

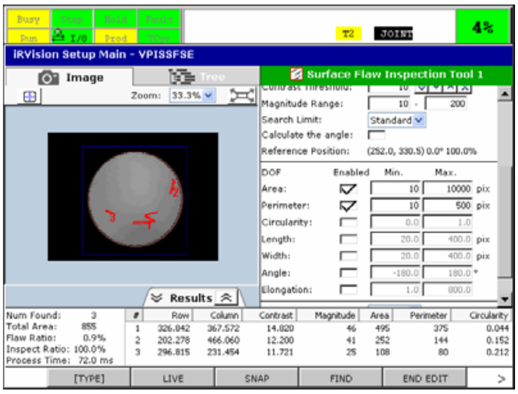
Teach the perimeter of the button-type battery as the model.

## Train Surface Flaw Inspection Tool 1

For an accurate inspection, set the inspection area using the search window and run-time mask. The scratches and dents are in black in the image, so select "black" as "Flaw Color".

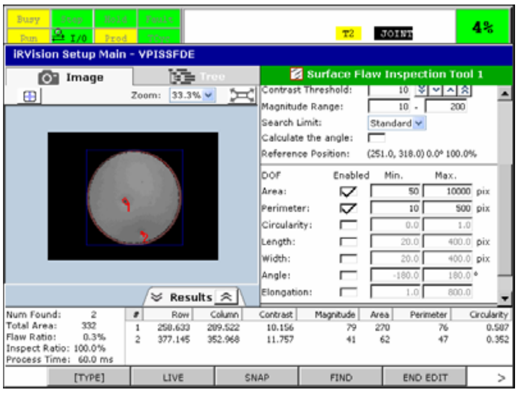



Run a test for multiple button-type batteries using the surface flaw inspection tool to detect the desired scratches. Adjust the parameters of "Area" or "Perimeter" and so on not to detect unnecessary scratches.




| # | Row | Column | Contrast | Magnitude | Area | Perimeter | Circularity |
|---|---|---|---|---|---|---|---|
| 1 | 326.842 | 367.572 | 14.820 | 46 | 495 | 375 | 0.044 |
| 2 | 202.278 | 466.060 | 12.200 | 41 | 252 | 144 | 0.152 |
| 3 | 296.815 | 231.454 | 11.721 | 25 | 108 | 80 | 0.212 |

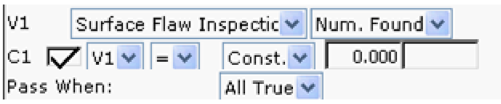
Run a test for multiple button-type batteries using the surface flaw inspection tool to detect the desired dents. Adjust the parameters of "Area" or "Perimeter" and so on not to detect unnecessary dents.




| # | Row | Column | Contrast | Magnitude | Area | Perimeter | Circularity |
|---|---|---|---|---|---|---|---|
| 1 | 258.633 | 289.522 | 10.156 | 79 | 270 | 76 | 0.587 |
| 2 | 377.145 | 352.968 | 11.757 | 41 | 62 | 47 | 0.352 |

## Train Evaluation Tool 1

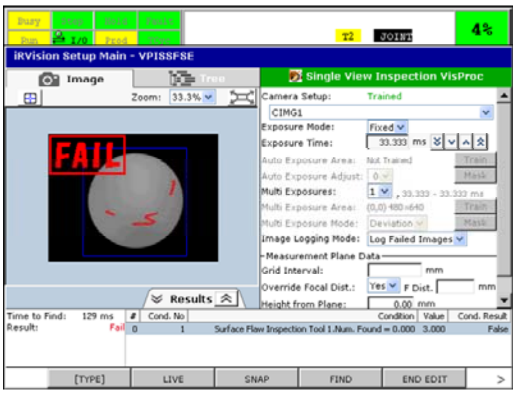
A condition for "Evaluation Tool 1" is set to evaluate the [Num. Found] value of "Surface Flaw Inspection Tool 1" and to output [True] when there are no defects on the surface of the target. Select [All True] for [Pass When] since only one condition is set.




## Running a test

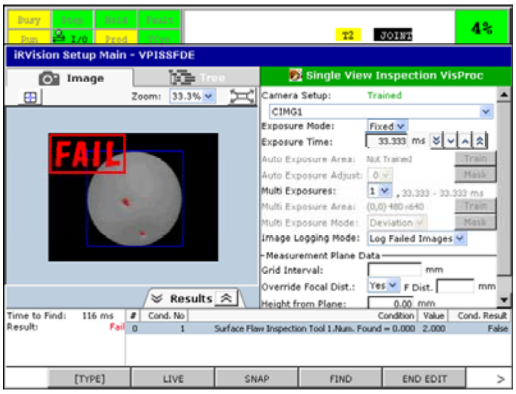
Run the vision process and check that the result is correct.

For the following image, the detected value did not satisfy the specified condition for scratches on the surface of the target. Since [All True] is selected for [Pass When], the target failed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Surface Flaw Inspection Tool 1.Num. Found = 0.000 | 3.000 | False |

For the following image, the detected value did not satisfy the specified condition for dents on the surface of the target. Since [All True] is selected for [Pass When], the target failed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Surface Flaw Inspection Tool 1.Num. Found = 0.000 | 2.000 | False |

## 5.7.2 Inspecting Scratches on Target with Printing

In this example, there are scratches and printed numbers on the surface of a button-type battery. Fig. 5.7.2 (a) shows an image with scratches and prints which is acquired using a coaxial vertical light, so both scratches and a prints appear in black. The printed numerical numbers vary according to the type of a button-type battery. When the printed numerical numbers change, a surface flaw inspection tool may detect the printed numbers as defects. If the lighting method makes the prints invisible in an image, there is a less possibility of wrong detection.

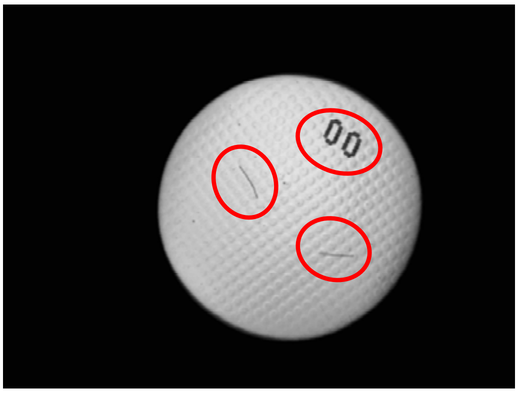


**Fig. 5.7.2 (a)**

### Study for application

Under the condition using a coaxial vertical light, both scratches and printed numbers appear in black. Consider which lighting methods are applicable for this application to erase the printed numbers and emphasize only the scratches. The printed area is flat but the area of the scratches has an uneven plane. When a low-angle lighting method is used for this kind of target, the surface of the target appears dark (in black) and the printed area is also in black. Only the area of the scratches scatters light, so it appears bright (in white). The following figures show the acquired images using a low-angle light instead of a coaxial vertical light. Fig. 5.7.2 (b) shows the scratch in white and erases the printed numbers from the image:

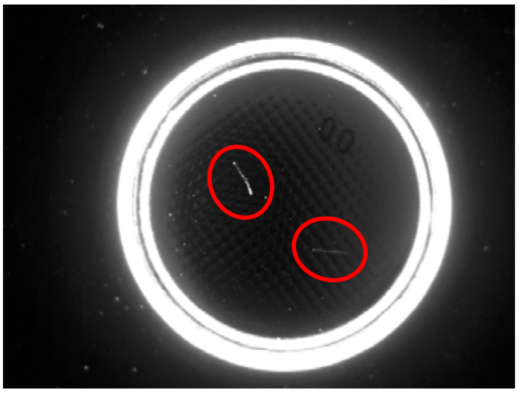


**Fig. 5.7.2 (b)**

After the appropriate images for the inspection are prepared, proceed to teaching the vision process. To carry out this inspection, a vision process having the following structure is created:

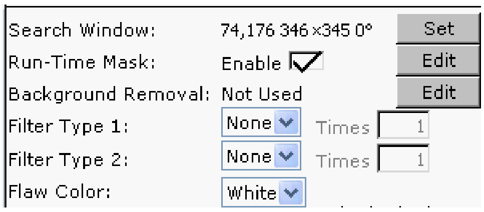

Use “Surface Flaw Inspection Tool 1” to inspect the defects on the surface of the button-type battery. Since the target is not fixed in position, use “Window Shift Tool 1” to detect the location of each target. Use “Evaluation Tool 1” to evaluate the defects detected by “Surface Flaw Inspection Tool 1” for the PASS/FAIL evaluation.

## Train GPM Locator Tool 1

Teach the perimeter of the button-type battery as the model.

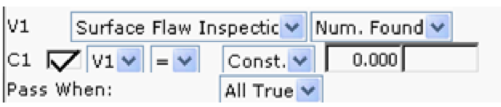
## Train Surface Flaw Inspection Tool 1

For an accurate inspection, set the inspection area using the search window and run-time mask. The scratches are in white in the image, so select “white” as “Flaw Color”.




## Train Evaluation Tool 1

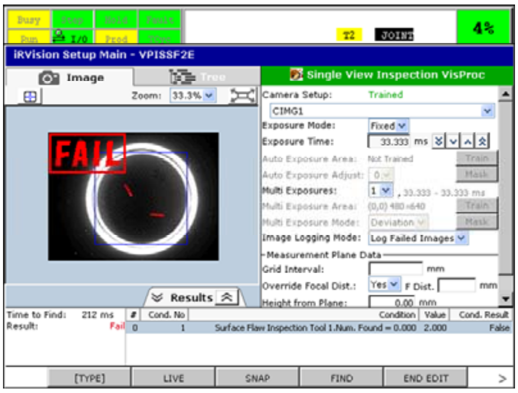
A condition for “Evaluation Tool 1” is set to evaluate the [Num. Found] value of “Surface Flaw Inspection Tool 1” and to output [True] when there are no defects on the surface of the target. Select [All True] for [Pass When] since only one condition is set.

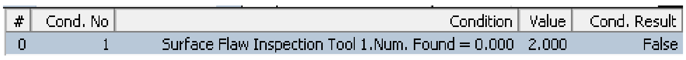


### Running a test

Run the vision process and check that the result is correct.

For the following image, the detected value did not satisfy the specified condition for scratches on the surface of the target. Since [All True] is selected for [Pass When], the target failed the inspection.




## 5.8 BEAD INSPECTIONS

This section introduces sample applications for inspecting a bead-formed object.

Bead Inspection tool provides a function to inspect a continuous linear object, such as sealant or adhesive. According to an inspection line you want to inspect, the Bead Inspection Tool generates many inspection points equally spaced along the inspection line. Then, you train the inspection contents as a child tool on a representative inspection point. Bead Inspection tool processes the same inspection contents over all of the generated inspection points. As a result, Bead Inspection Tool can be customized to inspect the bead brightness, bead width and/or bead position depending on the type of child tools. The following methods are available as main techniques to inspect the bead-formed object:

- Measure the brightness on a bead-formed object using a histogram tool.
- Measure the width of a bead-formed object using an edge pair locator tool.
- Measure the position of a bead-formed object using an edge pair locator tool.
- Measure the position of a bead-formed object using multiple histogram tools.

### 5.8.1 Inspecting the sealant for gaps

This application inspects the shortage of a sealant (gaps). Fig. 5.8.1 (a) shows an image without gaps along a sealant, which should pass the inspection. Fig. 5.8.1 (b) shows an image with a gap along a sealant, which should fail the inspection. Assume that the sealant is always applied at the same position.

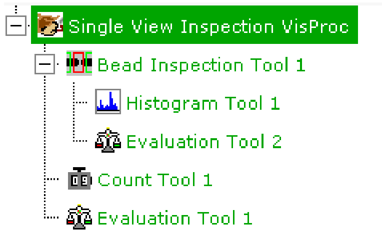
**Fig. 5.8.1 (a)**

**Fig. 5.8.1 (b)**

> **Hint**
>
> When the bead-formed object is roughly positioned, offset the inspection points of Bead Inspection tool using Window Shift tool. Please see the 5.8.4, "Inspecting the position of O–ring using two Histogram tools".

## Study for application

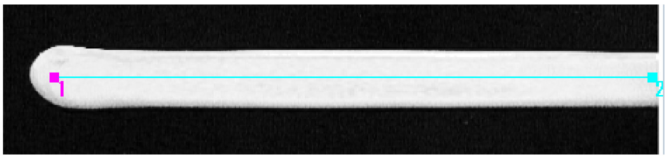
Consider which command tools are applicable for this application. In this example, an area with applied sealant appears white and relatively black when missing sealant. Histogram tool will be used to measure the brightness and an evaluation tool to evaluate the final PASS/FAIL. To carry out this inspection, a vision process having the following structure is created:



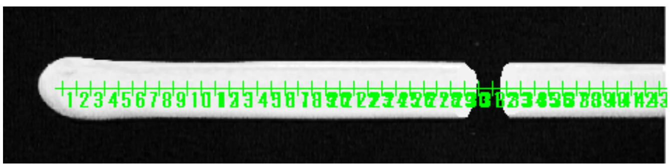

Use “Bead Inspection Tool 1” to generate inspection points along the sealant. Use “Histogram Tool 1” to measure the brightness at each inspection point and “Evaluation Tool 2” to evaluate the result of “Histogram Tool 1” at each inspection point. Use “Count Tool 1” to count the number of FAIL and the last “Evaluation Tool 1” to evaluate the PASS/FAIL for this inspection.

## Train Bead Inspection Tool 1

Train “Bead Inspection Tool 1” so that the inspection line runs through the center line of the sealant you want to inspect, as shown in the figure below.

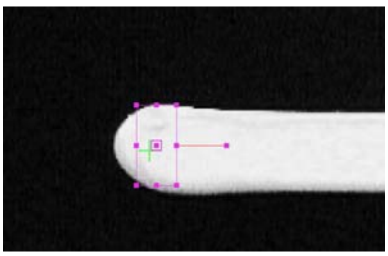


Set [inspection interval] you want to inspect in “Bead Inspection Tool 1”. First, you prepare a work piece with shortage along the sealant, and snap and find in “Bead Inspection Tool 1”. As a result, you can confirm all of the inspection points plotted on the image. The following image shows the inspection pointes plotted on the image. Each cross-hair is an inspection point. There are two inspection points in the gap region. In this case, it is a reasonable interval size for testing for gaps. For a more extensive evaluation, select a smaller interval value. In this example, the inspection interval field is set to 10.



## Train Histogram Tool 1

Set the measurement area of “Histogram Tool 1” at first inspection point. The cross-hair shows the first inspection point, and the rectangular shows the measurement area of “Histogram Tool 1”.



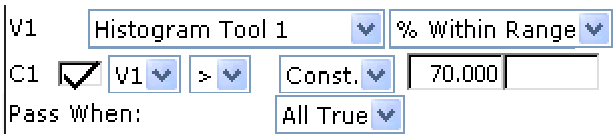
Let’s think about the relationship between a inspection interval and a width of the measurement area.

There is a complementary relationship between the inspection interval specified in “Bead Inspection Tool 1” and the width of the measurement area in “Histogram Tool 1”. For example, if you want to extend the inspection interval in “Bead Inspection Tool 1”, then you need to enlarge the width of the measurement area in “Histogram Tool 1” to clear the inspection gap. On the other hand, if you want to narrow the inspection interval, then you need to reduce the width of the measurement area to avoid overlapping inspection. It is possible for the detailed inspection to narrow the inspection interval, but the processing time to inspect may take longer.

## Train Evaluation Tool 2

In “Evaluation Tool 2”, the PASS/FAIL evaluation condition on each inspection point will be specified.

In this example, the histogram tool measurement results of the sealant with and without shortage are compared to determine the criterion. Run a test for multiple sealant samples using the histogram tool, to measure the percentage of pixel whose brightness values are between 128 and 255, for both good and bad. In these sealant samples, it was found that the percentage of pixels within range was 80% or more for the portion without shortage along the sealant, and was 60% or less for the portion with shortage along the sealant. From this data, around 70%, which is the middle value of 60% and 80%, seemed to be an appropriate criterion. Set the condition for “Evaluation Tool 2” to output [True] when the percentage of pixels is within the range 70 or more for all of the inspection pointes. Select [All True] for final evaluation condition of the evaluation tool.

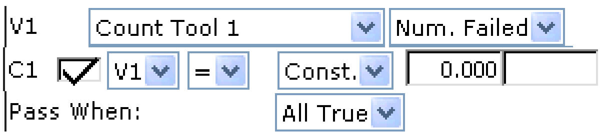



## Train Count Tool 1

“Count Tool 1” needs to count the PASS /FAIL number of all the inspection points. Leave “Count Tool 1” at default value setting since a count tool counts the total number of item evaluated by all sibling inspection tools that appears before it on the tree view.

## Train Evaluation Tool 1

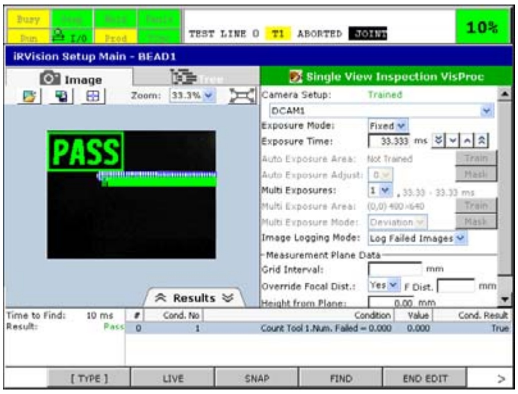
Set the condition for “Evaluation Tool 1” to output [True] when [Num. Failed] of “Count Tool 1” is 0. Select [All True] for [Pass When] since only one condition is set.




## Running a test

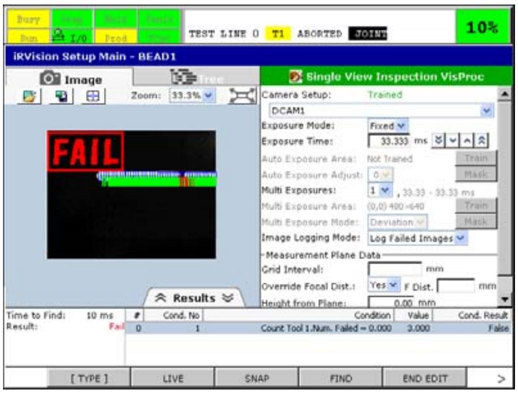
Run the vision process and check that the result is correct.

For the following image in Fig. 5.8.1 (a), the condition is [True] because none of the result in “Histogram Tool 1” is evaluated as [Fail] with “Count Tool 1”. Since [All True] is selected for [Pass When], the sealant passed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Count Tool 1.Num. Failed = 0.000 | 0.000 | True |

For the following image in Fig. 5.8.1 (b), the condition is [False] because 3 results in “Histogram Tool 1” are evaluated as [Fail] with “Count Tool 1”. Since [All True] is selected for [Pass When], the sealant failed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Count Tool 1.Num. Failed = 0.000 | 3.000 | False |

## 5.8.2 Inspecting the Width of a sealant

This application inspects the width of a sealant. In Fig. 5.8.2(a), the width to be inspected is indicated by arrows in the image. If a width is within the specified range, then should pass the inspection. If a width is outside the specified range as shown in Fig. 5.8.2(b), then should fail the inspection. Assume that the sealant is always applied at the same position.

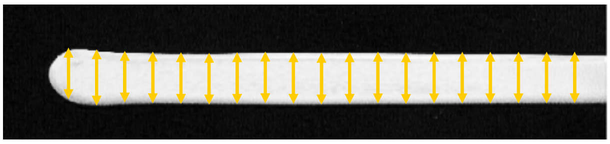


**Fig. 5.8.2(a)**

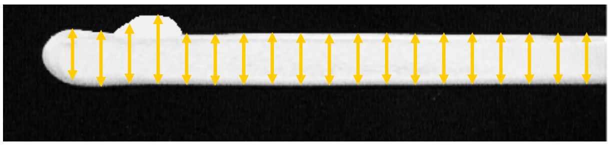
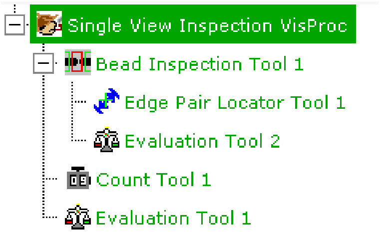


**Fig. 5.8.2(b)**

> **Hint**
> When the bead-formed object is roughly positioned, offset the inspection points of Bead Inspection tool using Window Shift tool. Please see the 5.8.4, “Inspecting the position of O–ring using two Histogram tools”.

### Study for application

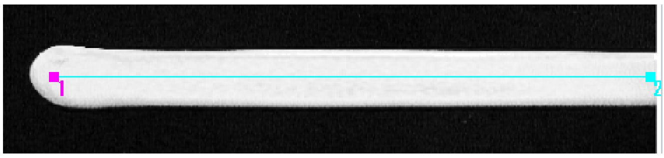
Consider which command tools are applicable for this application. In this example, the width to be inspected is the length between two straight lines. Therefore, the width can be measured using an edge pair locator tool. To carry out this inspection, a vision process having the following structure is created:



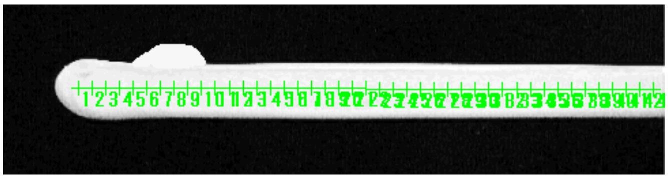

Use “Bead Inspection Tool 1” to generate inspection points along the sealant. Use “Edge Pair Locator Tool 1” to measure the width of the sealant on each inspection point and “Evaluation Tool 2” to evaluate the result of “Edge Pair Locator Tool 1” at each inspection point. Use “Count Tool 1” to count the number of points that FAIL and the last “Evaluation Tool 1” to evaluate the PASS/FAIL for this inspection.

## Train Bead Inspection Tool 1

Train “Bead Inspection Tool 1” so that the inspection line runs through the center line of the sealant you want to inspect, as shown in the figure below.

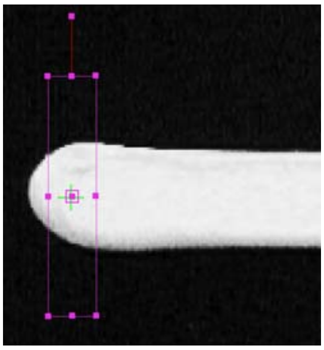


Set [inspection interval] you want to inspect in “Bead Inspection Tool 1”. First, you prepare a work piece with wide sealant, and snap and find in “Bead Inspection Tool 1”. As a result, you can confirm all of the inspection points plotted on the image. The following image shows the inspection points plotted on the image. Each cross-hair is an inspection point. You can see a few inspection points plotted around the wide sealant portion. In this case, it may be a reasonable interval size for inspecting for wide sealant. If you want more extensive evaluation, you can make the interval size smaller. In this example, the inspection interval field is set to 10.



## Train Edge Pair Locator Tool 1

Set [Search Area] to measure the sealant width at first inspection point. The cross-hair shows the first inspection point and the rectangular shows the [Search Area] of “Edge Pair Locator Tool 1”, as shown in the figure below. In this example, the edge pair locator tool measurement results of the sealant with and without wide portion are compared to determine the criterion. Run a test for multiple sealant samples using the edge pair locator tool, to measure the width of sealant for both good and bad. In these sealant samples, it was found that the sealant width of good portion was between 35 and 45 pixels. The edge pair locator tool needs to measure wide sealant for bad portion, so [Min. Length to Search] and [Max. Length to Search] in “Edge Pair Locator Tool 1” are set to 25 and 55, respectively. These values should be decided according to the work piece you want to inspect.



## Train Evaluation Tool 2

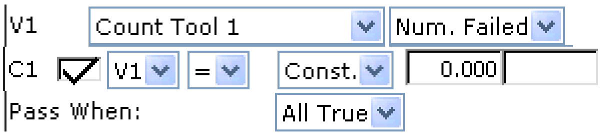
In “Evaluation Tool 2”, the PASS/FAIL evaluation condition on each inspection point will be specified. In this example, [Length] measured in “Edge Pair Locator Tool 1” should be evaluated and set the condition for “Evaluation Tool 2” to output [True] when the length is between 35 to 45 pixels. Select [All True] for final evaluation condition of the evaluation tool.


## Train Count Tool 1

“Count Tool 1” needs to count the PASS /FAIL number of all the inspection points. Leave “Count Tool 1” at default value setting since a count tool counts the total number of item evaluated by all sibling inspection tools that appears before it on the tree view.

## Train Evaluation Tool 1

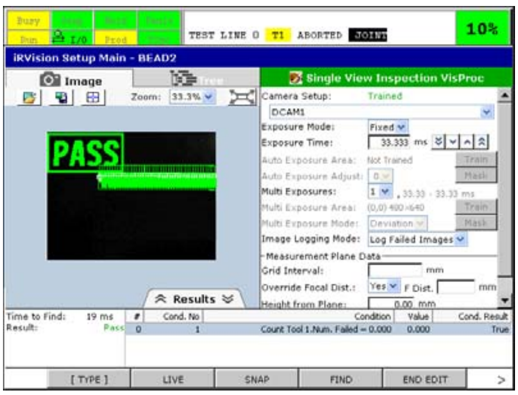
Set the condition for “Evaluation Tool 1” to output [True] when [Num. Failed] of “Count Tool 1” is 0. Select [All True] for [Pass When] since only one condition is set.

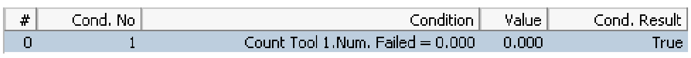



## Running a test

Run the vision process and check that the result is correct.

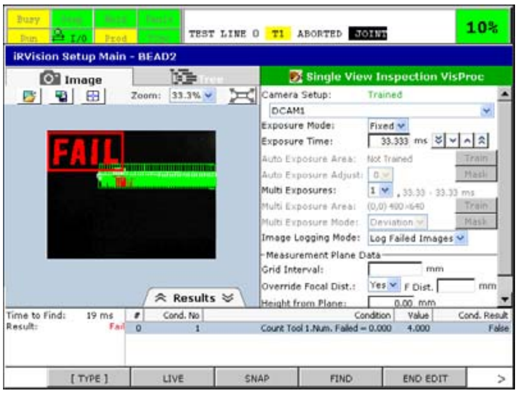
For the following image in Fig. 5.8.2 (a), the condition is [True] because none of the result in “Edge Pair Locator Tool 1” is evaluated as [Fail] with “Count Tool 1”. Since [All True] is selected for [Pass When], the sealant passed the inspection.



| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Count Tool 1.Num. Failed = 0.000 | 0.000 | True |

For the following image in Fig. 5.8.2 (b), the condition is [False] because 4 results in “Edge Pair Locator Tool 1” are evaluated as [Fail] with “Count Tool 1”. Since [All True] is selected for [Pass When], the sealant failed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Count Tool 1.Num. Failed = 0.000 | 4.000 | False |

## 5.8.3 Inspecting the Position of a sealant

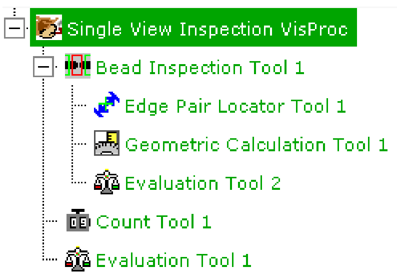
This application inspects the position of a sealant. Fig. 5.8.3(a) shows an image with a straight sealant, which should pass the inspection. Fig. 5.8.3(b) shows an image with a curved sealant, which should fail the inspection. This inspection requires that the sealant is always applied on the same position.


Fig. 5.8.3 (a)


Fig. 5.8.3 (b)

### Study for application

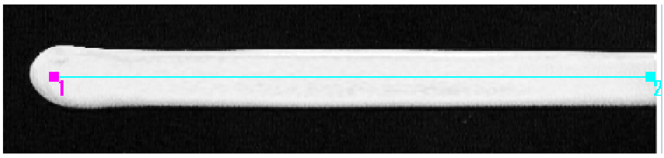
Consider which command tools are applicable for this application. In this example, a sealant has constant width along the sealant line, so edge pair locator tool can be used for this example. Edge Pair Locator Tool measures not only the width but also the position as midpoint of the width. And, Bead Inspection Tool outputs the position of each inspection point. We can inspect a curved sealant to evaluate the distance between the midpoint of the sealant width and the inspection point. The distance can be calculated by geometric calculation tool. Evaluation tool evaluates the calculated distance to decide final PASS/FAIL. To carry out this inspection, a vision process having the following structure is created.



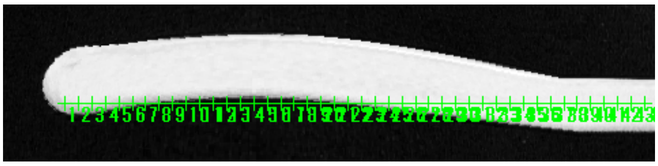

Use “Bead Inspection Tool 1” to generate the inspection points along the sealant. Use “Edge Pair Locator Tool 1” to measure the width and midpoint of the width on each inspection point and “Geometric Calculation Tool 1” to calculate the distance between the position which is generated by “Bead Inspection Tool 1” and the midpoint which is measured by “Edge Pair Locator Tool 1”. Use “Evaluation Tool 2” to evaluate the result of “Geometric Calculation Tool 1” on each inspection point. Use “Count Tool 1” to count the number of FAIL and the last “Evaluation Tool 1” to evaluate the PASS/FAIL for this inspection.

## Train Bead Inspection Tool 1

Train “Bead Inspection Tool 1” so the inspection line runs through the center line of the sealant you want to inspect, as shown in the figure below.

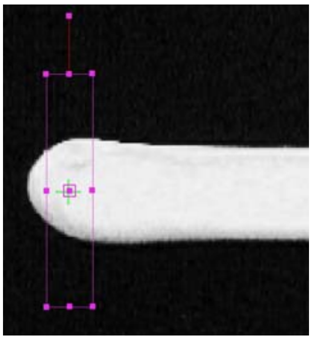



Set [inspection interval] you want to inspect in “Bead Inspection Tool 1”. First, you prepare a work piece with curved sealant, and snap and find in “Bead Inspection Tool 1”. As a result, you can confirm all of the inspection points plotted on the image. The following image shows the inspection pointes plotted on the image. Each cross-hair is inspection point and you can see a few inspection points plotted around curved sealant portion. In this case, it may be a reasonable interval for inspecting curved sealant. If you want more extensive evaluation, you can set more small value as interval. In this example, [inspection interval] is set to 10.



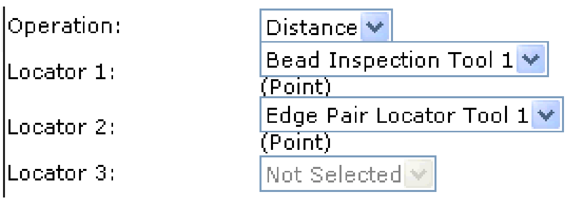
## Train Edge Pair Locator Tool 1

Set [Search Area] to measure the sealant width on first inspection point. The cross-hair shows the first inspection point and the rectangular shows the [Search Area] of “Edge Pair Locator Tool 1”, as shown in the figure below. [Search Area] should be set to measure not only straight portion but also curved portion, so you’d better set the enlarged [Search Area]..



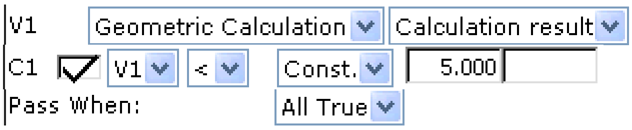
## Train Geometric Calculation Tool 1

“Geometric Calculation Tool 1” to calculate the distance between the position which is generated by “Bead Inspection Tool 1” and the midpoint which is measured by “Edge Pair Locator Tool 1”. Select “Distance” as [Operation], “Bead Inspection Tool 1” as [Locator1] and “Edge Pair Locator Tool 1”as [Locator 2].



## Train Evaluation Tool 2

In “Evaluation Tool 2”, the PASS/FAIL evaluation condition on each inspection point will be specified. In this example, [Calculation result] calculated in “Geometric Calculation Tool 1” should be evaluated and set the condition for “Evaluation Tool 2” to output [True] when the distance(=[Calculation result]) is less than 5 pixels. This evaluation value should be decided using multiple sample work pieces. Select [All True] for final evaluation condition of the evaluation tool.

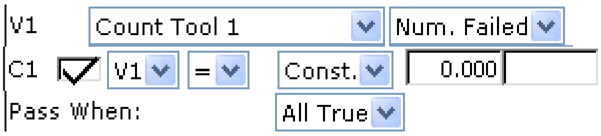



## Train Count Tool 1

“Count Tool 1” needs to count the PASS /FAIL number of all the inspection points. Leave “Count Tool 1” at default value setting since a count tool counts the total number of item evaluated by all sibling inspection tools that appears before it on the tree view.

## Train Evaluation Tool 1

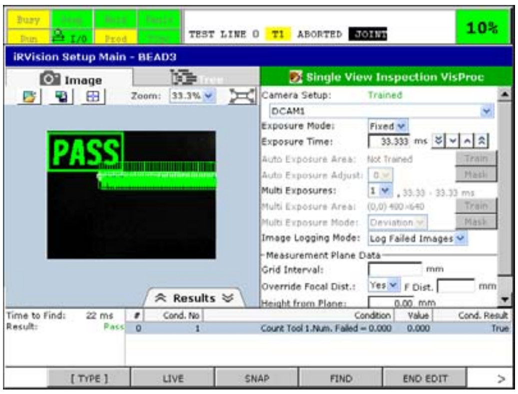
Set the condition for “Evaluation Tool 1” to output [True] when [Num. Failed] of “Count Tool 1” is 0. Select [All True] for [Pass When] since only one condition is set.




## Running a test

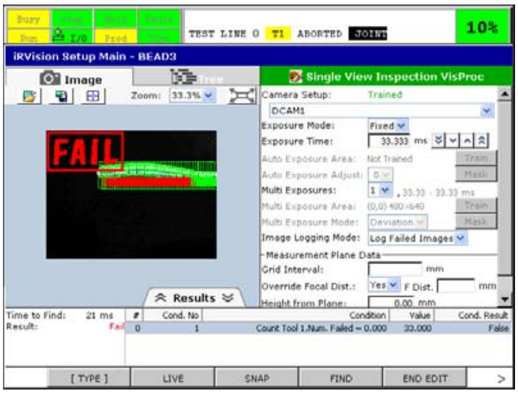
Run the vision process and check that the result is correct.

For the following image in Fig. 5.8.3 (a), the condition is [True] because none of the result in “Geometric Calculation Tool 1” is evaluated as [Fail] with “Count Tool 1”. Since [All True] is selected for [Pass When], the sealant passed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Count Tool 1.Num. Failed = 0.000 | 0.000 | True |

For the following image in Fig. 5.8.3 (b), the condition is [False] because 33 results in “Geometric Calculation Tool 1” are evaluated as [Fail] with “Count Tool 1”. Since [All True] is selected for [Pass When], the sealant failed the inspection.




| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Count Tool 1.Num. Failed = 0.000 | 33.000 | False |

## 5.8.4 Inspecting the Position of O-ring using two Histogram Tools

This application inspects whether an o-ring is set to correct position before assembling. Bead inspection tool is used for inspecting not only a sealant but also an o-ring. Fig. 5.8.4 (a) shows an O-ring is properly set in chase on a metal part, which should pass the inspection. Fig. 5.8.4 (b) shows an O-ring deviates a little from a chase on a metal part (shows bracket in the figure), which should fail the inspection. Assume that the metal part to be set an O-ring is roughly positioned.

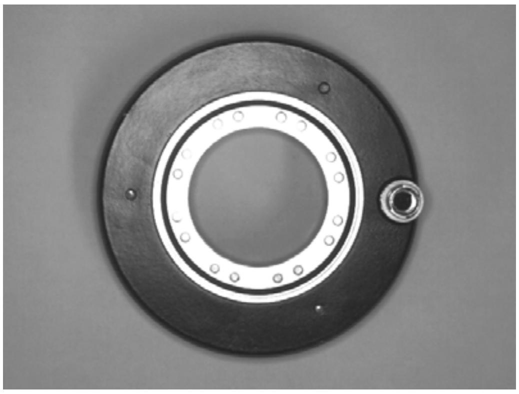


Fig. 5.8.4(a)

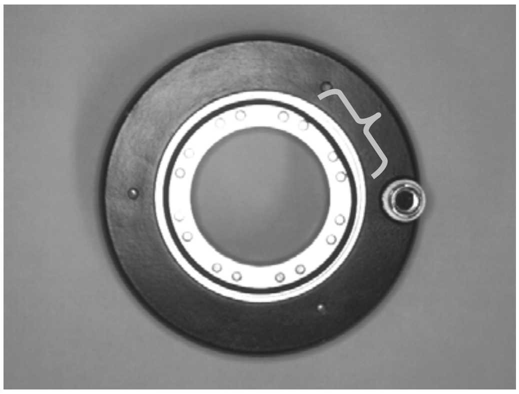


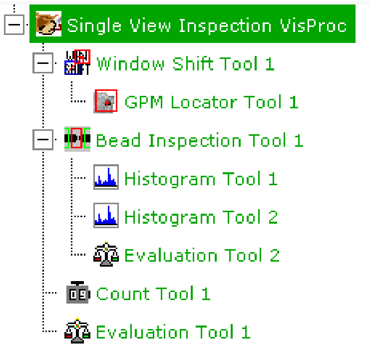
Fig. 5.8.4(b)

### Study for application

Consider which command tools are applicable for this application.

An O-ring is to be set exactly in a channel on the metal part. When it is set correctly the O-ring looks black and relatively white when the O-ring deviates a little from the channel. A Histogram tool will be used to measure the brightness along the channel on the metal part. Normally, an area which is slightly inside the chase will look white. If the O-ring deviates a little from a channel, its portion just inside the channel appears black. So, we also inspect the O-ring’s offset to measure the brightness just inside the channel. Histogram tool will be also used to measure the O-ring’s offset. In this example, two Histogram tools are used to inspect the position of an O-ring. As just described, Bead inspection tool can have multiple inspection or locator tools.
Of course, you might measure an area which is slightly outside the chase to inspect the O-ring’s offset.

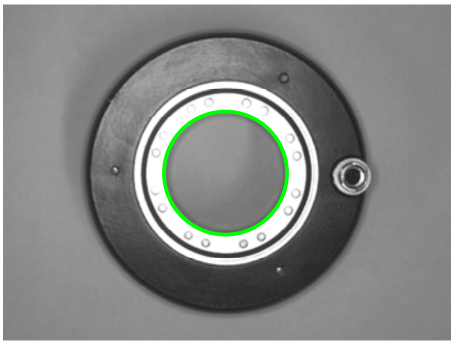
The metal part is roughly positioned, so use “Window Shift Tool” to shift the inspection points before inspection.




Use “Window Shift Tool 1” to shift the inspection points before inspection, and “GPM Locator Tool 1” to find the metal part. Use “Bead Inspection Tool 1” to generate inspection points along the channel on the metal part. Use “Histogram Tool 1” to measure the brightness at each inspection point and “Histogram Tool 2” to measure the brightness just inside the inspection point. “Evaluation Tool 2” to evaluate the result of “Histogram Tool 1” and “Histogram Tool 2” at each inspection point. Use “Count Tool 1” to count the number of FAIL, and the last “Evaluation Tool 1” to evaluate the PASS/FAIL for this inspection.

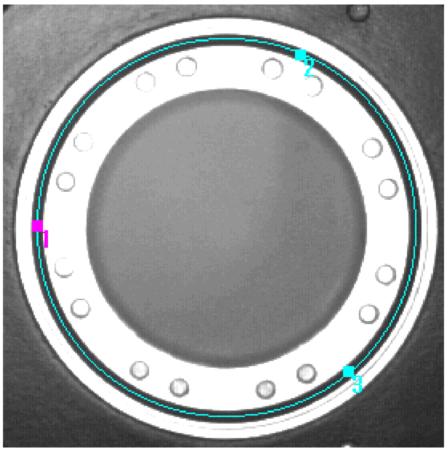
## Train GPM Locator Tool 1

An O-ring to be inspected and a feature on the metal part to be found should be on the same plane. So, train an inner circle on the metal part as the model of “GPM Locator Tool 1”, as shown in the figure below.

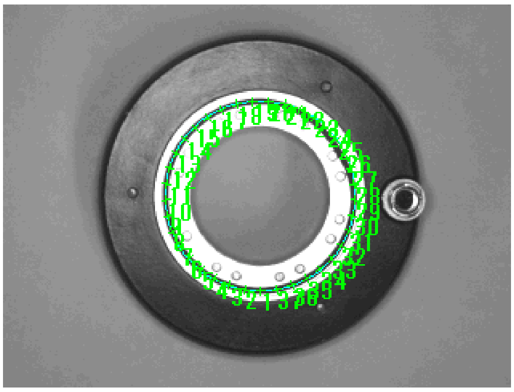


## Train Bead Inspection Tool 1

Train “Bead Inspection Tool 1” so that the inspection circle runs through the center of the O-ring you want to inspect, as shown in the figure below.

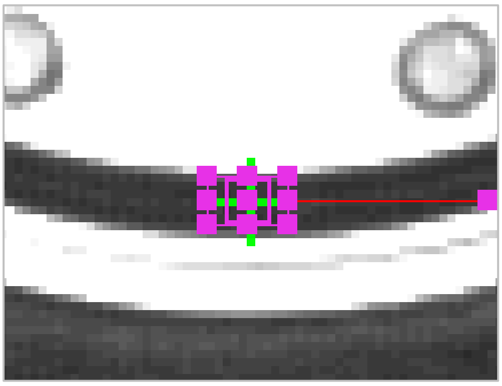


Set [inspection interval] you want to inspect in “Bead Inspection Tool 1”. First, you prepare a work piece with O-ring deviated from the channel, and snap and find in “Bead Inspection Tool 1”. As a result, you can confirm all of the inspection points plotted on the image. The following image shows the inspection pointes plotted on the image. Each cross-hair is inspection point, and you can see a few inspection points in upper right of the O-ring deviate a little from the channel. In this case, it may be a reasonable interval for inspecting the O-ring to be deviated from the channel. If you want more extensive evaluation, you can set more small value as interval. In this example, [inspection interval] is set to 20.



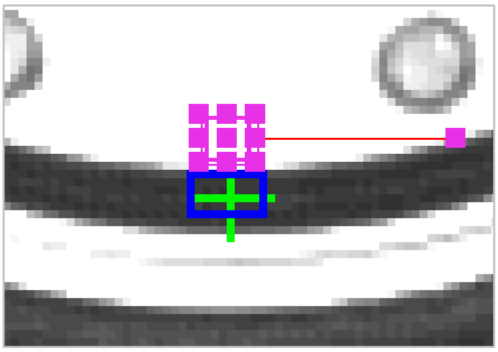
## Train Histogram Tool 1

Set the measurement area of “Histogram Tool 1” at first inspection point. The measurement area should be set in the channel on the metal part. The cross-hair shows the first inspection point and the rectangular shows the measurement area of “Histogram Tool 1”.



## Train Histogram Tool 2

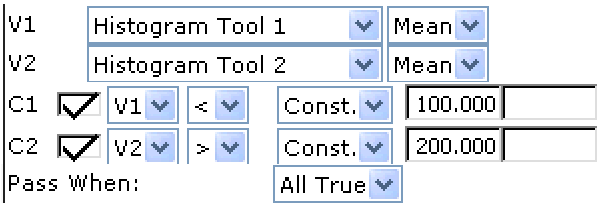
Set the measurement area of “Histogram Tool 2” at first inspection point. The measurement area should be set just inside the chase. The cross-hair shows the first inspection point and the rectangular shows the measurement area of “Histogram Tool 2.



## Train Evaluation Tool 2

In “Evaluation Tool 2”, the PASS/FAIL evaluation condition on each inspection point will be specified.

In this example, the histogram tool measurement results of the O-ring are compared to determine the criterion. Run a test for multiple work pieces using the histogram tool, to measure the mean brightness of the measurement area. The mean brightness of “Histogram Tool 1” was 70 or less for good portion and 130 or more for bad portion. And the mean brightness of “Histogram Tool 2” was 230 or more for good portion and 170 or less for bad portion. From this data, around 100, which is the middle value of 70 and 130 seemed to be an appropriate criterion for “Histogram Tool 1”, and around 200, which is the middle value of 230 and 170 seemed to be an appropriate criterion for “Histogram Tool 2”. Set the condition for “Evaluation Tool 2” to output [True] when the mean brightness of “Histogram Tool 1” is less than 100 and the mean brightness of “Histogram Tool 2” is more than 200 for all of the inspection pointes. Select [All True] for final evaluation condition of the evaluation tool.

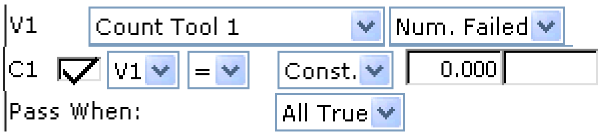



## Train Count Tool 1

“Count Tool 1” needs to count the PASS /FAIL number of all the inspection points. Leave “Count Tool 1” at default value setting since a count tool counts the total number of item evaluated by all sibling inspection tools that appears before it on the tree view.

## Train Evaluation Tool 1

Set the condition for “Evaluation Tool 1” to output [True] when [Num. Failed] of “Count Tool 1” is 0. Select [All True] for [Pass When] since only one condition is set.




## Running a test

Run the vision process and check that the result is correct.

For the following image in Fig. 5.8.4 (a), the condition is [True] because none of the result in “Histogram Tool 1” and “Histogram Tool 2” is evaluated as [Fail] with “Count Tool 1”. Since [All True] is selected for [Pass When], the O-ring passed the inspection.


| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Count Tool 1.Num. Failed = 0.000 | 0.000 | True |

For the following image in Fig. 5.8.4 (b), the condition is [False] because 7 results in “Histogram Tool 1” and “Histogram Tool 2” are evaluated as [Fail] with “Count Tool 1”. Since [All True] is selected for [Pass When], the O-ring failed the inspection.


| # | Cond. No | Condition | Value | Cond. Result |
|---|---|---|---|---|
| 0 | 1 | Count Tool 1.Num. Failed = 0.000 | 7.000 | False |

# INDEX

### *<T>*



### *<V>*



### *<W>*

# REVISION RECORD

| Edition | Date | Contents |
|---|---|---|
| 02 | Sep, 2013 | • Applied to series 7DC2 (V8.20P)<br>• Applied to R-30*i*B Mate |
| 01 | May, 2012 | |

B-83304EN-3/02

* B - 8 3 3 0 4 E N - 3 / 0 2 *