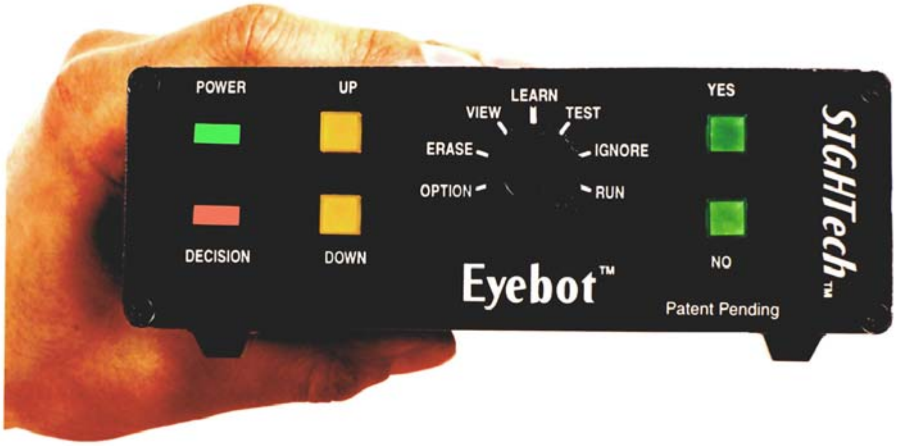
Eyebot™ Manual



Published 2004

Sightech Vision Systems, Inc.
1299 Parkmoor Ave.
San Jose, CA 95126
Tel 408.282.3770 • 800.992.4188
Fax 408.282.3501
Email sales@SIGHTech.com • www.SIGHTech.com

Information contained in this publication regarding device applications may be superceded by updates. No representation of warranty is given and no liability is assumed by SIGHTech Vision Systems, Inc. with respect to the accuracy or use of such information. Use of SIGHTech's products as critical components in life support systems is not authorized except with express written approval of SIGHTech. No licenses are conveyed, implicitly or otherwise. Eyebot and Neuro-RAM are trademarks of SIGHTech Vision Systems, Inc.

TABLE OF CONTENTS

Overview

Eyebot is a revolutionary device that allows you to take advantage of the state-of-the-art techniques of machine vision. No longer will you have to worry about the headaches of programming a machine vision system. It produces results faster than traditional machine vision approaches and it can learn to recognize parts in a matter of minutes (or seconds).

Features

- ✓ Self-learning machine vision system.
- ✓ No programming required—reduces setup time to a matter of minutes.
- ✓ Affordable and cost effective—can be used on a wider range of applications.
- ✓ Self-contained, requires no host computer.
- ✓ Compact size allows Eyebot to be installed almost anywhere.
- ✓ Can inspect up to 3,600 parts per minute (PPM).
- ✓ Does not need a strobe or trigger input for many applications.
- ✓ Can be used to detect defects or to identify one set of parts from another, as well as identify objects within cluttered backgrounds.
- ✓ Can recognize parts with full X/Y movement and full rotation.
- ✓ RS-232 serial and two optically isolated relays.
- ✓ Offers both Shape and Spectrum inspection.

Rather than developing code or dealing with complex computer vision software, Eyebot's installation is all completed via its intuitive push-button interface. SIGHTech has developed Eyebot based on the principle that machine vision should be simple and affordable.

Eyebot offers four inspection modes: SHAPE, SPECTRUM, COLOR and COLOR 2. You choose which is best for your application needs.

SHAPE mode allows Eyebot to learn and inspect shapes. Eyebot sees the world as millions of small edges, and alerts you when it sees new features that it was not taught. However, Shape mode does not learn colors.

SPECTRUM mode learns color spectrums, not shapes. Use Spectrum mode when you want to see a color difference and when you do not care about shape of product.

COLOR mode combines the Shape and Spectrum capabilities of Eyebot. It not only learns shapes, but also takes color into account. This mode generally weighs shape a bit more than color. This is useful when you want to see that both the shape and color are correct.

COLOR 2 mode also combines the Shape and Spectrum capabilities of Eyebot, but places more emphasis on color rather than shape.

To change between the inspection modes, turn Eyebot's knob to OPTIONS, select FEATURE and select MODE. (Refer to OPTION SETTINGS in the manual for detailed instructions.)

If you have a Multi-Session Eyebot, you effectively have eight Eyebot's in one unit. MS Eyebot offers the ability to customize each of the eight session settings and allows you to cycle through each session automatically. Most of the information pertaining specifically to MS Eyebot can be found within the OPTIONS heading under SESSIONS.

Quick Start Guide

Everyone should read this section to get a quick overview of Eyebot's operation. If this is the first time you are using Eyebot, we recommend that you read the sections that follow as well.

Plug and Play

✓ Connect the Eyebot between your video monitor and your video camera.

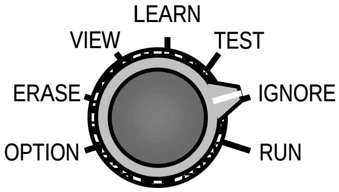
✓ If SIGHTech has supplied you with two wall transformers, use the wall transformer marked CUI to power the Eyebot. The other wall transformer should power the camera.

✓ Turn the Eyebot on. The green POWER light should come on. Adjust camera, lens, focus, and lighting to obtain a clear image.

Training to Inspect

1. Place a good part in the camera's field of view.
2. Turn knob to the ERASE position. Press YES. This erases memory.
3. Turn knob to VIEW. Press UP or DOWN to adjust the image so that you can clearly see the part and the details you want to inspect. The less complicated the image, the easier Eyebot will learn the part.
4. Turn knob to LEARN. Press YES. This will start the learning process. Small black dots on the image indicate areas of learning.
5. Observe the Score rise (the changing number on the bottom line). Continue training Eyebot until Score has risen to over 50 (the max is 99). If the part is stationary, this may take only a few seconds; if it is moving, it may take 2-10 minutes. Training should duplicate the conditions Eyebot will inspect under (so if there will be vibrations, tap the camera to simulate vibrations or other imprecision).
6. Turn knob to TEST position.
7. Place a defective part in the field of view of the camera and check to see if the Score drops.
8. Adjust the Decision Threshold to make sure that good parts are passing and bad ones are being caught. If this is not possible, make Eyebot LEARN more good parts by showing good parts and turning knob to LEARN, and pressing YES.
9. Return to step 5 and repeat until Eyebot is able to differentiate between good and bad parts.
10. When Eyebot is performing well in TEST, turn knob to the RUN position to activate outputs. This will allow Eyebot to send signals out of its RS-232 port and optically isolated relays to work. These outputs will allow you to activate a reject mechanism. Eyebot will also beep with defective parts.

Recognizing Parts (when to use IGNORE)



Most Eyebot users will use the inspection capacity of Eyebot. This means that Eyebot learns a process and whenever there is a deviation from what it learned it will tell you.

However, a few users may want to use the recognition capacity of Eyebot. You should use IGNORE when there is known universe of parts or images. *This works well when you know precisely all the parts Eyebot will face. If you cannot train Eyebot on all the types of parts it will face, do not use ignore.*

The goal is to LEARN Eyebot on the part(s) you want Eyebot to recognize and then IGNORE the part(s) you do not want it to recognize.

Eyebot will send out a signal whenever it recognizes the part(s) it learned; Eyebot will not send out a signal when it sees the part(s) it was taught to IGNORE.

For example, let us assume you make three types of soft drinks (Coke, Pepsi, 7UP). You want Eyebot to make sure that you are only make one type of can and that a wrong one does not slip in. There are two ways to solve this problem using IGNORE.

1. Train it to LEARN the two cans that are “bad” and IGNORE the one that it is “good.” Eyebot will send a signal whenever it sees one of the two “bad” cans. Whenever a “good” can passes, Eyebot will not send a signal.

2. Another way is to train it to LEARN the one “good” can and IGNORE the two “bad” cans. Again, Eyebot will send a signal whenever it recognizes the can it learned (the “good” one in this case) and it will not send out a signal on the ones that it was taught to ignore (the “bad” ones).

Most manufacturers would prefer using the first method because the result is more intuitive (i.e., a signal on a reject). However, either method is effective.

Once you start ignoring, you cannot return to LEARN. If there is something you want Eyebot to learn about the part you are training it to recognize, then you should ERASE and start over. The key to effective training when using IGNORE is to do as much learning about a good part at the beginning and then ignore all the differences. You can return to IGNORE as many times as you need.

Training to Recognize

1. Follow steps 1-9 above. Eyebot may learn more than one part.
2. Turn knob to IGNORE position.
3. Show the part(s) you want Eyebot to ignore on the screen.
4. Press YES. This will start the ignoring process. To stop or pause the ignoring process, press the NO button. The Score will drop as Eyebot learns to ignore the part(s).
5. Turn to TEST or RUN.
6. Place the first part (the learned part) in the field of view of the camera and check to see if the Score rises above the Decision Threshold.
7. Put one of the ignored parts in front of the camera and verify that the Score drops. If it does not, then start again from step 2 until Eyebot can differentiate between the learned part(s) and the ignored part(s). If Eyebot is unable to differentiate, change your setup and start over.

System Setup

This section will guide you through Eyebot's setup process.

The basic system setting up is shown in Figure 1. To set up Eyebot you will need the following components:

- ✓ Eyebot with a power supply.
- ✓ NTSC or PAL camera with a steady mount.
- ✓ BNC Video Cables to connect Eyebot to the camera and monitor.
- ✓ Monitor (needed only during training and is helpful to monitor the process).

You can follow these steps for an easy setup:

1. Connect the video output cable from the camera (or other video source) to the Camera Port on the back of Eyebot.
2. Connect the input cable from the monitor to the Monitor Port on the back of Eyebot.
3. Plug the power supply cables into an AC.
4. Turn on the POWER switch on the rear of Eyebot when you are ready for operation.

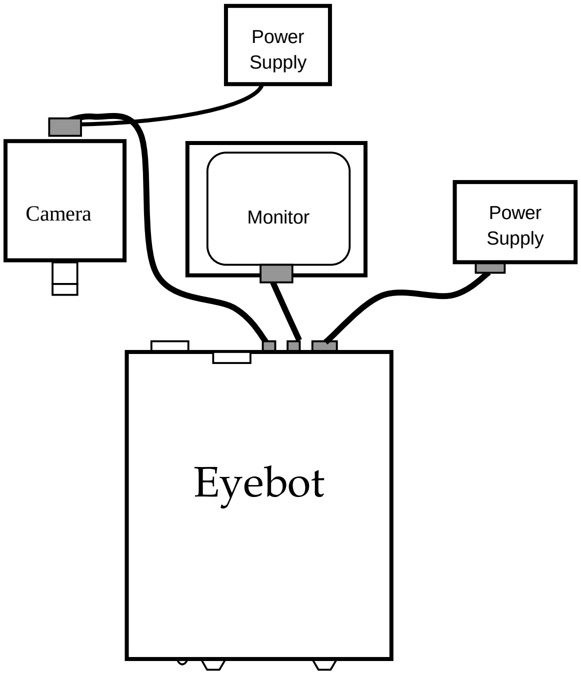
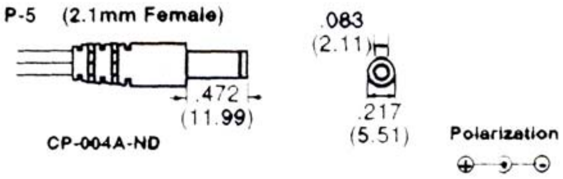


Figure 1 Eyebot System Connections

Power Plug

If you intend to use your own power supply or a European power supply, be sure that the inner center connector is +12 Volts with respect to the outer negative shell.

Electrical Ratings			Cord Plug
Volt.	MA	Reg.	
12VDC	**800**	**No**	**P-5**



Setting up the Camera

Mount the camera so it is stable—preferably on an adjustable mount so experimentation is easy.

Make sure the camera image is displayed on the monitor. Usually, it is best to focus specifically on the area you want to inspect. For example, if you want to want to verify that a car has a door handle, then you should make the door handle image fill up the video screen.

Importance of Lighting

In machine vision, lighting is frequently 90% of the battle (some would say 99%!). SIGHTech has designed Eyebot to make this battle relatively easy. Nevertheless, there are some applications that are challenging and demand attention to lighting conditions. Here are a few tricks that may help you use lighting to your advantage.

- ✓ Try different lights (halogen, incandescent, fluorescent, etc.).
- ✓ Try different intensities (by dimming or by approaching the part).
- ✓ Try different angles (to highlight grooves or eliminate Sparkle).
- ✓ Try to alter the lighting by moving your hand over the part you are viewing; once you find the optimum spot, place a permanent part to duplicate the soft-shadow your hand created.
- ✓ Work with a machine vision integrator that specializes in lighting.

Setting up the Part

Make sure the parts are oriented and placed so that the area of inspection is visible by the Eyebot. Use the monitor to determine this.

Minimize complex backgrounds—especially changing ones. Use a black or white backdrop if possible. Background interference can be managed by using a lens with a small depth-of-field and carefully focusing on the part while defocusing (blurring) the background.

If the parts are moving, try to fixture the movement in a consistent manner across the screen. Fixtured stationary parts are the easiest to inspect. Ideally, train Eyebot to LEARN and inspect the parts while they are stationary. The more aspects of a part Eyebot learns, the less discriminating it will be when inspecting. Ultimately, if Eyebot learns too many aspects, it will only catch gross defects.

Setting Fixture of the System

Eyebot has three option settings to designate the fixture mode for the parts to be inspected:

QUICK FIXTURED allows Eyebot to LEARN faster but loses some ability to detect small defects. Good for inspection processes containing parts with gross defects. QUICK FIXTURED is position independent like NON FIXTURED.

NON FIXTURED makes Eyebot relatively insensitive to X/Y movement of a part. This setting is best when you don't have much control over your parts during the inspection process.

HALF FIXTURED makes Eyebot relatively insensitive to right to left movement (on the screen), but very sensitive to up and down movement (on the screen). This is useful when you want Eyebot to detect minor up and down shifts (e.g., seeing a bottle cap coming loose).

FULL FIXTURED makes Eyebot very sensitive to both up and down and right to left movement. This is useful when you have a rigidly fixed part on an X/Y axis and it is important to detect minor changes.

Use HALF FIXTURED when parts are moving from right to left or from left to right on the monitor screen. Use FULL FIXTURED when parts are indexed into position. When using FULL FIXTURED, you will usually interrogate the relay output only when the part has stopped moving or use the Wake Up pin to signal Eyebot to start inspecting.

Most applications can and should be solved with NON FIXTURED. Some processes that seem very controlled are not, so NON FIXTURED works well in such circumstances.

To change the settings, switch the dial to OPTION and select FEATURE. Change the FIXTURED setting to QUICK FIXTURED, NON FIXTURED, HALF FIXTURED, or FULL FIXTURED. For more information on how to navigate through the OPTION menu, turn to the OPTION SETTINGS section.

For Spectrum Mode Users

In NON FIXTURED, Spectrum Mode learns all the acceptable color spectrums and applies them everywhere. It does not inspect shapes. Use NON FIXTURED only when you want to detect new colors. For example, if Spectrum Mode learns a US flag, it will pass a French flag since it has the same three colors.

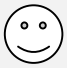
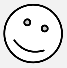
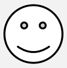
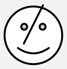
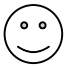
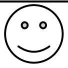
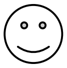
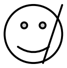
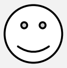
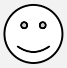
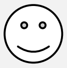
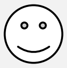
NON FIXTURED is useful in food processing, for example, where you want to learn the acceptable colors of French fries or tomatoes, and to fail when the colors start to change. Shape of the part is frequently irrelevant in food processing, but color can be critical.

Only FULL FIXTURED detects missing or misplaced colors. However, it depends on very tight fixturing of the part.

Examples of Different Fixture Settings

To help you decide what method is best for your application, look at Table 1 on the next page. In the examples, we will assume you are a manufacturer of smiling faces. In all the cases, you train Eyebot to learn the smiling face in the same X/Y position. You show it about 20 samples of the face in the same position. Here are the results of what will happen, depending on the setting on the Eyebot.

Table 1. Examples on Different Fixture Settings

Fixture Setting	Good Parts	Eyebot Will PASS	Eyebot Will FAIL
NON-FIXTURED			
Use this setting when part is not fixtured on an X or Y axis. Especially useful when there is some rotation. Eyebot can learn parts in various orientations.		Passes because it has not rotated and has no new features. Even though it is not positioned precisely on an X and Y axis, it does not matter because the setting is X and Y NON-FIXTURED.	Fails because it is either rotated or has a mark across it. Had the part been learned with +/- 20 degrees of rotation, then the top one would have passed, but it would still fail if it were at a 90° angle, for example.
NON-FIXTURED			
HALF-FIXTURED			
Use this setting when the part is fixtured on the X axis, but not the Y axis. Useful when parts are moving across the screen from the right to the left.		Passes because it remains on the Y axis. Moving up or down does not affect it because it is NON-FIXTURED on the Y axis.	Fails because either it has shifted off the Y axis (above), or a defect has appeared (below).
HALF-FIXTURED			
FULL-FIXTURED			
Use this setting when part is *precisely* fixtured on both the X axis and Y axis. Useful when parts are extremely controlled and you want to detect very subtle changes. It is rare that a process has sufficient control to employ this mode.		Passes because it remains on the X axis. Moving up or down does not affect it because it is NON-FIXTURED on the Y axis.	Fails because it moved (above), or a small defect has appeared (below, it's the dot above the mouth).
FULL-FIXTURED			

Further Explanation on QUICK-FIXTURED

When Eyebot is QUICK FIXTURED it is able to LEARN parts at a faster rate however it loses its ability to detect small defects. QUICK FIXTURED is useful for inspection processes containing parts with gross defects.

SIGHTech Tip

The majority of SIGHTech's customers use their Eyebots in NON FIXTURED mode.

Further Explanation on NON-FIXTURED

When Eyebot is NON FIXTURED, features are classified and saved into memory, without any regard to the physical location of the shape in the video image. If a part is learned in the upper right hand corner, that same part may pass in the lower left-hand corner. We say that it *may* pass because there may be subtle lighting differences from one corner to another corner of a camera view that may make the part appear different to the digital preprocessor. NON FIXTURED is useful since the position of an inspection part may be difficult or impossible to orient.

Further Explanation on HALF-FIXTURED

When Eyebot is HALF FIXTURED, the X coordinate corresponding to the vertical position is stored with the feature. This mode is typically used when parts are moving from side to side through the video image, like bottles on a labeling machine. A feature that is learned on the left side of the video image will pass inspection on the right side, but if learned on the bottom of the image, will not pass inspection at the top.

Further Explanation on FULL FIXTURED

In FULL FIXTURED mode, the features of a part and its position are combined and saved in memory. A part that is learned in the upper right hand corner will not pass in the lower left hand corner.

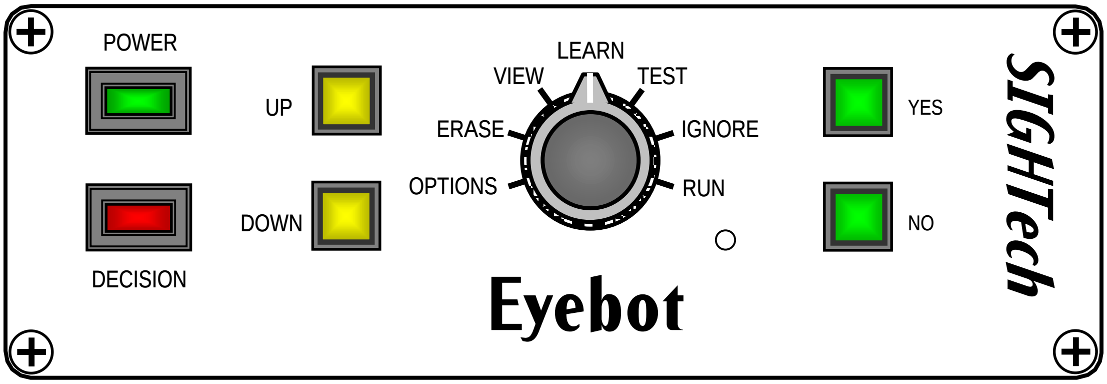
You may notice that a FULL FIXTURED application learns faster than a NON FIXTURED application, *if the part is stationary*. There are several reasons for this. One reason is that NON FIXTURED mode learns a higher complexity of feature details in order to properly identify it across the entire video image. A higher feature complexity takes more time to learn.

Another reason that FULL FIXTURED applications may learn faster is that they are less susceptible to VIDEO SPARKLE. VIDEO SPARKLE are small areas in your image that cannot be resolved with the video preprocessor and tends to generate a set of nearly random features. In a NON FIXTURED application, these random features will degrade the performance throughout the video image. In FULL FIXTURED mode these random features will be contained to that small area since the random features will have position information attached to it. A random feature learned in the upper right hand corner, for example, does not affect the performance in the lower left hand corner.

The final reason a FULL FIXTURED application generally learns faster than a NON FIXTURED application is that FULL FIXTURED applications are generally better designed. Usually, the amount of image variability is reduced because the inspection part is brought into a fixed and known position before learning or inspecting takes place.

Front Panel

Figure 2. illustrates the layout of Eyebot's front panel.



1. POWER Indicator: This green LED is lit whenever the Eyebot is receiving power and is turned on.
2. DECISION Indicator: This red LED is lit whenever the Score goes below the Decision Threshold.
3. UP/DOWN Buttons: These yellow buttons allow you to raise and lower values as well as navigate through the OPTION menu.
4. YES/NO Buttons: These green buttons activate the LEARN and IGNORE modes. Confirms ERASE. Button will light up whenever Eyebot is idle (NO) and active (YES). Also used to navigate in OPTION. The YES Button can also be used to step through various monitor viewing options in RUN, LEARN, TEST and IGNORE modes.
5. DIAL: This 7 position dial specifies which mode you want to enter. They are explained in Table 2. Press YES to confirm ERASE, LEARN, and IGNORE.

Table 2. Dial and Mode Summary

Dial Position	**Mode Explanation**
OPTION	Allows you to control several options, including changing fixtured setting, activating pixel counter, adjusting window size, and altering output delay. See OPTION section.
ERASE	Permanently erases memory when YES is pressed while in this position.
VIEW	Displays what Eyebot sees; pressing the UP and DOWN buttons will adjust the VIDEO THRESHOLD.
LEARN	When YES is pressed, Eyebot begins learning. NO will pause (or idle) learning. Used for training for either part recognition or part inspection.
TEST	Makes Eyebot stop learning and make decisions on what it sees. Score reflects Eyebot's decision confidence. Outputs are disabled.
IGNORE	Rarely used. When YES is pressed, Eyebot will learn to ignore whatever is currently passing through the camera. NO will pause (idle) this ignoring process. This is a special training mode for part recognition only; use this when you want to detect presence of features. This mode unlearns previous training. See "Recognizing Parts: When to Use IGNORE" section.
RUN	Same as TEST, but outputs are enabled.

Rear Panel

There is a power switch, a power jack, two BNC video connectors for camera and monitor, a serial connector and an external connector on the Rear Panel.

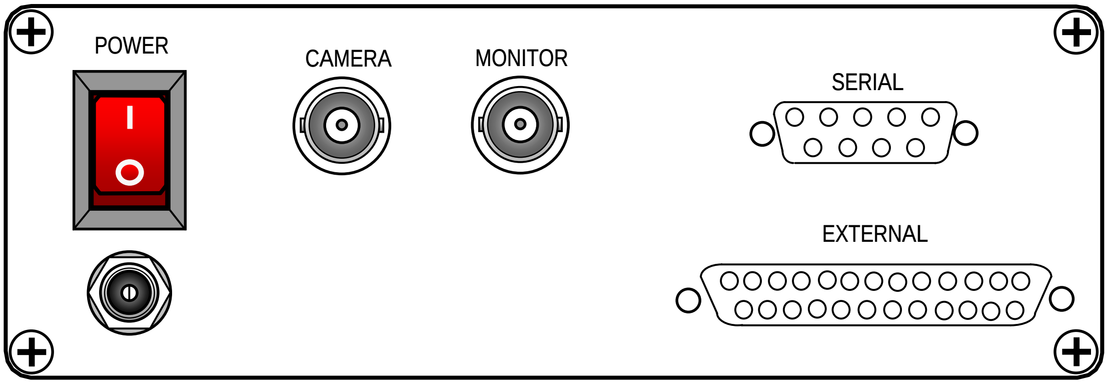


Table 3. Switch and Connectors Summary

ON / OFF Switch	Turns Eyebot on / off. ON = 1 and OFF = 0
POWER Plug	Plug for Eyebot AC Power pack adapter. 12-24 Volt DC.
MONITOR Plug	BNC Video output connector for NTSC or PAL monitor when needed.
CAMERA Plug	BNC Video input connector for NTSC or PAL camera.
SERIAL Connector	Female DCE RS-232 DB9 output connector for control purposes.
EXTERNAL Connector	Female DB-25 pin input/output connector for direct control purposes.

Table 4. Power Jack

Pin #	Description
Center	Unregulated 12-24V, 500ma, power input – Positive
Outside	Unregulated 12-24V, 500ma, power input – Negative

SERIAL (DB9) Connector

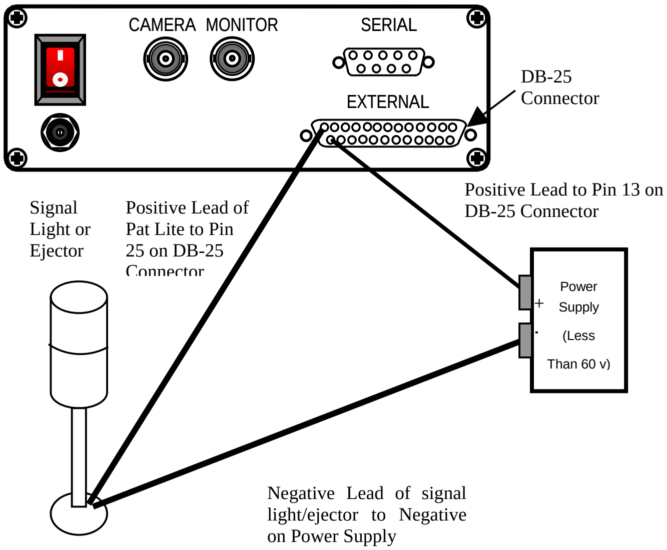
This is an isolated DB9 RS-232, DTE female connector so it can be plugged directly into a PC compatible serial port. It is a two-wire version of RS-232 – only RxD and TxD are active as shown in Table 7. The handshake lines are opened. If handshake lines need to be connected for your application, this must be done external to Eyebot

Table 5. Serial (DB9) Pins

Pin #	Name	Description
1	No Connection	
2	RxD	Serial Output from Eyebot to Host (PC, etc.) RS-232 +/-12V signal
3	TxD	Serial Input to Eyebot from Host (PC, etc.) RS-232 +/-12V signal
4	No Connection	
5	Serial Ground	Serial Signal Ground
6	No Connection	
7	No Connection	
8	No Connection	
9	No Connection	

OUTPUT CONNECTION

The diagram below shows a typical signal light output connection that utilizes the Normally Open Relay #1 contacts. These contacts are available on pins 13 and 25. An ejector mechanism can also be connected. Choose your signal light and power supply so that it does not exceed Eyebot's 3 Amp current limit. Pins 12 & 24, the Normally Closed contacts, can be used if you have a dual color signal light, and you want to have green for pass and red for fail.


CAMERA
MONITOR
SERIAL
EXTERNAL
DB-25 Connector
Positive Lead to Pin 13 on DB-25 Connector
Signal Light or Ejector
Positive Lead of Pat Lite to Pin 25 on DB-25 Connector
Power Supply (Less Than 60 v)
+
-
Negative Lead of signal light/ejector to Negative on Power Supply

EXTERNAL (DB-25) Connector

Eyebot has two optically isolated relays that provide decision outputs and Wake Up part-ready strobe input. This connector allows Eyebot to more directly interface with part lighting, movement, and processing. The figure below shows the connector as viewed looking at the back.

EXTERNAL

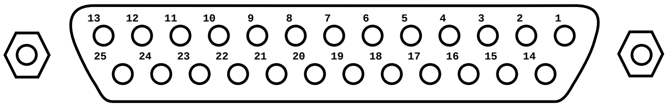
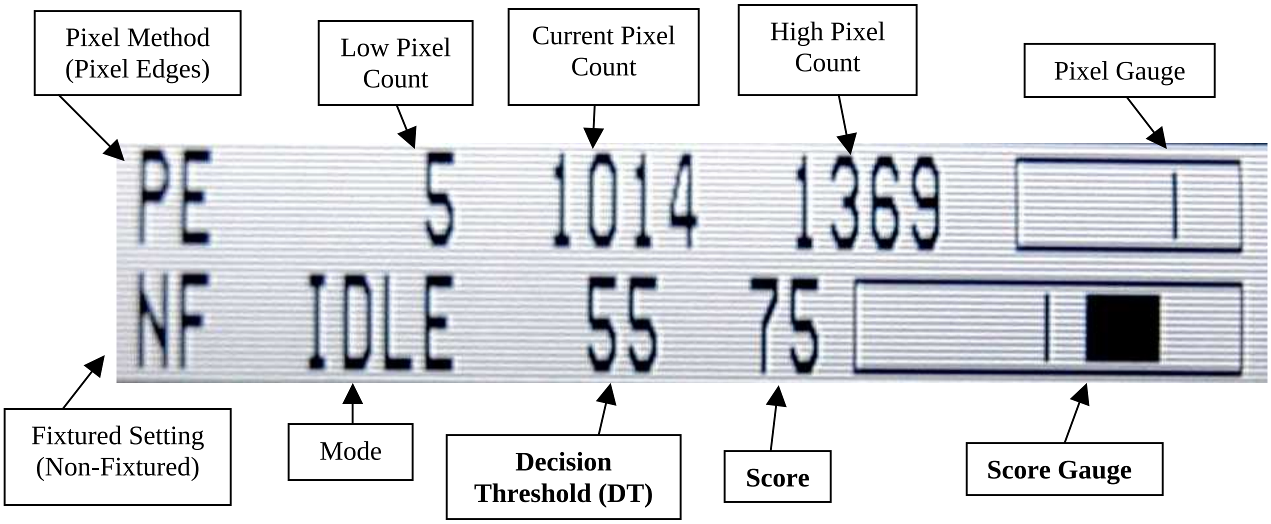


Table 6. DB25 Connector Pins on Rev E

Pin #	Name	Description
1	Common Ground	Use to ground Eyebot.
2	Option	Option select
3	B2 Session	Bit 2 of octal session select
4	B1 Session	Bit 1 of octal session select
5	B0 Session	Bit 0 of octal session select
6	Power +	Positive terminal unregulated 12-24V power @500ma
7	Power –	Negative terminal unregulated 12-24V power @500ma
8	Lock	Shorting this input to pin 11 disables the front panel.
9	RELAY_3	Reserved
10	Isolated Input +	Shorting this input to pin 11 wakes up Eyebot so operation can be toggled on/off. Set WAKEUP MODE to ON SWITCH CLOSED.
11	Isolated Input Ground –	Isolated Ground
12	Output #2	Relay Output #2 – switches 60V max, 3 Amp max, AC/DC. Normally closed.
13	Output #1	Relay Output #1 – switches 60V max, 3 Amp max, AC/DC. Normally open.
14	Yes	Momentary YES
15	Erase	ERASE mode select
16	View	VIEW mode select
17	Learn	LEARN mode select
18	Test	TEST mode select
19	Ignore	IGNORE mode select
20	Run	RUN mode select
21	Up	Momentary UP
22	Down	Momentary DOWN
23	No	Momentary NO
24	Relay #2 Return Ground	Common isolated return for **Relay Output #2**
25	Relay #1 Return Ground	Common isolated return for **Relay Output #1**

On Screen Display (OSD)

Make sure you have a good understanding of the OSD Status lines displayed on the screen before moving on. The Status lines appear on the monitor whenever you turn the dial to ERASE, LEARN, TEST, IGNORE, or RUN. It helps monitor the status of Eyebot all the time.



There are two Status lines. The top Pixel Line will not be visible if you have turned PIXEL MODE to OFF. The bottom line is the Neural-RAM Line:

Neural-RAM Line

- ✓ Fixture Setting: In the lower left hand corner of the white box you will see letters indicating the Fixture Setting of the Eyebot. The first two letters indicate how the Eyebot is Fixtured. NF indicates NON FIXTURED, HF indicates HALF FIXTURED, FF indicates FULL FIXTURED, and QU indicates QUICK. You can change this setting in OPTION under the FEATURE menu. With Multi-Session Eyebots, there will be a number after these two letters indicating the session number currently operating.

- ✓ Mode: Indicates what mode you have dialed. Some modes need confirmation by pressing YES. When you switch the dial to ERASE, LEARN, or IGNORE the mode indicator will read IDLE until you press YES. After hitting YES, the indicator will show the name of the mode you are in. Pressing NO will put Eyebot back to IDLE and pause learning or ignore. If you see READY, this means that the Eyebot is waiting for Wake Up.

- ✓ Decision Threshold: This is a value from 0 to 100. When the Score crosses this Threshold, Eyebot will fire its relays. In either LEARN, TEST, IGNORE, or RUN you can use UP and DOWN to adjust the Decision Threshold.

- ✓ Score: The Neuro-RAM Score indicates how many new (unlearned) features Eyebot sees. In LEARN and RUN Mode, the lower the value, the more new features, the higher the value, the fewer new features. The Score range is 0-99. Usually a high Score is a good part and low Score is bad part. If the Score crosses the Decision Threshold, Eyebot will signal a decision.

- ✓ Score Gauge: The purpose of the Score Gauge is to give you an easy graphic representation of how your parts are doing. The thin line in the middle represents the Decision Threshold, and does not move from the center, even if you do adjust the Decision Threshold. However, the score bar will change its relative position depending on Score **and** Decision Threshold. If you have a good part, the score bar will be to the right of center. If you have a bad part, it will be to the left. Ideally you want the score bar to be as wide as possible, indicating wide margins. Short score bars, or bars that are jumping from side to side, indicate marginal parts or bad training. How are you doing? Glancing at the Score Gauge, you have a quick answer.

Pixel Line

This line will not appear if Pixel Method is set to OFF.

- ✓ Pixel Method: A two letter abbreviation of the Pixel Method setting, which is defined in the PIXEL submenu of OPTION. Pixel Edges (PE) is the default; there are also Pixel White (PW), Pixel Dark (PD), and Pixel Neural (PN). The Pixel line showing PW or PD makes more sense when the VIDEO is set to INTENSITY. That is what these counting modes were designed for. If, on the other hand, you are using VIDEO EDGES, then PW corresponds to NON EDGES, PD corresponds to EDGES, and PE corresponds to EDGE OF EGES.

- ✓ Low Pixel Count: The lowest number of pixels Eyebot has recorded during LEARN.

- ✓ Current Pixel Count: Real-time number of pixels counted. If you are doing really fast applications, you may not see the Pixel Count register defective parts. Higher priority was assigned to the Pixel Gauge OSD update. You will see fast defects there.

- ✓ High Pixel Count: The highest number of pixels Eyebot has recorded during LEARN.

- ✓ Pixel Gauge: Graphic representation of pixel counts. When Current Pixel Count is within the Low-High range, then there is a thin line in the box. The closer the line is to left, means that it is near the Low Pixel Count; the closer the line is to the right, means that it is near the High Pixel Count. If the Current Pixel Count drops below the Low Pixel Count, a dark bar appears on the left; if the Current Pixel Count exceeds the High Pixel Count, a dark bar will appear on the right.

Extra Line

This line appears when you press YES twice. It was designed for fast applications where scores are often difficult to read on the monitor.

- ✓ Time Stamp: tells you how much time has elapsed since the Eyebot was turned on in hours, minutes, seconds and frames. Useful for documentation with a VCR.

- ✓ Worst Failing Score: tells you the worst failing score in TEST and RUN modes within the last two seconds.

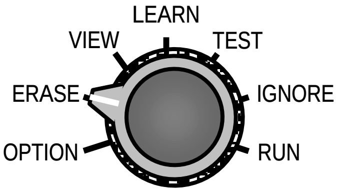
- ✓ Worst Score: tells you the worst score in TEST or RUN modes within the last two seconds.

- ✓ Reject Counter: indicates the number of times the relays have been activated. This number can be very useful in your final setup and timing of an actuator on a production line. You can reset this number by pressing the UP and DOWN buttons at the exact same time. When doing this, be careful that you don’t inadvertently change your Decision Threshold, which is also controlled by pressing the UP or DOWN buttons, but just one at a time.

Five Steps to Getting Up and Running

By now you should have your lighting and camera set up and understand usage of buttons and the dial on the control panel, therefore, you can turn on your convey with parts passing through the camera to train and run your Eyebot system for inspection with the following five step procedure.

Step One: ERASE

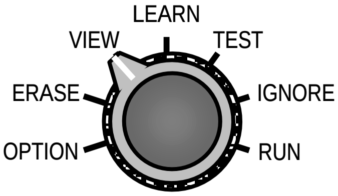


To begin an inspection session from erasing Eyebot's memory, turn the dial to ERASE and press the YES button. This will completely erase the information from any previous training that Eyebot has learned. ERASE clears away the training you have done with LEARN but it does not effect your settings under OPTION. For MS users, Erase only erases the current session.

Erase mode does *not* erase the system's settings, such as the window size, the output delay times, or the fixture setting. These user-changeable settings must be individually changed with OPTION. You can refer to the section entitled "Default OPTION Settings" to restore all the settings to their original values (the procedure is also outlined near the end of the "Troubleshooting" section).

Caution! Be careful with the ERASE since it will destroy all previous training!

Step Two: VIEW



After erasing, turn the dial to VIEW. View depicts the "World According To Eyebot". In a rectangle window on the monitor you see a threshold image or an edged image derived from the original image. The rectangle window is the area of inspection, the Eyebot's field of view. The image inside the window is exactly what Eyebot sees, learns and inspects.

Note! The objective is to image the features of the parts you want to learn and inspect in the VIEW mode. If you cannot see a difference between the good and bad parts in the VIEW mode, then Eyebot will be unable to see the difference.

Both the size of the window and the image within the window are adjustable.

The basic objective is to teach Eyebot the least amount of information it needs to do the job. Do not teach it things it does not need to know to do its job. To achieve this you can:

- ✓ Verify that the VIEW window is only looking at areas of the picture that are relevant to application. For example, if you just want to see the cap of a bottle, do not make your VIEW look at the whole bottle, just focus on the top. To adjust the VIEW window, go to OPTIONS and select WINDOW.

- ✓ Set your Video Threshold (in the VIEW) to be as dark as possible and yet still able to image the defect. Use the UP and DOWN buttons to adjust the Video Threshold in VIEW.

- ✓ By minimizing the information Eyebot must learn, your training time will decrease and your accuracy will increase. Keep this in mind when reading this manual.

- ✓ Image within the window is either the thresholded image (INTENSITY) or the edged image (EDGE). Most areas of the screen image should be solid black, the other areas represent the information of object parts. Some areas may be *glittering*. These glittering areas, called **Sparkle,** are random change occurring in an image near the threshold value. Avoid Sparkle because it quickly fills up Eyebot's memory and makes learning more difficult.

- ✓ You can wipe out most Sparkle by properly adjusting the Threshold by pressing UP/DOWN button. Although ideally you should try to eliminate Sparkle, it is difficult to eliminate it completely. Eyebot can handle learning some Sparkle.

An optimal image in VIEW shows just enough detail to see the defect in the VIEW rectangle, but nothing more. Just image enough detail to see the features you want Eyebot to learn and inspect.

Try the following experiment:

1. Place a good part while in VIEW and study the image. Then put a bad part under the VIEW, and ask yourself if you can you tell the difference between a good part and bad part by just looking at the VIEW window.

2. If the answer is no, then you must adjust the Threshold using the UP/DOWN buttons while in VIEW to bring out more details. Consider changing lighting or zooming in (or out) of the image.

SIGHTech Tip

Trying to get rid of Sparkle? If you have a part on a white background, try a black background instead, or vice versa. This may give Eyebot more contrast. Try adjusting the lighting.

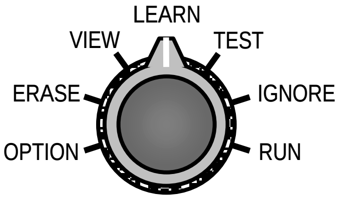
If the answer is yes, then continue darkening the image in the VIEW screen. This will simplify the amount of information Eyebot will have to learn. After making the VIEW darker, try step #1 again until you get an image where you can just barely see all the defects and features you want Eyebot to catch. You may have to put several bad parts under the VIEW screen to make sure that Eyebot will be able to see all your defects.

The point of the experiment above is to show you that you should not have Eyebot learn extraneous information, but just have it learn what is necessary to do the job. This will take some trial and error at first as you test to find the optimal Threshold setting in VIEW, but it will pay off when you actually set it up.

The VIEW adjustment is actually the *hysterisis* within the video preprocessor. It narrows or widens a video threshold dead zone. If a pixel's intensity falls within this dead zone, it will leave the state of the output as it was. If the dead zone is too small or zero, you will have Sparkle, which happens in gradually changing gray areas of an image. In these areas the preprocessor has a difficult time in determining correctly whether a pixel's intensity falls above or below the threshold. With unavoidable lighting variations or electronic noise, you will see black speckles or black sparkle in these gray areas.

Important: It is quite possible for you to see features that Eyebot cannot actually see when you are *not* in the VIEW mode. That is why it is important to pay careful attention of what you can see in the VIEW mode because this is what Eyebot actually processes.

Step Three: LEARN



This step is to make Eyebot LEARN. Turn the dial to LEARN, turn your conveyor on to have your good parts go through the camera's view, then press the YES button to activate LEARN Mode.

When the Eyebot is active in LEARN the Status Indicator appears on the monitor. While Eyebot is learning, you can notice the Score on the lower row of the Status Indicator rises from zero and will ideally progress to 99.

When Eyebot starts learning, the Score will have a large range; this is normal and indicates that there is a lot of Neural-RAM activity. The Eyebot can process up to 13 million features a second and processes all the video in real-time via its pipeline architecture. In a few seconds or minutes the Score's range will begin to stabilize at a higher range.

In LEARN Mode you want to set the Decision Threshold (DT) so that the Score is above the DT when it is a good part, and below the DT when it is a bad part. For example, let's say you get Scores that range from 75-95 when good parts go through. Therefore, you could set the DT at 70, so that when Eyebot sees something new or different, the Score will drop below the DT.

Important: Eyebot learns *everything* that goes through the VIEW rectangle. Yes, even your hand if you stick it in the field of view! Learning is cumulative and Eyebot is not just taking a few snapshots. There is no frame grabber in Eyebot. It is processing every video frame in real time and is making over 13 million micro decisions per second, which lead to one macro decision.

As the learning process continues the Neural-RAM activity subsides and the Score begins to rise. The higher the Score, the more Eyebot is trained.

When is Eyebot done learning?

Knowing when to stop learning may seem somewhat tricky because Neuro-RAM activity seldom goes away entirely. However, in most applications if the Score is consistently over 80, then Eyebot has learned enough.

How big of a defect am I trying to detect?

If it is a gross defect, then the Score may not even have to rise above 50. Why? Because you will get such a large dynamic range in the Score between a good part and bad part, that you can easily set your Decision Threshold with plenty of margin. However, if you are trying to see very subtle defects, then you may want to see Score staying consistently in the 90s before stopping learning.

Important: Only acceptable parts should be going through the camera view while Eyebot is in LEARN (unless you want it to learn bad ones). If a bad part should go through, some of its features will be assimilated into memory and these features will then pass in the future when Eyebot inspects.

The NO button allows you to pause learning. So if you can see or otherwise anticipate a bad part coming into camera view, press the NO button. Press YES after the bad part passes and more good parts appear. Having one bad part go by may not be catastrophic, but it will slightly degrade the inspection performance.

How long does learning take?

The more you features you demand Eyebot to learn, the longer it will take to learn it. For example, the more positions and orientations you make Eyebot learn, the longer the process will take.

Eyebot will usually learn stationary parts in a few seconds. Learning complex parts, such as a rotating dollar bill, might take 15 minutes or more. If the Score is below 50 and does not rise and/or is stuck in a zone, that means Eyebot is still learning the features in the image; however, if it does move after a couple of minutes, it means one of the following:

- ✓ There is too much sparkle interfering on the image (see VIEW section for how to correct this).
- ✓ The part and/or its movement are too complex.
- ✓ The part is not placed at the correct distance from the camera, or the camera is out of focus.
- ✓ Eyebot is out of memory (this is rare).

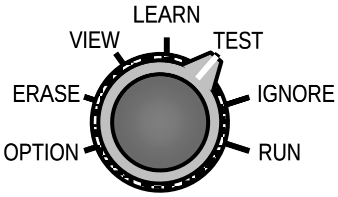
Important: Eyebot can still perform quite effectively even with partially learned parts. It is likely that you do not have to wait for the Score to hit the 90s for Eyebot to accomplish your application. If the training has taken it over 70, that is usually sufficient for most applications.

What do you see on the upper row in the Status Indicator?

While in LEARN the upper row in the Status Indicator shows three numbers: the lowest value, the real-time value and the highest value. During the LEARN phase, Eyebot keeps track of how many pixels there are on the screen and records the lowest and the highest values while counting.

During RUN, when counted pixel number is in the range, there will be a thin bar going back and forth within the zone. If Eyebot detects that the number of pixels drop below the low value, the gauge will signal a defect by highlighting the left side of the gauge, and Eyebot will fail the part. If the pixels go above the high value, the gauge will highlight the right side and Eyebot will fail the part.

Step Four: TEST



After learning, turn the dial to TEST. In this mode, Eyebot will stop learning and start inspecting. The outputs are disabled in TEST mode; if you want to activate the external outputs turn Eyebot to RUN.

Under TEST mode Eyebot will score all the parts based on what it has learned during the LEARN phase. The good parts (the parts you have taught Eyebot) obtain higher scores and the parts with defects are scored low.

While in TEST mode you can set and adjust the Decision Threshold. The Decision Threshold represents the level of strictness you want to set. If the Score goes lower this Decision Threshold Line, Eyebot's red DECISION light will turn on and Eyebot will activate its outputs if in RUN.

You can adjust the Decision Threshold using the UP and DOWN buttons when Eyebot is on either the LEARN, TEST, IGNORE, or RUN modes. You can make Eyebot stricter in its decisions by moving Decision Threshold value UP; to make Eyebot more lenient in its decisions push the DOWN button.

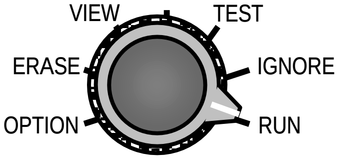
Using the UP and DOWN buttons, you set the Decision Threshold value slightly below the good part Score range. For example, after monitoring the process for a couple of minutes under TEST, you have observed that the good parts Score ranges from 80 to 95, in this case, 80 represents the marginally good parts and 95 are the best parts. Therefore, you would set the Decision Threshold around 75 (to give a little margin of error).

At this point you should test Eyebot by placing a defective part on screen and observing if the Score drops below the Decision Threshold.

If the Score of a defect part does not cross with a significant margin, you should re-adjust the Decision Threshold with UP/DOWN, so that the Score will go lower than Decision Threshold when a defect appears, and will rise above it when it is a good part. If this is not possible, you have to return to Step Three and train Eyebot some more.

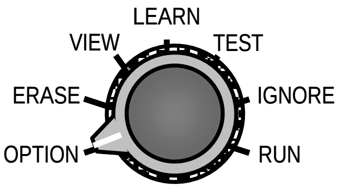
Eyebot's learning is cumulative; therefore, returning to learn will not erase previous learning, but rather will build on the previous learning experience.

Step Five: RUN



Assuming, that after TEST phase you find your Eyebot system can consistently distinguish between good and bad parts, then set Eyebot to RUN. Eyebot will start inspecting and it will activate the external outputs. In other words, RUN is the same as TEST, except the outputs are activated, including the Beeper**.**

Navigating the OPTION Settings



Many parameters in Eyebot can be adjusted such as the window size, the output delay, the video threshold, and the feature control. You can change these parameters by turning the dial to OPTION and follow the on-screen menu.

SIGHTech has set the default parameters to the most common settings.

Turning the dial to OPTION, you will see the OPTION Menu.

Note! If you turn the knob very quickly, Eyebot will not register it; so if the Menu does not pop up, put the dial to OPTION a bit slowly.

- ✓ Press UP and DOWN to go up and down the menu to chose (highlight) the item you want to check or adjust;

- ✓ Press YES when you want to modify a parameter or enter a submenu, this will highlight the parameter that you will adjust;

- ✓ Use UP and DOWN to adjust the parameter you have chosen;

- ✓ Your change is saved automatically pressing either YES or NO and exiting that setting. Or simply switch the knob to another position (e.g., TEST or VIEW). There is no way to cancel your adjustment and return to the previous value. Therefore, you may want to note the value before you change it, in case you want to reset it later.

WINDOW

This allows you set the size of the inspection Window. Eyebot only inspects what is in the VIEW Window, so it is important to adjust the Window to optimize performance. You can only have one VIEW Window and it must be a rectangle in shape. If you have an MS Eyebot, you will have one VIEW Window per session. You can adjust the Window by using the following keys:

TOP press UP to raise the top of the VIEW rectangle and DOWN to lower it.

BOTTOM press UP to raise the bottom of the VIEW rectangle and DOWN to lower it.

LEFT press UP to make the left of the VIEW rectangle to move right and DOWN to move it left.

RIGHT press UP to make the right of the VIEW rectangle move right and DOWN to move it left.

HITS determines the color of the neural RAM activity or “hits” on the monitor. Can be black hits on a gray and white background that shows the video thresholding or red hits on normal video.

LEARN

FEATURE allows you to set which method Eyebot uses to learn the information it sees.

SHAPE mode will see the world as having millions of small features. These features consist mainly of edges. It learns these small features in relation to other small features and automatically compiles a massive database of known features. It will always seek to new features and it will tell you when it sees them. Shape does not learn colors.

SPECTRUM is when you want Eyebot to learn colors, not shapes. Spectrum mode can learn thousands of colors and it will tell you when it sees new colors. Therefore, you show it what are the acceptable color spectrums, and Eyebot will tell you when it sees new ones. The more tightly you train Eyebot while in Spectrum mode, the more sensitive it will be. Spectrum mode does not learn shapes. The edges you see in VIEW mode are irrelevant.

COLOR mode combines the Shape and Spectrum capabilities of Eyebot. In other words, it not only learns shapes, but also takes color into account. This mode generally weighs shape a bit more than color. This is useful when you want to see that both the shape and color are correct.

COLOR 2 mode also combines the Shape and Spectrum capabilities, but places more emphasis on colors rather than shapes.

FIXTURED allows you control how sensitive Eyebot is to XY positioning and fixturing. Please see the "Setting Fixture of the System" section for details on which setting to use. There are the four settings:

QUICK is used when your inspection processes contain parts that have gross defects. The advantage to the QUICK FIXTURED setting is that it enables Eyebot to learn at a faster rate.

NON learns a feature and applies it everywhere. In other words, it will be relatively insensitive to up and down, right to left movements of the parts. It will still be sensitive to rotation. This is the most forgiving FIXTURE setting.

HALF is used when you want Eyebot to be very sensitive to up and down movements on screen.

FULL is used when you want Eyebot to be very sensitive to up and down, and right to left movements on screen.

VIDEO

EDGES turn on the edge detector processing. This is the default video threshold method. Use EDGES if you are working with 3D parts that have different levels of gray and/or uneven lighting. You can raise or lower the value in OPTION or in VIEW Mode.

INTENSITY allows you set a fixed video threshold throughout the image. INTENSITY can be used in two extreme cases. It can be used when the image is very defined, that is, an evenly lit image with sharp contrast. Or it can be used in the other extreme, where an image is blurred so much that the EDEGES processing doesn't work.

THRESHOLD Eyebot maintains two threshold settings, one for EDGE processing and another for INTENSITY. Only one THRESHOLD will show for the selected VIDEO setting. You can adjust the THRESHOLD while in the OPTION menu or in VIEW mode.

TIMED allows you to set up a length of time you want Eyebot to learn. Eyebot will return to RUN after the specified amount of time (in fields) has elasped. This setting is useful if you want your floor operators to be able to relearn small changes in product or parts.

PIXEL

MODE selects the mode of pixel counting that Eyebot will use. Your *s*ettings include OFF, NEURAL, and EDGE. If you go to the ADVANCED menu and select VIDEO INTENSITY, you also get WHITE and DARK. In LEARN mode Eyebot automatically counts the number of pixels in the video image and determines an upper and a lower limit. Then, in RUN or TEST, if either pixel limit is exceeded the relays will fire. Pixel counting is especially important to determine if the part is actually there.

Eyebot can inspect an image of a black circle better by counting the number of DARK pixels, rather than counting the transition pixels on the edges.

MODE

OFF turns the pixel counter off. You will notice that the pixel gauge will disappear. Eyebot will NOT detect the absence of features. If you want to detect the presence or absence of a label, then set the PIXEL MODE to EDGE.

NEURAL directs the raw Neural-RAM Score to the pixel counter. Consider using this setting if you want to set a high and low limit to a training session. This works well while using the Spectrum mode. For example, Eyebot could learn the orange color and then IGNORE other colors, and then Eyebot could tell you if there is not enough orange present (or too much of it).

EDGE counts the numbers of white-to-black or black-to-white pixel transitions. This setting is the most robust in terms of lighting variations and should be used first.

WHITE counts white or light pixels. Consider using this setting if your parts are white and on a black background. WHITE and DARK are only available when VIDEO INTENSITY is selected under the OPTION ADVANCED menu.

DARK counts dark or black pixels. Consider using this setting if your parts are dark and on a white background.

HIGH is the value of the greatest number of pixels Eyebot saw during LEARN. This value corresponds to the value on the HIGH Pixel value that Eyebot displays in LEARN, TEST, IGNORE, and RUN. To manually adjust this value, use UP and DOWN. You will get very different HIGH values depending on the METHOD of pixel counting you use (see above).

LOW is the value of the lowest number of pixels Eyebot saw during LEARN. This value corresponds to the value on the LOW Pixel value that Eyebot displays in LEARN, TEST, IGNORE, and RUN. To manually adjust this value, use UP and DOWN. It is common that this value is 0 because if there ever is a blank screen during LEARN, Eyebot will set the low pixel value as 0, since there are no pixels on a blank screen.

TRIGGER is used to set a pass band beneath LOW. It was designed to allow the use of strobe lights or camera triggers where certain video fields will be absolutely dark. These dark fields will generate zero or very low pixel counts but, with a small value for TRIGGER, will not generate a fail situation.

RELAYS

MODE allows various relay options to be selected.

OFF turns the relays off. Off means that relay #1 (pins 13 and 25) is open circuit, and that relay #2, the complimentary relay, (pins 12 and 24) is closed. The relays are solid state and optically isolated.

TIMED adjusts timing parameters affecting the relay outputs.

DELAY controls the amount of time between the time when Eyebot sees the defect and when Eyebot turns on the relays.

MIN specifies a minimum relay output time interval. The timings of these settings are in video fields or 1/60th of a second for NTSC users and 1/50th of a second for PAL users.

MAX specifies a maximum relay output. Setting MIN equal to MAX will make the relay output a set length of time. If MAX is OFF then the relays can be activated as long as the Eyebot is seeing something bad. **MIN** overrides **MAX** if you make a mistake and set **MIN** higher than **MAX**.

TIMED ADVANCED further adjusts the relay output timing.

FILTER can be used to filter out fleeting Neural-RAM activity. In general, it is a length of time that the Eyebot must see something bad before the output responds. If the camera were pointed at a door, for example, FILTER could be adjusted so that a person could open, go through, and close the door, without a response output. However, if he left the door open for a longer period of time, then Eyebot would turn on the output.

PAUSE specifies an amount of recovery time after a relay output where the relays will be off for a known minimum length of time.

TRIGGER indicates whether the Eyebot should fire its relay outputs after a MAX specified timeout. If TRIGGER is set to LEVEL, Eyebot will refire after a MAX timeout or will fire when Eyebot is initially turned on. With MAX, PAUSE, and TRIGGER LEVEL you can specify a repeating relay output pulse. These timings are again measured as video fields.

REVERSE flips the decision around so that relays fire on good parts. This has been useful in applications where one learns an object, for example a label, then ignores the background, and then the Eyebot will fire the relays when the Eyebot doesn't see the object.

SHOT is a mode where the output relays fire for 2.5ms during any field where the decision has come on. This is our fastest mode, although you will have to latch the results and deal with timing issues. The settings under SHOT still refer back to TIMED MENU 1.

LATCH is used when you want the Eyebot alarm to sound, and stay on until a person clears the problem. If you press YES, the relays will fire indefinitely (as will the buzzer). A person must press NO to make Eyebot stop firing its relays.

WAKE UP

These parameters control the Wake Up pin of the DB-25 connector that controls whether or not Eyebot is, effectively, asleep. An indexing machine, for instance, should send a signal to the Wake Up pin when it has stopped moving. As soon as Eyebot detects activity on the Wake Up pin, Eyebot will either LEARN, TEST, IGNORE, or RUN. When Eyebot is waiting for a Wake UP it indicates READY on the Status line.

MODE

OFF Wake Up is disabled.

ON SWITCH CLOSED WAKE UP occurs when the Wake Up pin is electrically shorted as with a switch or an NPN transistor.

ON SWITCH OPEN WAKE UP occurs when a switched is opened.

MIN adjusts the minimum length of time the Eyebot wakes up. This is helpful if the signal from your conveyor line or indexing machine is too short.

MAX adjusts the maximum length of time the Eyebot wakes up. Normally, the Eyebot runs as long as the WAKE UP pin is on. When the WAKE UP pin is turned off, the Eyebot stops. If, for some reason, the manufacturing line stops and the WAKE UP pin is left on, **MAX** guards against the Eyebot being on too long. MAX may be a useful parameter to set for LEARN since it guarantees each part an equal amount of training time.

DELAY is a dwell time from the point when your machine first sends the signal, to the time the Eyebot Wakes Up. This parameter is useful if a part needs to settle into a permanent and stable inspection position just after a sensor has detected its arrival.

ADVANCED

RESPONSE TIME determines the number of video fields Eyebot sees before making a decision. The lower the number of fields, the faster Eyebot will go. Although faster may always seem better, the quality of a decision decreases as you make Eyebot make faster decisions. Indeed, humans generally make a better decision if they see a part for 2 minutes rather than 2 seconds. The default speed is 18**,** which is 18 video fields, or 9 video frames, or about a third of a second (NTSC). At the default speed Eyebot can easily handle 200 Parts Per Minute (PPM). If your application needs higher speed, the Eyebot system can work at the speed of up to 3600 parts per Minute.

BEEP

OFF turns the internal beeper off.

ON DECISION beeps when Eyebot sees something.

ON RELAYS beeps when the relays are activated.

GAIN adjust the value of the video to digital decoder gain. This setting is useful if you have very controlled lighting conditions.

SWEEP varies the video threshold around the VIEW setting during LEARN mode. This setting may improve the robustness of the learning when light varies slowly over time.

SESSION (only available with MS Eyebot)

Multi-Session Eyebot (or MS Eyebot) allows you teach Eyebot up to 8 training sessions. Think of it as the same as having 8 Eyebots in a box. MS Eyebot only works with one camera at a time.

NUMBER is the number of the session you are currently using. To work on a different session, use UP and DOWN to select a session 1-8. MS Eyebot cannot have more than 8 sessions.

CYCLE tells MS Eyebot if it should include the selected session as part of an automatic inspection cycle. If you want it to include the current session in the automatic inspection cycle, select ON; otherwise, select OFF. For example, if you want to inspect 3 areas of a part automatically, you could train MS Eyebot to learn one area per session. Then you would include all three sessions in the auto inspection cycle by selecting ON for each session. MS Eyebot would then automatically cycle through each session and make a decision.

DWELL sets the dwell time of the selected session. Each session can have its own dwell time. Each unit represents one video field.

When you use the Wake Up pin in conjunction with session cycling, Eyebot will start with the first active (CYCLE ON) session upon receipt of the Wake Up signal. Eyebot will then proceed to the next active session after the specified dwell time.

OPTION OVERVIEW

Table 7 summarizes all of the setting available under OPTION. Range and default settings are noted. You can restore the default settings of the OPTION menu by holding the UP button for 2 seconds while you turn power to the Eyebot on. The YES or NO under MS (Multi-Session) indicates whether or not the setting can be set independently for each session. If you are you using the serial port to communicate to your Eyebot, the corresponding serial command is listed under the column SERIAL.

Table 8. OPTION MENU SETTINGS

MENU	SETTING	RANGE	DEFAULT	MS	SERIAL
WINDOW	TOP	0 - 240	210 (NTSC)	NO	WT value
	BOTTOM	0 - 240	32	NO	WB value
	LEFT	1 - 720	100	NO	WL value
	RIGHT	1 - 720	620	NO	WR value
	HITS	BLACK RED	BLACK	NO	HB HR
LEARN	FEATURE	SHAPE COLOR COLOR2 SPECTRUM	SHAPE	YES	FA FB FC FS
	FIXTURED	QUICK NON HALF FULL	NON	YES	FQ FN FH FF
	VIDEO	INTENSITY EDGES	EDGES	YES	VX 0 VX 1
	THRESHOLD	OFF – 250 (INTENSITY) OFF – 150 (EDGES)	100 (INTENSITY) 20 (EDGES)	YES	VI value VE value
	TIMED	OFF - 65530	OFF		ST value
PIXEL	MODE	OFF NEURAL EDGE DARK (INTENSITY) WHITE (INTENSITY)	EDGE	YES	PF PN PE PD PW
	HIGH LIMIT	OFF - 65530	OFF	YES	PH value
	LOW LIMIT	OFF - 65530	OFF	YES	PL value
	TRIGGER	OFF - 65530	OFF	YES	PT value
RELAYS	MODE	OFF TIMED TIMED ADVANCED SHOT LATCH	TIMED	NO	RO RT RA RS RL
	DELAY	OFF - 65530	OFF	NO	RD value
	MIN	OFF - 65530	OFF	NO	RN value
	MAX	OFF - 65530	OFF	NO	RX value
	FILTR	OFF - 65530	OFF	NO	RF value
	PAUSE	OFF - 65530	OFF	NO	RP value
	TRIGGER	EDGE LEVEL	EDGE	NO	RT 0 RT 1
	REVERSE	OFF ON	OFF	NO	RR 0 RR 1
WAKE UP	MODE	OFF ON SWITCH CLOSED ON SWITCH OPEN	OFF	NO	WF WC WO
	MIN	OFF - 65530	18	NO	WN value
	MAX	OFF - 65530	OFF	NO	WX value
	DELAY	OFF - 65530	OFF	NO	WD value
ADVANCED	RESPONSE TIME	1 - 18	18	YES	DT value
	BEEP	OFF ON DECISION ON RELAYS	ON DECISION	NO	BF BD BR
SESSION	NUMBER	1 - 8	1	YES	SN value
	CYCLE	OFF ON	OFF	YES	SR 0 SR 1
	DWELL	OFF - 65530	120	YES	SD value

Using the Serial Port

Although Eyebot does not require a PC, you may connect it to one via the Serial port (DB9 RS232), and control your Eyebot via RS232 by running a terminal emulation program Procomm on your PC. Set the communications parameters to 19,200 N81.

MS Eyebot Upload/Download Functions

To save a part into a host system from any Emulation Terminal Program:

1. Type MO Linefeed (Mode Option)
2. Get a log file ready to receive data
3. Type MU Linefeed (Memory Upload)

To restore a part to the MS Eyebot:

1. Type MD Linefeed (Memory Download)
2. Send file

An example using Procomm from a DOS window:

To save a part:

1. Get a file ready to receive
    a. Type ALT+F1
    b. When it asks for a log file enter in a part name
2. Type **MU Enter** (Memory Upload)
    a. After 10 minutes the upload is complete
    b. Type ALT+F1 to close the file

To restore a part:

1. Type **MD Enter** (Memory Download)
2. Send a file
    a. Press PageUp
    b. Choose ASCII
    c. Enter the file name

Since July 26 2000, serial communications are interrupt driven and more than one character per field may be sent. All the buttons and the dial on Eyebot Front Panel are still active while sending serial. Control commands to Eyebot can come from the serial port or the Eyebot Front Panel. The last commands that Eyebot receives will determine Eyebot's operation.

Table 9 is a list of serial commands, the range of values that can be entered, and the description. Serial commands must end with a linefeed, or with the hex value 0D hex. If the range is not specified in Table 9 then no value need be entered. For example, MR Linefeed will switch Eyebot into Run Mode.

Do not send DL or @ unless you have new code on hand that you wish to load. @X will restart the Eyebot.

List of Serial Commands

Table 9 Serial Commands

XX	RANGE	DESCRIPTION
BD	-	BEEP ON DECISION
BF	-	BEEP OFF
BR	-	BEEP ON RELAY
DL	-	DOWNLOAD CODE
DS	0–100	DESCISION SENSITVITY
DT	1-18	DECISION RESPONSE TIME
FA	-	FEATURE SHAPE
FB	-	FEATURE COLOR
FC	-	FEATURE COLOR2
FS	-	FEATURE SPECTRUM
FF	-	FIXTURED FULL
FH	-	FIXTURED HALF
FN	-	FIXTURED NON
FQ	-	FIXTURED QUICK
HB	-	HITS BLACK
HR	-	HITS RED
MC	-	MEMORY CHECKSUM
MD	-	MEMORY DOWNLOAD (MS)
MU	-	MEMORY UPLOAD
ME	-	MODE ERASE
MI	-	MODE IGNORE
ML	-	MODE LEARN
MO	-	MODE OPTION
MR	-	MODE RUN
MT	-	MODE TEST
MV	-	MODE VIEW
PD	-	PIXEL DARK
PE	-	PIXEL EDGE
PF	-	PIXEL OFF
PH	0-65530	PIXEL HIGH LIMIT
PL	0-65530	PIXEL LOW LIMIT
PN	-	PIXEL NEURAL
PW	-	PIXEL WHITE
RA	-	RELAY TIMED ADVANCE
RD	0-65530	RELAY DELAY
RF	0-65530	RELAY FILTER
RG	0-1	RELAY TRIGGER
RL	0-1	RELAY LATCH

XX	RANGE	DESCRIPTION
RN	0-65530	RELAY MIN
RO	-	RELAY OFF
RP	0-65530	RELAY PAUSE
RR	0-1	RELAY REVERSE
RS	-	RELAY SHOT
RT	-	RELAY TIMED
RX	0-65530	RELAY MAX
SD	0-65530	SESSION DWELL (MS)
SL	0-65530	LEARN TIMED
SN	1-8	SESSION NUMBER (MS)
SR	0-1	SESSION CYCLE (MS)
VE	0-50	VIDEO EDGE THRESHOLD
VG	0-500	VIDEO GAIN
VI	0-250	VIDEO INTENSITY THRESHOLD
VS	0-10	VIDEO SWEEP
VX	0-1	VIDEO MODE
WC	-	WAKE UP ON SWITCH CLOSED
WD	0-65530	WAKE UP DELAY
WF	-	WAKE UP OFF
WO	-	WAKE UP ON SWITCH OPEN
WN	0-65530	WAKE UP MINIMUM
WX	0-65530	WAKE UP MAXIMUM
WB	0-240	WINDOW BOTTOM
WL	1-720	WINDOW LEFT
WR	1-720	WINDOW RIGHT
WT	0-240	WINDOW TOP
@	-	DOWNLOAD CODE
#	-	SERIAL RAW NEURAL ACTIVITY
$	-	SERIAL SCORE
%	-	SERIAL PIXEL COUNT
^	-	SERIAL LINEFEED
&	-	SERIAL DIAGNOSTICS
*	-	SERIAL PBN. FIELD RESULTS
!	-	SERIAL RESET
>	-	UP
<	-	DOWN
]	-	YES
[	-	NO

Trouble Shooting

Make sure to visit SIGHTech's web page (www.SIGHTech.com) for answers to some Frequently Asked Questions (FAQ). The list below is a sample of some of the most common questions.

Eyebot does not come on—the green LED does not light up.

Check AC power pack connection to Eyebot. Turn on power switch. Double check that all cables are connected and that the monitor, camera, and Eyebot are all getting power.

There is no video display.

Check power connections and video cables. Are the cables on the Eyebot connected correctly – is the monitor plugged into the monitor jack? Is the Eyebot turned on – the switch is on the rear. Is the Monitor adjusted properly? Is the camera's power connected? Plug the camera directly into the monitor to see if the video system is working.

The Score does not move.

Make sure you have pressed the YES button when either in LEARN or IGNORE. This activates the mode. You must also press YES to confirm that you want to ERASE. To pause learning or ignoring, press NO.

Eyebot does not see signal a defect when there is nothing on the screen.

Make sure that the PIXEL Counter is on. Turn on the PIXEL counter by turning to OPTION, selecting PIXEL, then MODE, and then select EDGE. ERASE and retrain Eyebot. Make sure that Eyebot does not learn anything on the screen or the pixel counter will not work, because it will learn that no pixels are acceptable. The pixel counter learns the acceptable range of pixels while Eyebot is in LEARN. When placed in TEST or RUN, Eyebot will alert you when either there is a large jump in the number of pixels or a large drop. The more narrow range of pixels you show Eyebot, the more effective the pixel counter will work. The Pixel Gauge shows you the high and low values.

System trains for a long time without the Score rising.

Try erasing Eyebot and starting over. Look at the VIEW Mode so you can see, on the screen, what Eyebot sees. Is the image too complicated or is it moving too much in a non-repeatable fashion? If so, try better fixturing it or try using the Eyebot's NON FIXTURED Mode for training and operation.

Eyebot learns a part in one position in the center of the screen and when I move the part to another area of the screen, Eyebot signals a defect (i.e., the Score drops).

First, verify that Eyebot is in NON FIXTURED mode.

Second, even if it appears that your lighting setup is uniform, it usually is not. As a result, when you move the part to another area of the screen, small new features appear because the light hits the object in a different way and causes new shadows, which causes the new features.

The solution is to LEARN the part in various areas of the screen. Be careful to not put your hand in the screen while Eyebot is learning, because Eyebot will learn you hand, which may diminish the performance. Another trick is to tap the camera to cause vibrations, so that Eyebot is less sensitive.

Third, it is possible that when you move the part to another area of the screen that you placed it in a slightly different orientation, which Eyebot never learned. Therefore, have Eyebot LEARN the part with a little bit of realistic variation and changes in orientation.

I think I'm filling up Eyebot's memory.

It is very rare that a customer fills up memory. Eyebot's Neuro-RAM algorithm is extremely memory efficient and so the on board battery backed-up SRAM is sufficient for nearly all applications.

Eyebot fills up memory when it learns a complex part in various forms, positions, and orientations. By making Eyebot learn a part in one position, it can spot defects that are nearly imperceptible to the human eye. However, the more positions Eyebot has to learn, the more generalized its learning becomes and the more it fills memory.

There are three ways to avoid filling up memory:

1. Decrease the number of forms, positions and orientations that Eyebot has to learn.
2. Eliminate *Sparkle* (see above **VIEW** section).
3. Increase Eyebot's memory (contact SIGHTech).

I have tried everything and my Eyebot is still behaving strangely!

Try turning it on and off. If this does not work, you can *reinitialize Eyebot*. When you reinitialize Eyebot you will restore Eyebot to its defaults. For example, all your window and delay settings will be lost. To reinitialize Eyebot you must:

1. Turn off Eyebot.
2. Hold the UP button.
3. Turn Eyebot on (while holding the UP button). You should hear a beep.
4. Release the UP button about two seconds after you turned Eyebot on (it is OK to hold it down for longer).

I still have some unanswered questions.

Please visit our web site at www.SIGHTech.com and select the FAQ hyperlink. There you will find a long list of Frequently Asked Questions (FAQ).

Support

Integration Support

If you need assistance in integrating Eyebot with your needs, please contact SIGHTech. We can either help you directly or point you to a SIGHTech Approved integrator or Reseller.

Phone and Online Support

With your purchase of Eyebot you are entitled to unlimited technical telephone and email support. For telephone support, please call 408.557.6773 or 800.992.4188. For online support, please visit SIGHTech's web page at www.SIGHTech.com, and/or write to Support@SIGHTech.com. If you purchased Eyebot through a SIGHTech reseller, please contact the reseller.

Feedback

SIGHTech would like to know what you think of Eyebot and/or this manual. Please contact SIGHTech on your comments. We take your feedback seriously and would like to know how we can help you.

Eyebot Specifications

Processor	Proprietary ASIC
Training time	0.1-10 minutes
Response Time	1/3 of second or 200 PPM. Maximum is 3600 PPM.
Input / Output video	NTSC or PAL
Connections	RS-232 Serial Output and Two Optically Isolated Relays which switch 3 amps at 60 volts
Dimensions	14cm x 23cm x 4cm
Power	12 to 24 volt DC