

**THIS APPLICATION CAN BE DONE USING:**

**PRODUCTS**

Block Diagram
RIDE
ImageDSP
VIDSP Studio
VIDSP Suite
OORVL Design Studio

# Image Applications with Hypersignal®

## Overview

Hypersignal products including Block Diagram with the optional Image Processing Library, RIDE® with the optional Image Processing Library, and ImageDSP (formerly ImPro Lab) allow for a variety of image processing applications to be performed on the PC in a Windows 98/95/NT environment. The direct support of low-cost standard video CCD cameras allows for a number of interesting 'live' video processing applications. This paper will briefly introduce some of them. This application note describes four applications using a standard video frame grabber – a color detector (red), a motion detector, an object tracker, and an object counter. These applications can be built using a standard low-cost CCD camera connected to the PC (typically through parallel port, USB, or an internal video camera interface card [ISA, PCI, etc.]).

## Product Specific Information

These applications may be done with either Hypersignal RIDE or Block Diagram and the optional Image Processing Library, or with the ImageDSP product.

## Example 1: A Color Detector (red color detector)

The first example, *red\_dtct.lst*, is a red detector. The worksheet is saved as the first image below. The knob is used to adjust the Image Threshold block, adjusting the sensitivity of the red detection.

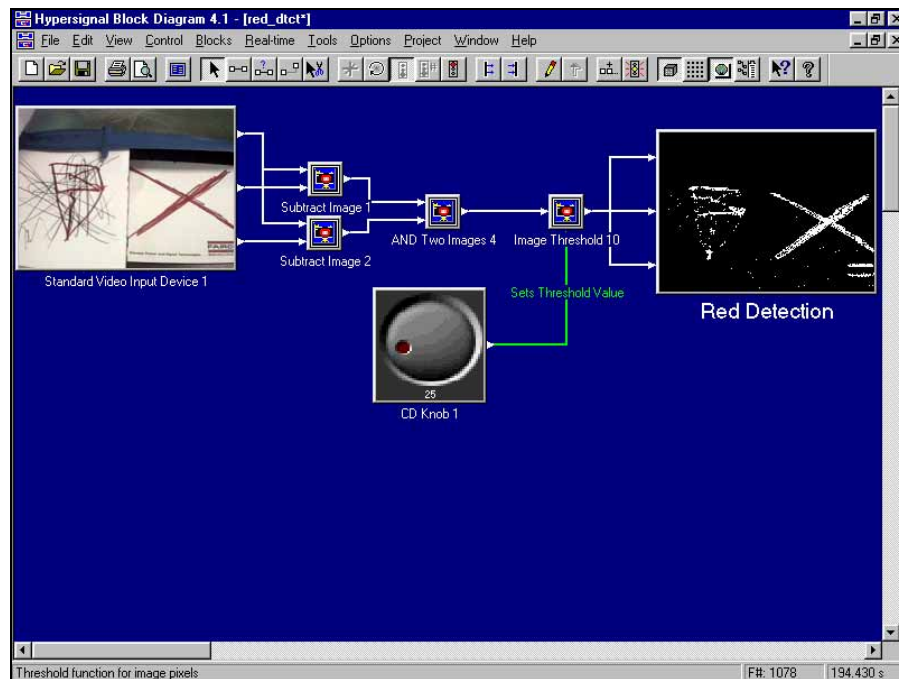
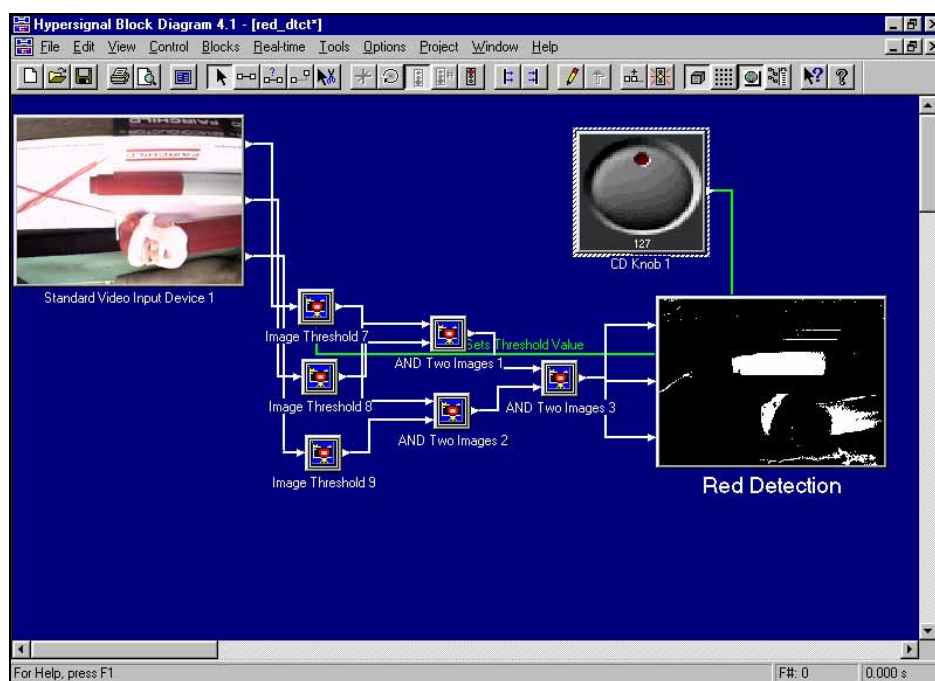


Figure 1 - Red Detector Block Diagram Worksheet using inverted G, B color planes

In the above worksheet, Figure 1, the Red Channel (desired color) is used along with the inverted (negated through the subtract block function) Green Channel as one input to the Image AND function. The other input to the AND is the Red Channel combined with the inverted (negated through the subtract block function) Blue Channel. The resulting output of the AND function is an image which is inherently RED and NOT GREEN and RED and NOT BLUE, or just RED. Notice the threshold block used to set the discriminating level for the RED color.

The following worksheet, Figure 2, is another Red Detector that was designed in a different manner using Logical AND blocks and Image Threshold blocks on all color planes. It works in a similar fashion, with the Image Threshold parameters set to favor the color Red and to discriminate against other colors.



**Figure 2 - Red Detector Block Diagram Worksheet using threshold logic**

Both of the above worksheets use a standard CCD camera as its image source (note standard frame grabber function). In addition, with the Professional Editions of the Hypersignal products, these worksheets can easily become stand-alone applications using the included Application Builder - HAppl Wizard.

This worksheet, built in Hypersignal Block Diagram, requires the optional Image Processing Library. The optional library would be necessary as well if built in Hypersignal RIDE. If this worksheet were to be built in the Hypersignal ImageDSP environment, no optional libraries are necessary as Hypersignal ImageDSP is designed towards image processing applications such as this as well as many others.

## Example 2: A Motion Detector

The worksheet below, Figure 3 - *motiondet.lst*, is a motion detector which uses a Delta Frame technique to calculate the energy of motion. The knob is used to adjust a global parameter, *Sensitivity*, adjusting the sensitivity of the motion detected. A File Read block is set to play a wave file when motion is detected and will continue playing the wave file in an infinite loop until motion is not detected; this effectively allows for an 'Annunciator' type of user output. The display labeled "Area of Motion" accents in the color red the area of the image where motion was detected.

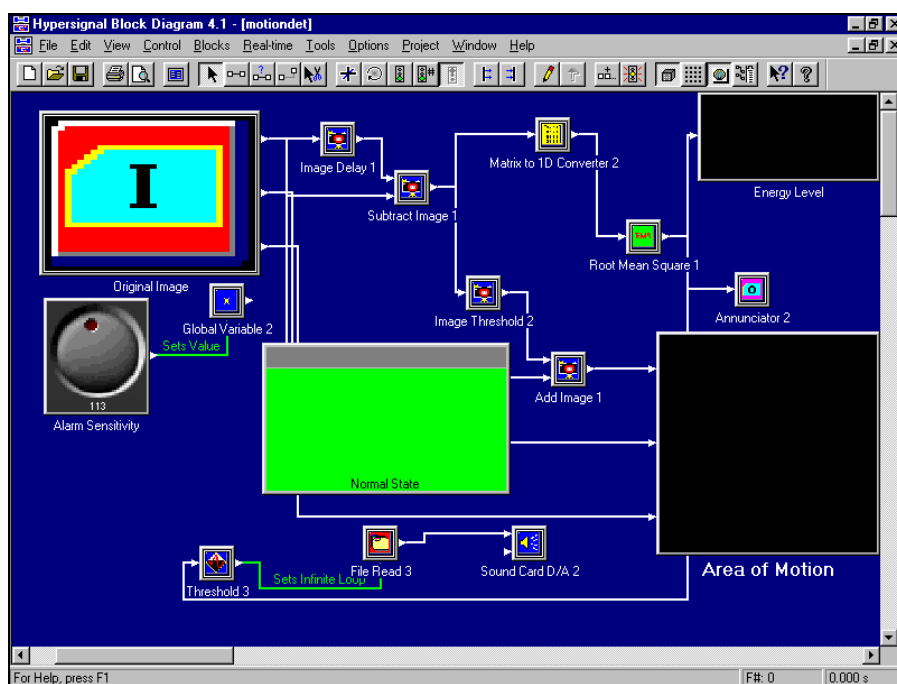


Figure 3 – Motion Detection Example using a Delta Frame technique

Again, this worksheet can easily become a stand-alone application for Windows 98/95/NT with Professional Editions using the Application Builder - HAppl Wizard. After setting the necessary blocks in place and running the worksheet through the HAppl Wizard, we have created the below application which will automatically work on any PC which has a standard PC-based Video Camera attached (and a sound card for the audio portion of the application).

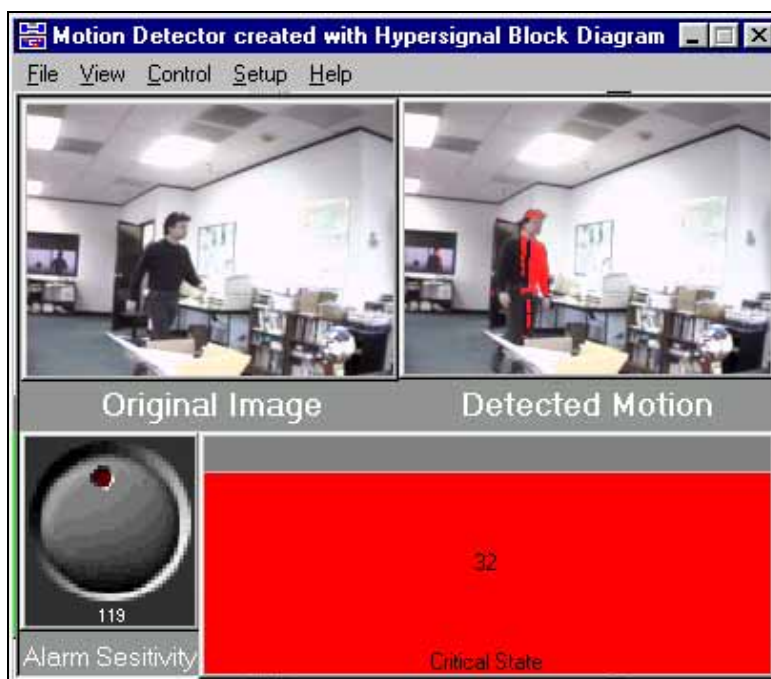


Figure 4 – Stand-alone application created with Professional Editions

Remember, this worksheet, built in Hypersignal Block Diagram, requires the optional Image Processing Library. The optional library would be necessary as well if built in Hypersignal RIDE. If this worksheet were to be built in the Hypersignal ImageDSP environment, no optional libraries are necessary as Hypersignal ImageDSP is designed towards applications such as this.

### Example 3: Object Tracking

The below worksheet, Figure 5 - *obtrckr.lst*, is an object tracker. This worksheet will automatically identify a moving object based on the parameters of the Object Tracker block and either identify it by merely highlighting the detected object or both highlighting the object and placing a cross-hair at the center of the object. As seen in this worksheet, the two digital displays in the lower right-hand corner of the worksheet represent the row-value and the column-value of the center of the detected image. The upper left-hand corner of images within any of the Hypersignal environments is represented as (0,0).

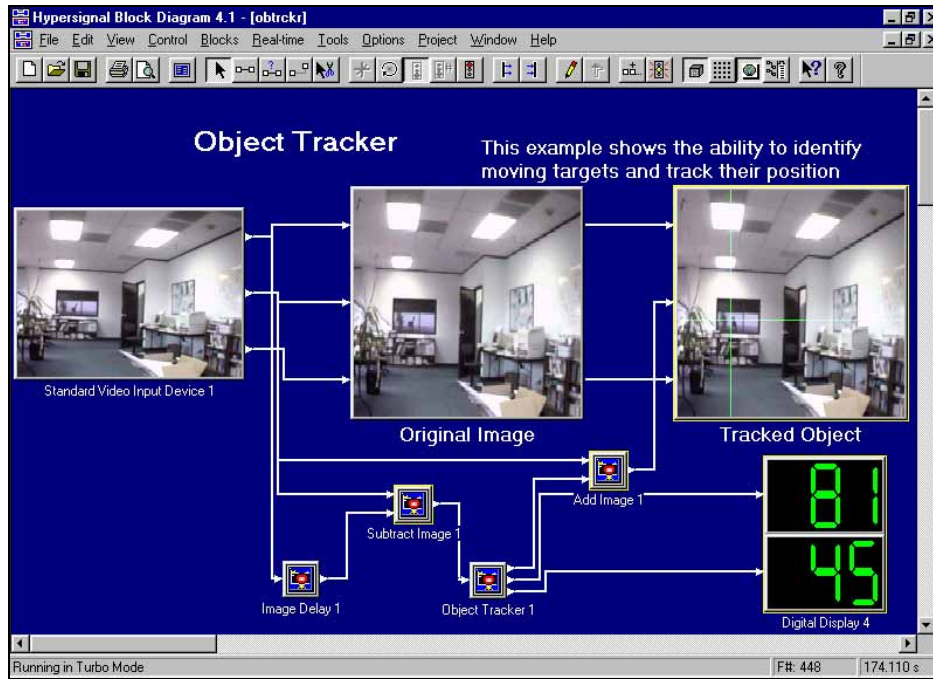


Figure 5 – Object Tracker for Image Processing

In the case of this worksheet, the Object Tracker block is set to be so sensitive that a small part of the original image, which happens to be a reflection in a window, is detected as having a moving object. Again, this worksheet uses a standard CCD camera as its image source. As in the previous examples, this worksheet can easily become a stand-alone application using the Application Builder - HAppl Wizard found in the Professional Editions of Hypersignal software products. Again, this worksheet requires the optional Image Processing Library.

### Example 4: Object Counting

In the final example, the worksheet in Figure 6 - *machvisn.lst*, implements an object counter. This worksheet identifies and counts the number of objects in the image based on how the user sets the parameters of the Object Counting block. The Object Counting block detects and counts objects of a certain size, specified by the parameters Minimum Area and Maximum Area. This block contains two output channels. Output Channel 0 is a single value (an integer) containing the number of objects detected. Output Channel 1 is an image with the same dimensions as the input channel; the output image "highlights" the objects that were detected and counted.

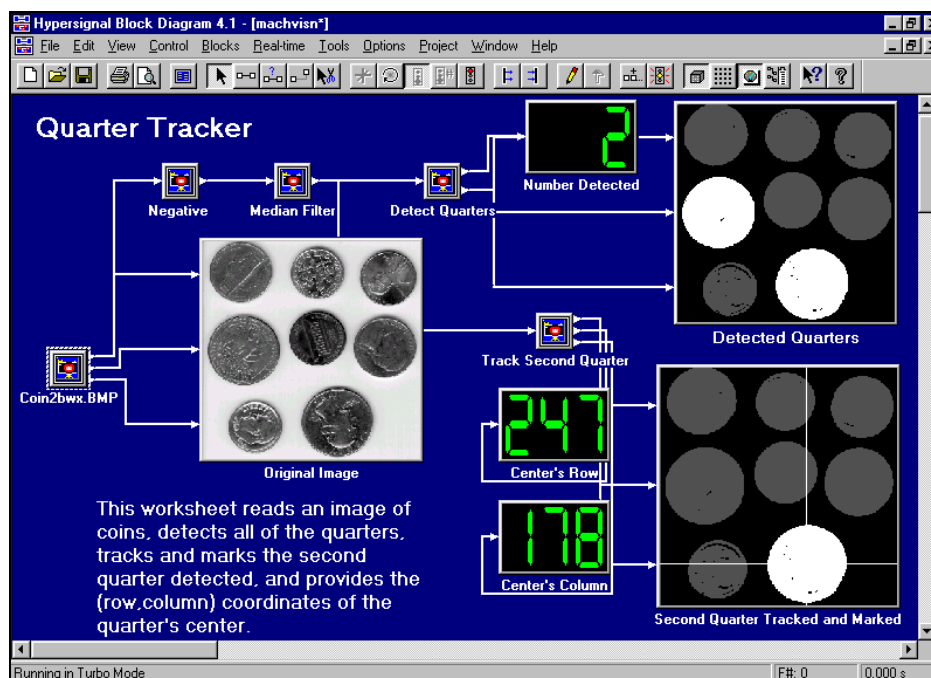


Figure 6 – Object counting example, counting coins

In the case of this worksheet, the Object Tracker block is used to detect the second quarter from the top of the image. This worksheet uses a bitmap file as its image source and is typically included as one of the standard examples with the Image Processing Library or the Hypersignal ImageDSP development environment.

Again, this worksheet requires the optional Image Processing Library and can easily become a stand-alone application using the Application Builder, HAppl Wizard, which is included in the Professional Editions of Hypersignal software products.

## Applications

The above examples show some of the many applications possible using the Image Processing Library in conjunction with Hypersignal RIDE or Block Diagram, or the ImageDSP product. Applications in Machine Vision, Image Compression, Recognition, and many other Research and Development projects are addressed with these products.

## References

1. Hypersignal RIDE User Manual, Hyperception, Inc., 1998.

**Hyperception**

*The Leader in DSP*

9550 Skillman LB125  
Dallas, Texas 75243  
Tel: (214) 343-8525 \* Fax: (214) 343-2457  
E-Mail: [info@hyperception.com](mailto:info@hyperception.com)  
[www.hyperception.com](http://www.hyperception.com)

Hyperception is continually improving and modifying its product line, and reserves the right to change the specifications in this product information sheet at any time, without notice. While the utmost care and precaution have been taken in the preparation of this application note, Hyperception assumes neither responsibility for errors or omissions, nor any liability for damages resulting from the use of the information contained herein. Hypersignal is a registered trademark and RIDE, OORVL are trademarks of Hyperception, Microsoft and Windows are registered trademarks of Microsoft Corporation.