

Real-time, PC-based Color Fusion Displays

15 January 1999

P. Warren, J. G. Howard*, J. Waterman, D. Scribner, J. Schuler, M. Kruer

Naval Research Laboratory, Code 5636
Washington, D.C. 20375

ABSTRACT

Color fusion has been developed to simultaneously display multi-spectral data to the human viewer, for the purpose of target detection, discrimination, and identification. Real-time capability of a color fusion system allows interactive laboratory testing of issues such as band selection and comparison of fusion algorithms. Proof of the real-time capabilities of the system is necessary to expedite transition to the fleet.

NRL had developed two inexpensive systems for displaying color fusion algorithms real-time with PC's and COTS hardware. The systems are general and capable of processing data from any cameras, but are demonstrated with specific infrared and visible cameras. The infrared and visible cameras are bore-sighted with no common optic.

With these two systems, it is possible to rapidly change camera combinations and/or fusion algorithms to capture data with multiple system arrangements at the field site. Viewing the fused data, as it is collected, allows us to capture a variety of interesting phenomenology which demonstrate the advantages of color fusion.

1. Introduction

This paper presents low-cost, adaptable hardware and software systems to study color fusion of newly available infrared and visible cameras. The final product will be a color fusion display for human visualization. In Section 2, the hardware for two real-time color fusion display systems, which have been built and demonstrated, are described. In Section 3, the computational tasks of the systems are described, from reading data from the camera to displaying the fused image to the viewer. Also, the fusion algorithms are introduced. In Section 4, the performance of the systems is described. Section 5 summarizes the paper.

2. Two hardware configurations for real-time display

Two hardware approaches to solving the problem of creating a real-time color fusion display are presented in this paper. Each system has its distinct advantage. Both systems read data from cameras

* On-site NRL contract employee affiliated with Raven, Inc.

Form SF298 Citation Data

Report Date <i>("DD MON YYYY")</i> 15011999	Report Type N/A	Dates Covered (from... to) <i>("DD MON YYYY")</i>
Title and Subtitle Real-time, PC-based Color Fusion Displays		Contract or Grant Number
		Program Element Number
Authors		Project Number
		Task Number
		Work Unit Number
Performing Organization Name(s) and Address(es) Naval Research Laboratory, Code 5636 Washington, D.C. 20375		Performing Organization Number(s)
Sponsoring/Monitoring Agency Name(s) and Address(es)		Monitoring Agency Acronym
		Monitoring Agency Report Number(s)
Distribution/Availability Statement Approved for public release, distribution unlimited		
Supplementary Notes		
Abstract		
Subject Terms		
Document Classification unclassified	Classification of SF298 unclassified	
Classification of Abstract unclassified	Limitation of Abstract unlimited	
Number of Pages 14		

using frame grabbers in PC's. The first hardware system, System A, uses C80 chips and memory on-board a frame grabber for processing and display. The second system, System B, employs simpler frame grabbers and data processing in the PC CPU, Figure 1. The cameras used in the systems are a SBRC midwave/midwave infrared stacked focal plane array read as 256*128 (2 times 128*128), 16 bit output, at 60 frames per second (fps). Only 12 bits of the 16 are significant data. In System B, a visible camera is also used. Its data rate was set at 512*480 pixels with an RS-170, 8-bit output, at 30 fps. In System A, the two midwave images are fused and the resulting 2-color fused image is displayed in real-time. In System B, any two bands, or all three, of two midwave infrared and a visible band, are fused and the 2-color or 3-color fused image is displayed.

2.1. On-board frame grabber processing

The first real-time display system, System A, uses a Pentium 200 MHz running Windows NT, a Matrox Genesis PCI frame-grabber, and the SBRC MW/MW camera (Ref 1). An RS-422 cable, for digital data transfer, connects the camera to the frame-grabber. Any camera with RS-422 output, or for analog data RS-170 output, could be used for this system. In this demonstration, the algorithms were tailored to this SBRC camera. The Matrox Genesis frame-grabber is capable of image processing and has on-board: a C80 processor chip, memory buffers, and a video display module.

For this system, C-code is written in DOS or Visual C and executed on the host CPU. This application calls the Matrox Genesis Native Library routines that execute on the frame-grabber C80 chip. Even though the main process is active in the host CPU, the operations are performed on-board the frame-grabber. The data acquisition and fusion algorithms are not set in hardware; they are memory-resident and very adaptable.

Although there are many available image-processing frame-grabbers with native libraries, there are some specific properties of the Matrox Genesis board, which make it very useable. The C80, a multi-processor DSP, is dedicated to the image processing operations and uses floating point arithmetic. The video display module is capable of 1600*1200 non-interlaced refresh at 85 Hz. This system can read 32-bits of data, and is capable of processing four, 8-bit analog signals, or two, 16-bit signals, or one 32-bit signal. The frame-grabber has on-board AD converters.

An advantage of this system is that there are no host bus issues; the processor and display modules are dedicated to the image-processing task. The VGA interface to an external monitor allows for a fast, large, final display. A disadvantage of this system is that it is limited to 32-bits, so it is harder to add multiple cameras to this system than to the system described in the next section.

For this particular set-up, 256*128*16-bit images, which included both bands, are read from the SBRC MW/MW camera into a memory buffer on the Matrox Genesis frame-grabber at 60 fps. The images are immediately separated in memory into two 128*128, 16-bit images. After processing, the fused image is shown on a 21" color monitor via the external VGA interface of the frame-grabber.

2.2. Only CPU processing

In a second real-time display system, System B, the data streams from the cameras into frame grabbers in the PCI slots of a 400 MHz Dual-Pentium, running Windows NT 4.0. Although many cameras can be used, the cameras used to benchmark the system were a SBRC stacked MW/MW focal plane array 256*128, 16 bit, RS-422, 60 fps, and a visible band 512*480, 8 bit, RS-170, over-sampled at 60 fps. The frame grabbers used are an Imaging Technology IC-PCI motherboard and AM-DIG daughter board, for

RS-422 input, and an IC-PCI motherboard and AM-FA daughter board, for RS-170 input. Many good, comparable COTS frame grabbers are available.

An in-house Win32 application, written in Microsoft C Version 5, reads from the frame grabbers, performs desired processing, opens Windows display windows on the PC monitor, and displays the color-fused images. Up to three windows, displaying different color fusion routines, can be shown side-by-side. The unprocessed, single band data can also be simultaneously displayed in additional windows.

From the dual-band infrared camera, both bands read as one 256*128, 16-bit image. This data is read into a 2*128*128, 16-bit memory buffer on the frame-grabber, which is defined by the programmer, i.e. is not factory set. This buffer is then transferred to the PC RAM. The 512*480 visible camera is actually 8-bits but it is also read into a 16-bit memory buffer on the second frame-grabber. This data is immediately clipped to a 256*256 image which roughly overlaps with the field-of-view of the image from the infrared camera. Only this 256*256, 16-bits of the visible data is transferred to the PC RAM. Immediately the visible data is registered to match the infrared image, described in the next section, reducing it to 128*128, 16-bits. This 3*128*128, 16-bit data buffer is the base from which all of the color fusion processing is done for this system.

The raw data can also be stored to hard disk. While the data is being stored to disk, to achieve maximum rate, the fused image is not displayed. The hard disks used for storage are 2 Seagate Cheetah drives of 9.1 GB connected via an Adaptec 2940UW SCSI board.

3. System Functions of System A

A real-time color fusion display system must display the imagery from the cameras to a viewer in an intuitive method that makes the data easily to understand and analyze. This section and the next will focus on the analytical processes that accomplish these tasks using two different hardware systems. System A uses a smart frame-grabber and on-board processing. System B uses simple frame-grabbers and PC CPU processing.

At the beginning of the data processing in System A, the data from the SBRC dual-band, midwave infrared camera is resident in a memory buffer on the Matrox frame-grabber as two 128*128, 16-bit images. Since the camera is a built around a stacked dual-band focal plane array, a pixel of one band directly corresponds to a pixel in the other band. To "register" the two images, the data has been organized so that matching pixels are in corresponding elements in the two 128*128 data buffers. For cameras with dissimilar fields of view and magnification, a registration process similar to the one used for System B, described below, could be implemented.

3.1. Simple Dual-Band Color Fusion

Human color vision combines three visible bands, from the red, green and blue retinal cones. Displaying imagery from two infrared bands is not a direct correspondence. Since human vision also works on the basis of color opponency, a variant of this concept can be used to create a two-color fused infrared image. An intuitive, linear image can be displayed by presenting the infrared bands as the color opponents, red and cyan. The final display is a 24-bit true color display, made of three 8-bit red, green, and blue buffers. Data from the longer of the two infrared bands is written to the red buffer and the data from the shorter for the two infrared bands is written to both the green and blue buffers, which combine to be cyan.

During processing on the frame-grabber, the data from each band is normalized from 0 to 255, corresponding to the number of shades of colors in the 24-bit color display. The mean of the data is set to

128 and the standard deviation to 64, using a look-up table for maximum speed. In this Simple Color Fusion method, the relative intensities of each two bands can be represented as a chromatic continuum, starting as red, going through gray, and ending as cyan. Each pixel has a chrominant value, red to cyan, and a brightness value, black to white. In the final image, a pixel bright in both bands will be colored white. A pixel bright in only the longer band will be displayed as red. A pixel bright in only the shorter of the midwave bands will be displayed as cyan. Pixels whose values are very different between the two, single bands images are readily apparent as highly colored pixels in the fused image.

This straightforward method of color fusion addresses one of two important issues in processing multi-band color imagery, obtaining good color contrast enhancement between bands. A second issue, obtaining good color constancy regardless of illumination and temperature, is more complex, and is discussed in another paper (Ref 2).

3.2. Principle Components Color Fusion

A second color fusion algorithm improves color contrast enhancement by addressing the fact that the imagery from the two infrared bands is highly correlated. The distribution of pixels tends to be positioned along the darkness-brightness direction. If a new coordinate system is established where the primary axis is along the brightness-darkness direction and the secondary, orthogonal axis is the chrominant direction; the difference between the pixel intensities of the two bands along the chrominant direction can be displayed with maximum color.

The principle component direction is found by first calculating the covariance matrix of all the pixel values, and then finding the eigenvectors of the covariance matrix. The first eigenvector is the principle component direction. The second axis, orthogonal to the first, is the chrominant direction. A rotation matrix can be found which transforms the pixel distribution from the original red-cyan space into the principle component space. For these highly correlated midwave bands, the transformation is essentially a 45-degree rotation. In the principle component space, the data is scaled in the chrominant direction to achieve maximum color. The data is then rotated back to the red-cyan coordinate space to be displayed.

The Principle Components algorithm used in the real-time system simplifies this concept. The principle component direction is assumed to be 45 degrees to the red and cyan axes. It is not calculated in real-time but is pre-set and static. In this simplified approximation, the brightness is the sum of the short and long midwave bands and the chrominance is the difference of the short and long midwave bands, Equation 1.

The data is multiplied by factors, α_1 and α_2 , which essentially compose a rotation matrix. To reduce this method to the Simple Color Fusion method, α_2 is set to zero. The settings used for α_1 and α_2 are a normalization factor and the cosine of 45 degrees. Equation 2 represents the rotation of the data back to the red-cyan coordinate space. The data has undergone two lookup operations, to normalize the data and

$$\begin{aligned} red' &= \alpha_1 * red + \alpha_2 * cyan \\ cyan' &= \alpha_1 * red - \alpha_2 * cyan \end{aligned} \tag{1}$$

$$\begin{aligned} red &= \alpha_1 * red' - \alpha_2 * cyan' \\ cyan &= \alpha_1 * red' + \alpha_2 * cyan' \end{aligned} \tag{2}$$

stretch it in the color direction, and two arithmetic operations, to rotate the data into the principle component space and back to the red-cyan space. Principal Component Fusion is an improvement to Simple Color Fusion and is achieved in this first hardware system with little additional processing.

3.3 Red Enhancement

A third fusion algorithm implemented on this system is called “Red Enhancement”, in which any pixel that has an amount of red intensity above a set threshold will be set completely to the maximum red value.

Any one of these three fusion algorithms can be shown on the 21” monitor in real-time. The purpose of this system was to display processed camera data in real-time. For this system, only the SBRC MW/MW data was processed. It was not a requirement of this system to store data to disk, although the system is capable of performing that task. No measurements of the speed of storage to disk, or playback of stored data from disk, were performed. This system displays the results of one fusion algorithm at a time and does not display scatter plots.

4. System Functions of System B

In System B, the first step in the system is to acquire data from the frame grabbers that read data from the MW/MW dual-band camera and the visible camera. The next step is to register, or “rubber-sheet”, the data, so that the images from all the cameras have the same magnitude, orientation and field of view. Next the data is processed according to any of various color fusion routines. Finally the data is combined into a fused image and displayed, possibly in three windows, each using a different fusion algorithm.

4.1. Registration

In System B, the images from the two cameras have differing field of views, different magnifications and, possibly, different angles of rotation. In previous systems, expensive optics, specific to the camera, are required to match the fields of view. These optics often do not provide the anticipated pixel-to-pixel correlation between images, especially at image edges. The task of this system is to replace the lenses by software, registering the image from the visible camera so that it is warped to match the mid-wave image, Figure 2. The two mid-wave images, from the stacked focal plane array, are already pixel-to-pixel registered with respect to each other. If three separate cameras are used, two camera images can be rubber-sheeted to the third chosen camera image, slowing down the entire process very minimally. The fact that this system can accommodate disparate imagery is a large part of its strength. In this system, to add a new camera, only the calculation of a new rotation matrix needs to be made.

(3)

Registering the image means making an affine transformation (Ref 3-4) in which the image is multiplied by a matrix that includes elements for rotation, translation, and magnification. A calibration matrix is created by adjusting the elements until the image overlaps with the image chosen as the standard,

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix} * \begin{bmatrix} x_{ref} \\ y_{ref} \end{bmatrix} + \begin{bmatrix} a_{00} \\ b_{00} \end{bmatrix}$$

Equation 3. Each (x,y) reference pixel is mapped to a new point. The a,b elements scale and rotate the image, the a₀₀ and b₀₀ elements translate the image.

To implement this matrix multiplication in an algorithm, it is much faster to make a map once than to multiply each pixel by the rotation matrix for every frame. As soon as the new rotation matrix is loaded, the map is created. For this system, which maps the 256*256 visible camera to the 128*128 infrared camera, the x' and y' map matrices are 256*256. The process pulls pixels from the old image to fill the new image. It could happen that two pixels in the reference image map to the same location in the new

image. Since there was no apparent difference between only using one candidate pixel or the average of all candidates, and the processing is faster with one, only one candidate was mapped to the new image. It could also happen that no pixels from the reference image are mapped to a particular pixel in the new image. This pixel would appear blank in the new image. To avoid this, a reference image with a larger magnification than the desired final image was chosen. Instead of mapping the midwave camera to the visible camera, the visible camera was mapped to the midwave camera because its image size is larger than the midwave's. Pixels in the reference image that map to a position outside of the bounds of the desired new image are clipped. The visible image starts as a 256*256, 16-bit image and is shrunk and translated to be a 128*128, 16-bit image.

In this system, the off-diagonal elements, which rotate the image, were not needed and were set to zero. The registration routine only scaled and translated the reference image. The use of off-diagonal elements could be included in the map, which would not slow the process down at all. A new rotation matrix can be loaded at any time without ceasing data acquisition. With this registration routine, real-time fusion of cameras with disparate fields-of-view is achievable.

4.2. Color Fusion Algorithms

4.2.1. Simple Color Fusion

For this 3-color system, the most basic approach to presenting color-fused images from the cameras is similar to that presented for the previous system, which fused two-color data. However, there are three bands of data available, so each band can be made to correspond to a band in human color vision. The final display is a 24-bit true color image, made of three 8-bit red, green, and blue buffers. The longer of the midwave bands is sent to the red buffer, the shorter of the midwave bands is sent to the green buffer, and the visible data is sent to the blue buffer.

First, for each band, a gain and offset is calculated which will normalized the data. In a dialog box, a window in which the user can type, these gain and offset values are suggested. The user can apply them or type in new values. A mean of 128 and a standard deviation of 64 work well for Gaussian distributions. For bi-modal distributions, such as an image of a dark plane and a bright sky, the suggested values would cause the largest and smallest pixel values to be forced to the edge of the distribution, and a handset gain might be a better choice. This algorithm uses integer math, making it faster by a factor of two than if it used floating point math.

The normalized data is sent to a window on the PC monitor via a Windows display function to present the 3-color real-time fused image.

4.2.2. Principle Components Fusion

The Principle Component fusion method is also implemented in this system. The first eigenvector of the covariance matrix defines the principle component direction, which tends to lie along the brightness-darkness line of the pixel distribution. Two vectors, orthogonal to the principle component direction, define a chrominant plane, as oppose to the chrominant line in the two-color system. Final colors possible in the display are red, green, blue, and any combination of these three, including all shades of gray from black to white.

In this algorithm, the raw data is normalized frame by frame in real-time. No pixel values are clipped at this stage, since the data will be rotated into the principle component space. A dialog box allows the user to set the angles used for the rotation. Eigenvectors are not calculated. Instead, they are pre-set. Since the two mid-wave bands are correlated, the initial suggested rotation angles are a theta of 45 degrees for the first rotation and a phi of 0 or 54 degrees for the second rotation, depending on the degree of anti-correlation of the visible and infrared bands. The rotations back to the red-green-blue plane are always a phi of 45 degrees and a theta of 54 degrees.

In the principle component space, the mean in the brightness–darkness direction is set to be 221, half of the magnitude of a vector that would have red, green and blue components equal to 255. In this way, when the data is rotated back to the red-green-blue space, the maximum color value, 255, of each band can be displayed. The dialog box also allows the user to enter gain values that stretch the data in the chrominant plane, or shift the data in the brightness–darkness direction. In the final stage, the pixel values are clipped to a minimum of 0 and a maximum of 255, however the data was expanded or condensed in the principle component space so that clipping is not often necessary.

The 3-band pixel distribution is displayed as a 3-color fused image that depicts contrasts between bands as pixel color.

4.2.3. Monochrome Fusion

The three images can be fused into one black and white image (Ref 5). To display the monochrome fused image in real-time, the simple color algorithm is followed to normalize the data. Then, in the final stage, the data from all three bands for each pixel is averaged and that value is sent to each of the red, green and blue data buffers. If a pixel is very bright in the longest band and dark in the others, the final pixel has a gray value, as opposed to the color fusion system, in which the pixel would have a large, and readily apparent, red value. In monochrome fusion, the information can be averaged and lost.

4.2.4. Red Enhancement Fusion

To accentuate even the slightest signal in the longest wavelength, in the Red Enhancement fusion, any pixel with a value above a threshold is set to the maximum pixel value. This fusion method pegs the pixel at the maximum red color, if there is any red in the pixel at all. Any pixel with a below threshold value will be a simple color fusion of the three bands. The threshold value used for demonstration is arbitrarily set to 60.

4.2.5. Gamma Stretching Fusion

Gamma stretching refers to the method of applying a non-linear scale function to at least one of the bands. For example, to represent a wider range of values in the longest wave of 3 bands, the data is stretched,

$$Red' = 128 * \left(\frac{Red}{128} \right)^\gamma$$

The output is now less sensitive to variations in pixel value for dim objects and more sensitive to variations in pixel value of bright objects, which would have saturated in the previous fusion modes. For this particular formulation, no pixels with a value below the mean of 128 are diminished. Pixels with a value greater than 128 quickly approach the maximum value of 255. A value of gamma equal to 3 was

used for demonstration. For gamma equal to one, this method reduced to the Simple Color Fusion method.

4.3. System B Performance

This hardware system is useful for at least four different functions. The first is to simply display fused imagery from two or three cameras, which are usually non-registered optically, in real-time. It is also important to store the data and to be able to replay stored data in real-time. The second function of the system is to allow side-by-side comparison of two fusion algorithms. A third function is to display two different combinations of cameras in real-time so that comparisons can be made between choices of band selections. The final function is to display the scatter plots of pixel intensities in real-time so that the difference of a target's pixels compared to other objects in the image can be quantitatively represented.

4.3.1. Rate of Display and Storage

The rates achieved for displaying the data as fused imagery from the cameras, storing the unprocessed data to hard disk, and displaying the unprocessed data from hard disk as fused imagery are tabulated in Table 1 for four different fusion algorithms, which were described in the previous section. It is assumed that one display window is being shown at a time and no other applications are running on the processing system. Generally, real-time display means anything over 30 fps, the limit of a human's visual ability to discern changes in motion.

The cameras pixel size, bit size, and frame rate are: MW (256 pixels * 128 pixels * 2 bytes per pixel * 60 fps), visible (512 pixels * 480 pixels * 1 byte * 30 fps). Table 1 shows the frames per second at which the data buffer in the algorithm is updated. The MW camera can only run at 60 fps. The visible camera can only run at 30 fps. However, the algorithm can over-sample the frame grabber. The limiting factors to the actual display rate are camera frame rate and monitor display rate, although both are faster than a person's vision. The processing does not lower the frame rate below the 30 fps limit.

Since the data is clipped and decreased to 24-bit color data (3 colors of 8 bits each) as it is processed, frames per second is a more meaningful measure than megabytes per second when discussing the display rate of the system. However, when the data is stored to disk, no processing is done. The bits read are the bits written. For the storage column in the table, megabytes per second is meaningful, so it is listed. The rate for storage of raw data to hard disk is 13 MBps.

	Display rate from camera	Storage rate to hard disk	Display Rate From hard disk
Simple Color Fusion	270 fps	68 fps (13 MBps)	45 fps
Principle Components Color Fusion	87 fps	68 fps (13 MBps)	34 fps
Red Enhancement	247 fps	68 fps (13 MBps)	45 fps
Gamma Stretching	83 fps	68 fps (13 MBps)	32 fps

All of these algorithms can display, store, and display stored data in at least 30 fps.

4.3.2. Comparison of Algorithms

In the CPU-only processing system, the four algorithms can be compared side-by-side. Instead of sorting through Gigabytes of collected data, one can identify interesting phenomenology while in the field. Figure 3 represents a display of three fusion algorithms side-by-side in real-time. The three algorithms represented are Simple Color Fusion, Red Enhancement, and Principle Component Color Fusion. In the image, a power plant is shown and the CO₂ emissions of the exhaust plumes are very apparent. There is glint on the lens apparent in the shorter of the two midwave bands, the green band. In the Red Enhancement image, pixels in the face of the power plant, which were slightly red, are presented as very red. In the Principle Component image, the difference between the color of the sky, power plant, glint, and water in front of the plant is accentuated.

4.3.3. Comparison of Band Combination

Band selection is a pertinent issue for fused camera systems – which of the available bands should one choose considering intended applications, target emission properties, atmospheric conditions, and available light? While this can be speculated, and the manufacturer documents basic sensor performance, the performance of a system in the field is not obvious. These real-time hardware systems are an adaptable, inexpensive test tool to carry to the field to perform the question of band combination success. Figure 4 represents the same image fused using two different band combinations.

4.3.4. Visualization of Scatter Plots

It is important, while trying to develop a visual representation, to know how the target pixels compare to background or other objects in a quantitative way. Differences in target and background pixel values can be exploited only if they are identified. Seeing the scatter plots also helps the user set values such as gain, offset, and the optimum angles of rotation into the principle component space. Scatter plots of still imagery have been used widely but are even more powerful in a real-time system in which the scatter plots change as the scene changes. An example of using scatter plots to maximize differences in target and background pixels is given using Figures 5 and 6. Figure 5 is a 3-color fused image made with the Principle Components Fusion algorithm. In the image, a person is holding a piece of plastic that transmits in the shorter of the mid-wave bands and not in the longer of the mid-wave bands, so the plastic appears very green. The top row of scatter plots in Figure 6 are taken in the RGB coordinate space. Each plot shows the pixel values of two of the three bands versus each other. In the last scatter plot, which shows the longer mid-wave, red, versus the shorter mid-wave, green, the plastic pixels can be seen above the main distribution, toward the positive green axis. The plastic pixels are also apparent in the second row of scatter plots, which are taken in the principle coordinate space. In the first and last scatter plots in the second row, the plastic pixels are seen on the right hand side of the distribution. This is the coordinate frame in which the data is to be normalized along the red-green and blue-yellow axes. After the data is normalized, it is rotated back to the RGB space, shown in the scatter plots of the last row. Now, in the last scatter plot that shows the red versus green bands, the plastic pixels are in the upper left-hand corner, well separated from the main distribution. The scatter plots allow immediate feedback to changes in the normalization and the angles used to rotate into the principle coordinate space.

5. Summary

Two hardware configurations for real-time display of a few cameras have been presented. The systems can display fused images in real-time. One system processed the data on the frame-grabber. The other system processed data on the PC CPU. For the second system, real-time storage to hard disk was demonstrated. For this system scatter plots of the pixel distributions can be viewed in real-time. A strength of the systems is that they are able to fuse imagery from cameras without matching optics. This is a great money and time saver. The systems are inexpensive and adaptable. This tool will greatly aid in investigating the questions, “which band combinations should be used for this application” and “which algorithms perform this fusion best for this scenario”.

6. References

1. James R. Waterman, Dean Scribner, “Real-time Fused Color Imagery from Two-color Midwave HgCdTd IRFPAs”, 1998 Meeting of the IRIS Specialty Group on Materials and Detectors, Volume I, August 1998.
2. Dean Scribner et al, “Infrared Color Vision: Separating Objects from Backgrounds”, April 1998 SPIE Conference 3379
3. Robert A. Schowengerdt, Remote Sensing, p333.
4. P. Warren, D. Scribner, J. Schuler, M. Kruer, “Multi-band Color Fusion”, 1998 Meeting of the IRIS Specialty Group on Passive Sensors, Volume I (unclassified), March 1998
5. T. Peli, 1996 Meeting of the Iris Specialty Group on Passive Sensors, Volume II (classified), March 1997, p.13.

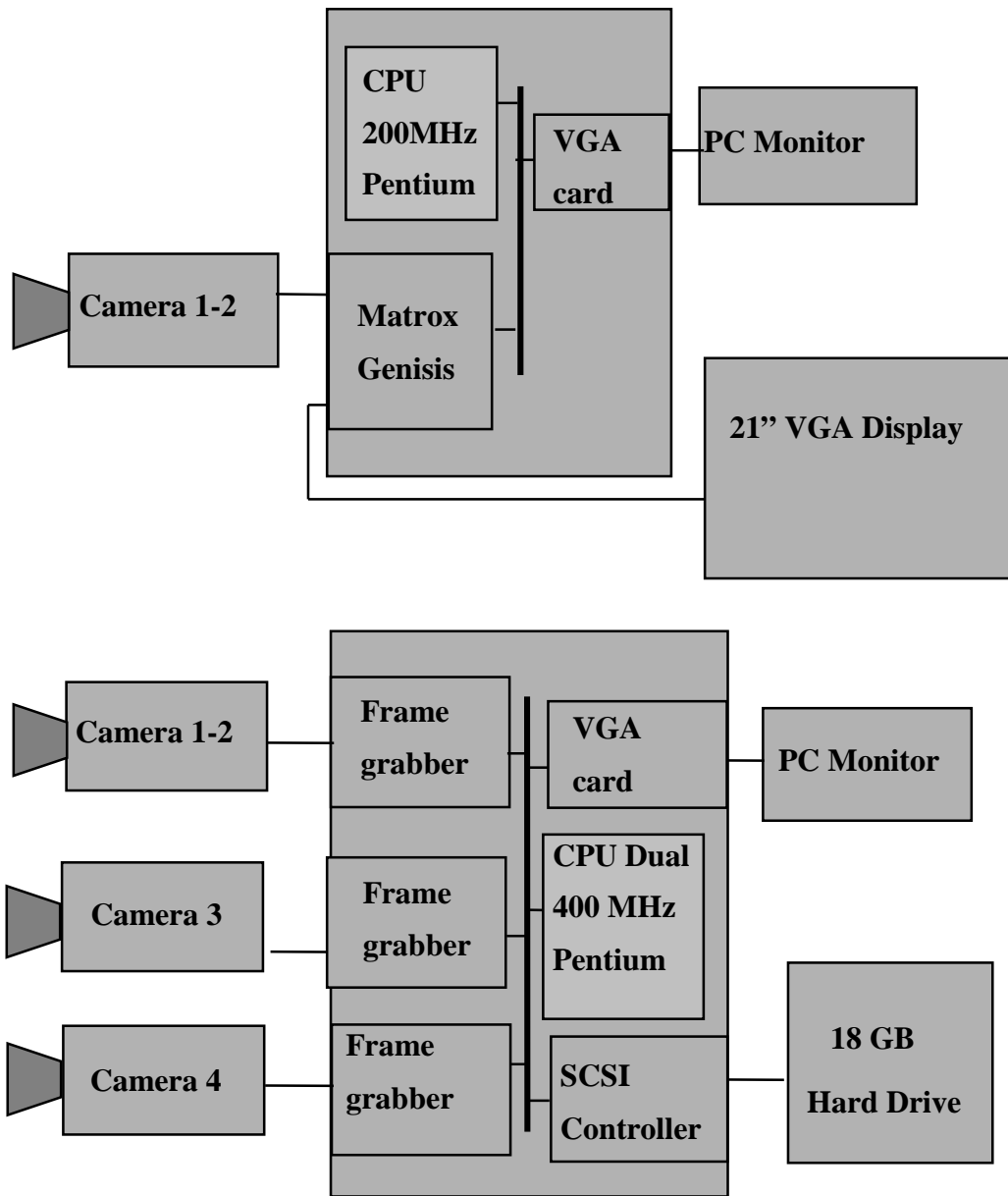


Figure 1. Diagram of Hardware Systems. The top system, System A, reads data via a RS-422 or RS-170 cable into a Matrox Genesis frame-grabber with on-board image processing ability, memory, and connection to an external VGA monitor. Although the fusion application is active in the PC CPU, it calls library routines that are processed on the frame-grabber C80 chips. The bottom system, System B, uses multiple Imaging Technology frame-grabbers. The IC-PCI motherboards are combined with AM-FA daughter boards to read RS-170, or AM-Dig daughter boards to read RS-422 images. The processing for this system is all done in the PC CPU and displayed on the PC monitor. The raw data can also be stored to a hard drive connected via an Adaptec 2940UW SCSI controller and later played back from the hard drive to be processed and displayed on the monitor.

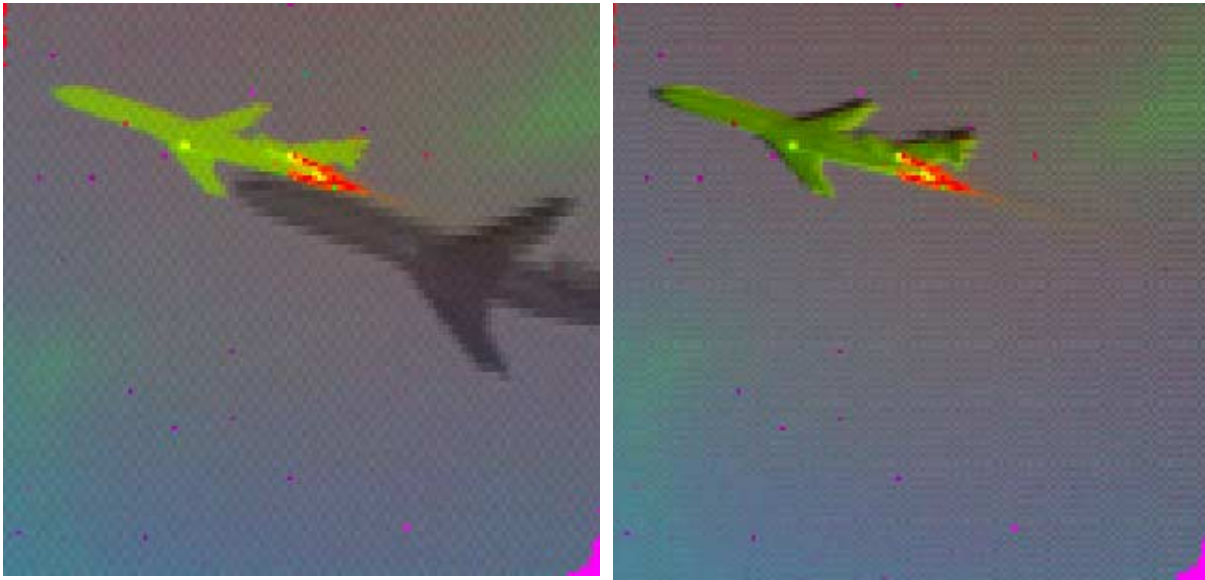


Figure 2. The system’s ability to register two disparate images is shown. The left image shows combined images from a dual band midwave infrared camera and a visible camera before registration. The right image is after registration of the visible image to the midwave image. The visible image has been scaled and translated to fit to the midwave image.



Figure 3. The system can compare three fusion algorithms simultaneously in real-time. The left image is created using Simple Color Fusion, the middle image with Red Enhancement, and the right image with Principle Component Fusion. There is glint on the lens apparent in the shorter of the two midwave bands, the green band. In the Red Enhancement, image pixels in the face of the power plant that were slightly red are presented with maximum red value. In the Principle Component image, the difference between the color of the sky, power plant, glint, and water in front of the plant is accentuated.



Figure 4. The system allows combination of bands to be compared. The left image is a two-color fused image of the two midwave bands, red and green. The data in the two bands is very similar and the person appears as a combination of red and green, yellow. The right image is a 2-color fusion of the shorter of the two midwave bands and the visible bands, green and blue. Note the visible band has information about the background. The person's shirt is more reflective in the visible and his skin emits in the infrared band. These two color fusion images were made by disabling the blue or red band in the 3-color Simple Fusion method and not with the 2-color color-opponency method.



Figure 5. In this 3-color image created with the Principle Component algorithm, the man is holding a piece of plastic that transmits in the shorter of the two mid-wave bands. The plastic is an obvious green color. This image is used for the next figure, which shows the scatter plots of the intensity of the pixels.

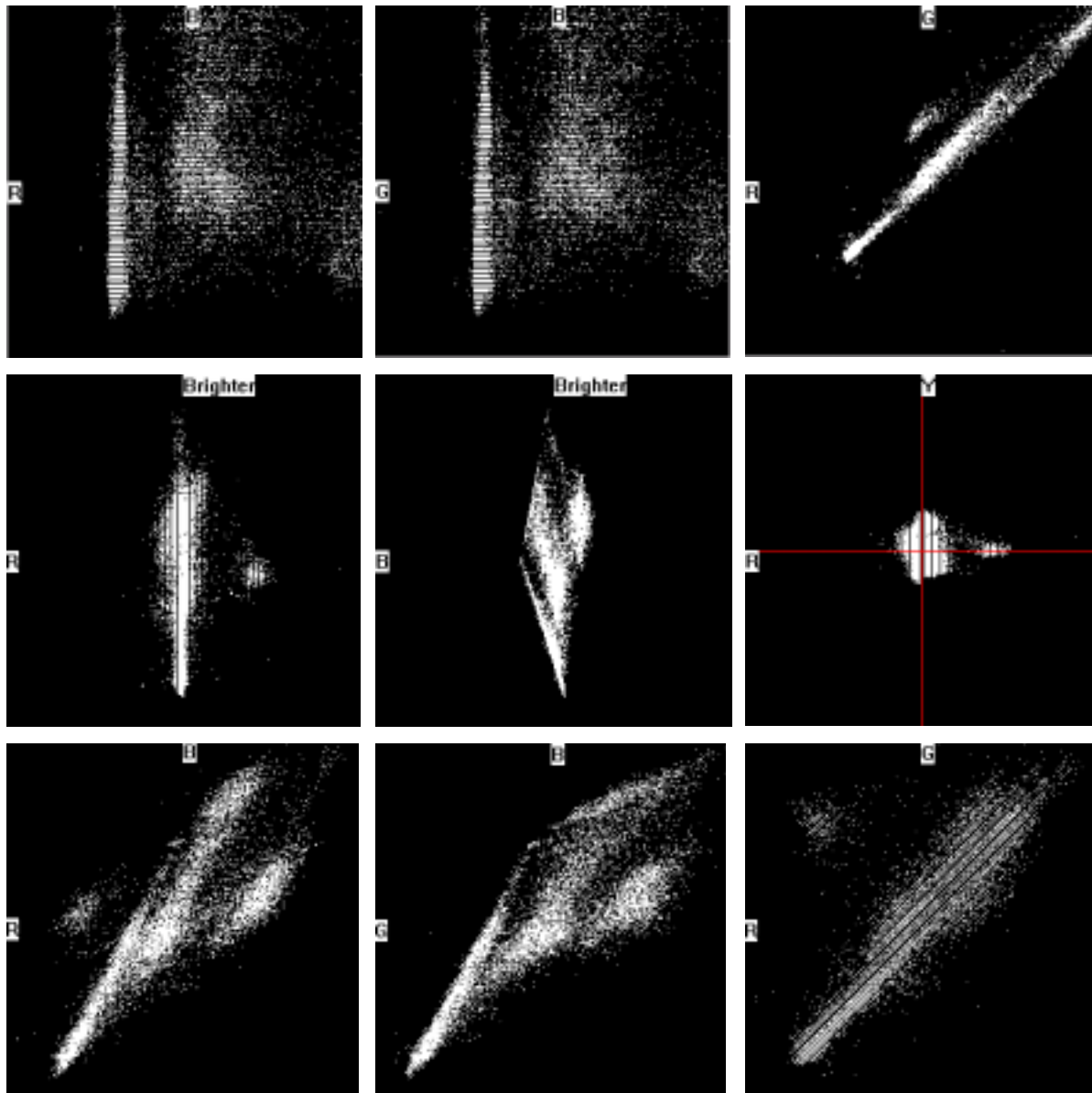


Figure 6. This set of scatter plots is associated with the 3-color fused image in the previous figure that was made with the Principle Components algorithm. The top row shows scatter plots of the intensity of the pixels of one camera versus another taken in the initial RGB coordinate space. The second row is of the principle component coordinate space and the bottom row is again in the RGB coordinate space, after normalization in the principle component space. All scatter plots can be displayed in real-time. In the RGB space, the longer of the infrared bands is labeled R for red, G for the shorter midwave band, and B for visible. In the principle component space, the axes are dark-bright, red-green and blue-yellow. The first two plots in the top row show that the visible (blue) data is virtually uncorrelated with the infrared cameras. In the top right plot, it is apparent that the infrared cameras are very correlated. In this plot, the set of pixels above the main distribution are from the plastic filter held up to the man's face. The left plot in the second row is the red-green direction versus the brightness direction in the principle component space; as if the yellow-blue direction is out of the page. The right plot is the yellow-blue versus the red-green axes – this is the chromaticity plane. In this space, the pixels from the plastic filter are the greenest pixels in the image. The lower set of plots are the distributions after they have been normalized in the principal component space and rotated back to the RGB space. The filter pixels have moved from a “gray” position in the middle of the plot, to a more colored position, at the edge of the plot. Looking at these scatter plots in real-time can help to identify which algorithms separate the target pixels from the background pixels.