SANDIA REPORT
SAND2007-5869
Unlimited Release
Printed September 2007

Fusion of Image Data for Beyond the Fence Intruder Detection and Assessment

Cynthia L. Nelson

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Approved for public release; further dissemination unlimited.

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd.
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2007-5869
Unlimited Release
Printed September 2007

Fusion of Image Data for Beyond-the-Fence Intruder Detection and Assessment

Cynthia L. Nelson
Security Systems and Technology Center
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-0780

Abstract

The use of combined imagery from different imaging sensors has the potential to provide significant performance improvements over the use of a single image sensor for beyond-the-fence detection and assessment of intruders. Sensing beyond the fence is very challenging for imagers due to uncertain dynamic and harsh environmental conditions. The use of imagery from varying spectral bands can alleviate some of this difficulty by providing stronger truth data that can be combined with truth data from other spectral bands to increase detection capabilities. Imagery fusion of collocated, aligned sensors[1] covering varying spectral bands [1,2,3] has already been shown to improve the probability of detection and the reduction of nuisance alarms. The development of new multi-spectral sensing algorithms that incorporate sensors that are not collocated will enable automated sensor-based detection, assessment, localization, and tracking in harsh dynamic environments. This level of image fusion will provide the capability of creating spatial information about the intruders. In turn, the fidelity of sensed activities is increased resulting in opportunities for greater system intelligence for inferring and interpreting these activities and formulating automated responses. The goal of this work is to develop algorithms that will enable the fusion of multi-spectral data for improved detection of intruders and the creation of spatial information that can be further used in assessment decisions.

[1] Collocated, aligned sensors refers to those that are placed in the same, or close to the same, physical location and that are aimed in directions that will maximize the overlap in their fields-of-view.

Fusion of Image Data for Beyond the Fence Intruder Detection and Assessment

1. Introduction

Image fusion is an important step to improve imaging and automatic detection and classification performance over that of single imaging sensors. Registered images from different times (multi-temporal) and different sensors (multi-spectral and multi-resolution) can be combined to produce composite imagery with spectral and spatial characteristics equal to or better than that of the individual imagers. When the sensors are collocated and aligned, emphasis is placed on registration of the images. This amounts to spatial alignment of overlapping images so they can be mapped to a common two-dimensional coordinate system. Once registration is achieved, the images can be combined into a composite image for further processing. One approach to this is discussed in Section 2.

When the images are not collocated, but are viewing a common scene, a mapping is still required, but the registration becomes a problem of mapping 2-D information into a common 3-D coordinate space. There are generally two approaches to this fusion process. In the first approach, objects of interest are extracted from a 2D image and combined with those extracted from other 2D images. The result is a set of targets represented as point objects in a three-dimensional space model. These point objects are then used to locate, ID, and track targets in space and time. Common examples of point objects represented in a 3-D space model are aircraft in airspace or targets on a battlefield. This approach to fusion is considered the general fusion of data rather than the fusion of imagery. Discussion of this approach to fusion can be found in [4]. The second fusion approach is aimed more at image fusion in which *complete* 2-D data sets from multiple source imagery are acquired and combined. A complete spatial mapping from the multiple sources is then created, and the composite mapping is processed to extract useful assessment information. Steps needed for this fusion process are discussed in Section 3.

2. Fusion of Multi-Spectral Images from Collocated, Aligned Sensors

An image scene viewed through different spectral bands has varying characteristics depending on the spectrum being observed. For example, the images in Figure 1 show the visible band at the low (blue) and high (red) end of the spectrum, respectively. There are noticeable changes that can be easily seen such as the brightness of the rectangular sign in the lower left of each image. Other differences are not quite as noticeable by simple observation, but can be extracted through various filtering techniques. For example, the texture of the mountains approaches a maximum towards the “green” portion of the spectrum and is reduced at both the red and blue ends. The goal in fusing

images from the different spectrums is to draw out the distinguishing features from each spectrum and create a composite image, which can then be more easily assessed.

Figure 1. An image scene observed in the low and high extremes of the visible spectrum.

The first step in combining different images from aligned sensors is to register all images to a common coordinate system. This spatial alignment is a prerequisite for further operations and can occur at the raw image level, in which each pixel in an image is referenced with known accuracy to either a pixel or pixels in another image or to a selected coordinate mapping. This will allow for the creation of a composite image from multiple light spectrums. For the general data fusion approach, the registration can also occur at higher levels, relating objects rather than individual pixels.

Composite images can be created by applying a high-pass filter over each of the input images obtained from different narrow spectral bands. Pixel values for the composite image are selected from the band with the greatest high frequency response at each particular pixel location. With this approach, the goal is to create composite images with maximum textures. The composite image, useful for assessment purposes, leverages the complementary strengths from the different spectral bands. The pictures in Figure 2 illustrate the process using data obtained from just 2 bands (blue and red). Data collected in the "blue" portion of the spectrum is more focused in the near field (upper left thumbnail image) while data collected in the "red" portion is more focused in the far field (upper right thumbnail image). The result of the high-pass filtering operation is seen in the pair of thumbnail images on the lower left. The composite image (far right image) is in sharp focus in both the near field and far field, and maximizes the texture throughout the entire image. By maximizing the texture, more useful information can be extracted from the scene that can then enhance the intelligence needed for assessment purposes.

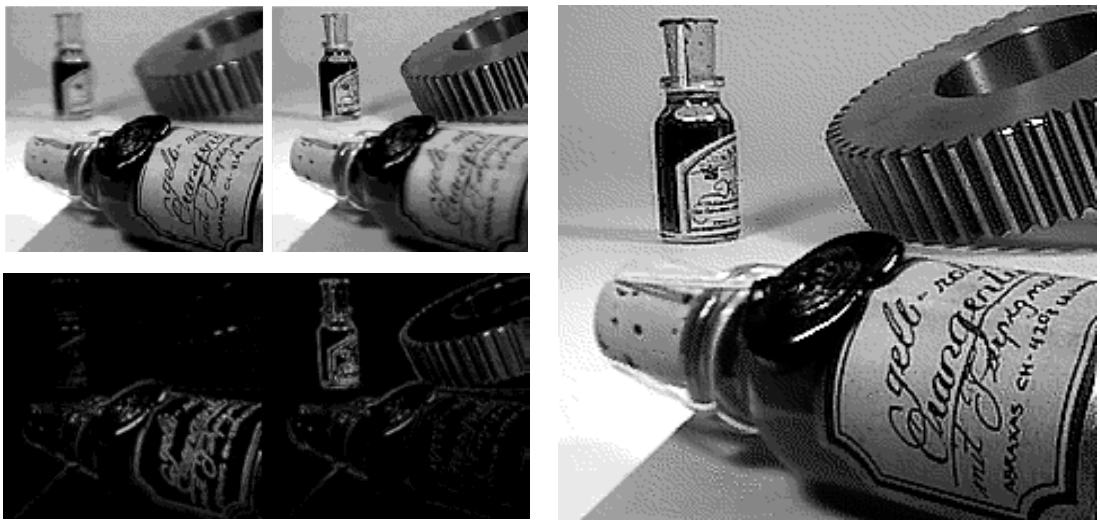

Figure 2. Processing images from the blue and red ends of the visible spectrum is performed to create an enhanced composite image.

3. Fusion of Images from Non-Collocated Sensors that Observe a Common Region of Interest

It is not always feasible or practical to collocate image sensors that observe a similar region of interest. For applications such as beyond-the-fence monitoring, the environment or terrain may limit the placement of individual sensors or imagers from different spectrums may operate optimally from different locations. The image data can still be fused by properly mapping it into a common time, space, and spectral reference frame. This process is often referred to as data alignment. Data alignment involves spatial transformations to project (or warp) image data to a common three-dimensional coordinate system or earth reference model. At this point, non-imaging data that can be spatially referenced (e.g., acoustic, seismic) can also be associated with the image data. This leads into the capability to do data association, which is very important in order to correlate new data with previous data for detection and segmentation of targets on the basis of temporal changes (motion) or spatial changes (behavior).

There are two major areas that need to be addressed to perform the mapping of two-dimensional image data to a common reference model. Section 3.1 describes the development of the camera model that can be used to perform the mapping. A pin-hole camera model is assumed for simplicity (i.e., no distortion). If there is significant distortion in the image, it must be characterized and corrected in order to use the pinhole model. Then, once the camera model is defined, two-dimensional images can be mapped to a common three-dimensional reference model. The 2-D to 3-D mapping is discussed in Section 3.2.

3.1. Defining a camera model to map 2-D images onto a common 3-D coordinate frame

This section describes the development of a camera model that can be used to transform world points to image coordinates. Once this model is developed, an inverse mapping is used to go from the pixel coordinates to world coordinates. Development of the camera model is broken down into three major steps: transforming world coordinates to camera coordinates, mapping camera coordinates to sensor coordinates, and mapping sensor coordinates to pixel coordinates.

Mapping World Coordinates to Camera Coordinates

The mapping of world coordinates to camera coordinates is performed by the use of a coordinate transform.

Let's assume

$\mathbf{P_j} = [x_j, y_j, z_j]$ = world position of camera j
$\mathbf{P_{wij}} = [x_w, y_w, z_w]$ = world position of a point i in camera j.
$\mathbf{P_c} = [x_c, y_c, z_c]$ = camera coordinates of the point i.

To go from world to camera coordinates, the following mapping must be solved:

$$\mathbf{P_c} = \mathbf{R}[\mathbf{P_{wij}} - \mathbf{P_j}],$$

where **R** is a 3X3 matrix representing the angles of rotation of the camera. The matrix **R** can be defined as

$$\mathbf{R} = \mathbf{R_x}\mathbf{R_y}\mathbf{R_z},$$

where

$$\mathbf{R_j} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \quad \text{for j} \in \{x, y, z\},$$

and θ is the angle of rotation about the x, y, or z axis. The equation for $\mathbf{P_c}$ can be expanded to

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} x_w - x_j \\ y_w - y_j \\ z_w - z_j \end{bmatrix},$$

and rewritten as

$$\begin{aligned} x_c &= r_{11}(x_w - x_j) + r_{12}(y_w - y_j) + r_{13}(z_w - z_j), \\ y_c &= r_{21}(x_w - x_j) + r_{22}(y_w - y_j) + r_{23}(z_w - z_j), \\ z_c &= r_{31}(x_w - x_j) + r_{32}(y_w - y_j) + r_{33}(z_w - z_j). \end{aligned} \tag{1}$$

The camera coordinates are a function of a rotation of the difference between the world coordinate of a point and the world coordinate of the camera.

Camera Coordinates to Sensor Coordinates

A mapping from camera coordinates to sensor coordinates can be established by using a perspective transform. This can be described using the following diagram.

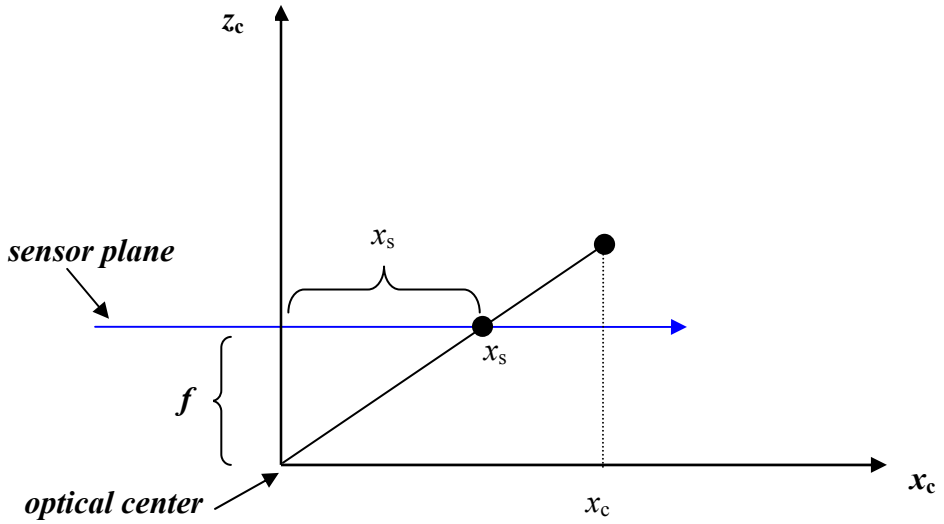

Figure 3. Relationship between the camera and sensor coordinates.

In this figure, z_c extends out from the front of the camera and x_c extends across the image plane of the camera. Although it is not shown, y_c extends out the top of the camera. The view is down from the top of the camera, down the y axis. The focal length is represented by f and x_s is a pixel on the focal plane. The relationships in the above diagram are as follows.

$$\frac{x_s}{f} = \frac{x_c}{z_c} \Rightarrow x_s = f * \frac{x_c}{z_c}$$

and (2)

$$\frac{y_s}{f} = \frac{y_c}{z_c} \Rightarrow y_s = f * \frac{y_c}{z_c}$$

Sensor Coordinates to Pixel Coordinates

A relationship is now defined to go from sensor coordinates to pixel coordinates by applying a scaling transform. The sensor format of the imager is required to complete this step. For example, a 1/3" format imager has a sensor height of 3.6 mm and a sensor width of 4.8 mm. The following terminology is used to define the necessary equations:

N_v = number of pixels vertically (e.g., 480 for a 640x480 image)
N_h = number of pixels horizontally (e.g., 640 for a 640x480 image)
h = sensor height (e.g., 3.6 mm for a 1/3" imager format)
w = sensor width (e.g., 4.8 mm for a 1/3" imager format)
(x_p, y_p) = pixel location on the image.

Using the equations from (2), we can define the relationships between the pixels and the sensor format as

$$x_p = \frac{N_h}{w} * x_s = \frac{N_h}{w} * f * \frac{x_c}{z_c} = \frac{N_h f}{w} * \frac{x_c}{z_c} = S_h * \frac{x_c}{z_c}$$

and

$$y_p = \frac{N_v}{h} * x_s = \frac{N_v}{h} * f * \frac{y_c}{z_c} = \frac{N_v f}{h} * \frac{y_c}{z_c} = S_v * \frac{y_c}{z_c}$$

where,

S_h is the ratio of horizontal pixels to sensor height and
S_v is the ratio of vertical pixels to sensor width.

If the pixels are square, we can assume $S_h = S_v$ and the resulting relationship is

$$S = \frac{N_h f}{w} = \frac{N_v f}{h}.$$

This is called the scaled focal length and can be substituted into the previous equations to get

$$x_p = S * \frac{x_c}{z_c}$$

and (3)

$$y_p = S * \frac{y_c}{z_c}$$

We can now substitute equation (1) into equation (3) to describe the mapping of world coordinates to pixels coordinates.

$$x_p = S * \frac{r_{11}(x_w - x_j) + r_{12}(y_w - y_j) + r_{13}(z_w - z_j)}{r_{31}(x_w - x_j) + r_{32}(y_w - y_j) + r_{33}(z_w - z_j)}$$

and (4)

$$y_p = S * \frac{r_{21}(x_w - x_j) + r_{22}(y_w - y_j) + r_{23}(z_w - z_j)}{r_{31}(x_w - x_j) + r_{32}(y_w - y_j) + r_{33}(z_w - z_j)}$$

These pixel positions assume a pinhole camera model in which there is no distortion involved and the world point, image point, and optical center are collinear. Under this assumption, in which there is no distortion in the images, an inverse mapping can be used to go from pixel coordinates to world coordinates. This can be shown as follows:

$$\begin{bmatrix} x_w - x_j \\ y_w - y_j \\ z_w - z_j \end{bmatrix} = R^{-1} \begin{bmatrix} (x_p \cdot z_c)/S \\ (y_p \cdot z_c)/S \\ z_c \end{bmatrix}. \quad (5)$$

This relationship provides a method to map unregistered two-dimensional images onto a common world coordinate frame. An example is shown in Figure 4. The top two images are of the same scene taken from different viewpoints (imagers are not collocated). The bottom two images illustrate the mapping to the common world coordinate frame in which the images are now registered. For simplicity and illustration purposes, the images in this example are mapped to a plane ($z_w = 0$).

Registration is expected to allow better tracking and assessment capabilities in an extended detection system. Detection and tracking will be greatly improved by mapping data from all spectrums (collocated or not) onto the common world coordinate grid. Thus, a better layout of activity will be seen by the tracking algorithms.

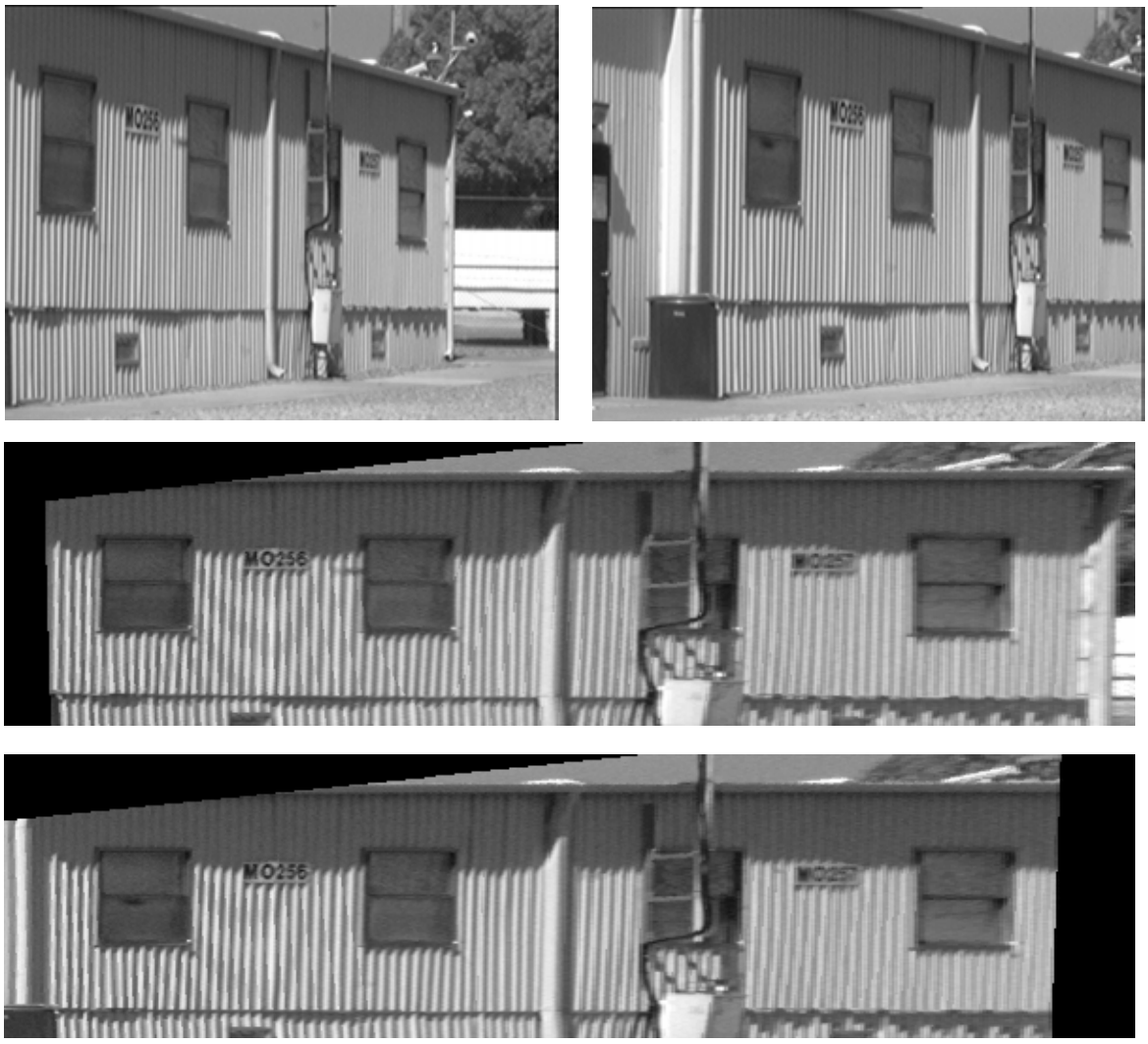

Figure 4. Result of mapping unregistered 2D images to a common coordinate space.

World Coordinates to Pixel Coordinates and the Need for Distortion Correction

It is important to note that x_p and y_p from the previous section represent undistorted pixels. Therefore, the mapping in Figure 4 assumed distortion correction was not needed. Generally, in real-world applications, this assumption does not hold and the most important deviation is in terms of radial barrel distortion. In order to correct for this distortion error, the image measurements must be mapped to those that would have been obtained under a perfect linear camera. Lens distortion takes place during the initial projection of the world onto the image plane. A distortion function is applied to the measured coordinates to yield corrected coordinates that will obey a linear projection. In defining a distortion model, it is be useful to refer to the diagram below.

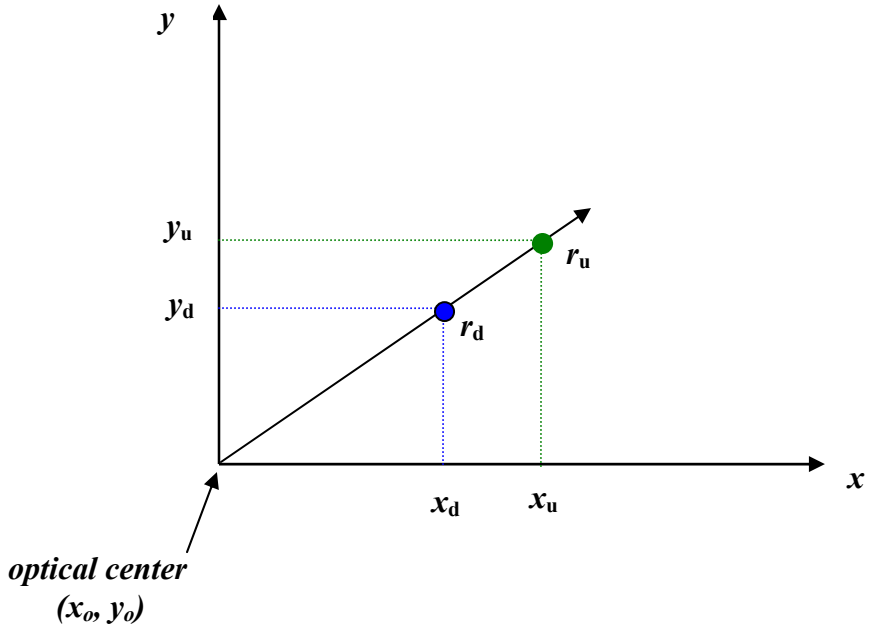

Figure 5. Pixel relationships with radial distortion.

In this diagram, (x_u,y_u) is the undistorted pixel position and (x_d,y_d) is the distorted (measured) pixel position. The center pixel position on the optical plane is represented by (x_o,y_o). The distance from the center pixel to the pixel position of a point as seen in the distorted (observed) image is r_d. The variable r_u is the distance from the center pixel to the pixel position of a point as would be seen in the undistorted image. From the figure, the following relationships can be defined:

$$\cos\theta = \frac{x_d - x_o}{r_d} = \frac{x_u - x_o}{r_u}$$
$$\sin\theta = \frac{y_d - y_o}{r_d} = \frac{y_u - y_o}{r_u} \quad , \qquad (5)$$

and, from Pythgorean's theorem,

$$r_u^2 = (x_u - x_o)^2 + (y_u - y_o)^2$$
$$r_d^2 = (x_d - x_o)^2 + (y_d - y_o)^2 \quad .$$

A distortion factor, which is a function of the radius only, must be defined. The distortion factor is used in a distortion model to change an undistorted pixel to a distorted pixel (or vise versa). That is,

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = \begin{bmatrix} x_o \\ y_o \end{bmatrix} + F(r_u) \begin{bmatrix} x_u - x_o \\ y_u - y_o \end{bmatrix}$$

Using this correction, the coordinates (x_u, y_u) have a linear relationship to the coordinates of the world point. They represent the ideal image position if there was no distortion. An approximation to an arbitrary distortion factor is given by a Taylor series expansion

$$F(r) = 1 + k_1 r + k_2 r^2 + k_3 r^3 + k_4 r^4 + \ldots$$

The function $F(r)$ is defined for positive values of r and $F(0) = 1$. The coefficients for the radial correction are $\{k_1, k_2, k_3, k_4, \ldots, x_o, y_o\}$ and are considered to be a part of the intrinsic calibration of the camera. This correction is needed to map from an image point to a ray in the camera coordinate system. An example optical model that relates the undistorted radius to the distorted radius is

$$r_u = r_d(1 + k \cdot r_d)^2 .$$

The parameter k is adjusted to correct for the observed distortion. If the correct value for k equals 0, there is no distortion in the observed image. Larger values of k indicate more distortion. Applying the equations in (5), the relationships between distorted and undistorted coordinates can be written as

$$x_u = x_o + (x_d - x_o)(1 + k \cdot r_d)^2$$
$$y_u = y_o + (y_d - y_o)(1 + k \cdot r_d)^2 \quad .$$

An example using this procedure for radial distortion is shown in Figure 6. The image on the left was taken through an ultra-wide angle lens. It is easy to notice the severe amount of distortion by the curvature of the horizon and fence line. The image on the right is the corrected, undistorted image.

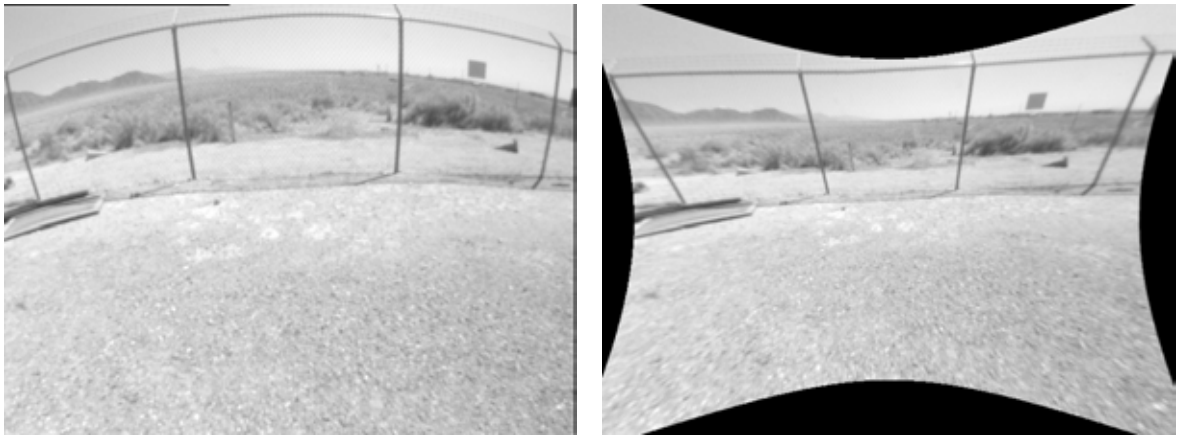

Figure 6. An image before and after radial distortion is removed.

Once distortion has been removed from the image, the inverse mapping shown in equation (5) can be applied. This will provide the world coordinates of points in the image. A further discussion of a process to determine the distortion correction coefficients for a given camera is provided in Appendix A.

3.2. Mapping multiple camera views to a common coordinate system

Undistorted two-dimensional images can be mapped onto a common coordinate system. To accomplish this, knowledge of the Euclidean structure of the scene must be available. If this information is not available, a method of self-registration can be used in which the relative positions and orientations of cameras can be determined. The goal is to determine these parameters without ever having to measure the world positions of calibration targets. This is a significant benefit for deploying systems in uncontrolled or hazardous territories.

Self registration determines the relative three-dimensional positions and orientations of cameras in a multi-camera network by only observing distinct points in the common field of view. It will also determine the relative three-dimensional positions of the points. The relative positions of the cameras and points can only be determined within a scale factor, but the relative orientations are precisely determined. This process facilitates data association for fusion algorithms.

Self-registration requires at least two different views of an area, either from two separate cameras or from one camera at two different positions. Matching of distinct features or points from multiple camera views is required. This can be a difficult problem and different techniques can be exploited to simplify the correspondence. The approach of interest for this work is a problem in optimization. It is defined in the equations below.

Define the world space as

$$\mathbf{Pc}_{ij} = \mathbf{R}_j (\mathbf{Pw}_i - \mathbf{P}_j),$$

where

$\mathbf{Pc_{ij}} = \mathbf{[Xc_{ij}, Yc_{ij}, Zc_{ij}]}$ = world position of point **i** in camera **j** coordinates
$\mathbf{R_j}$ = 3x3 rotation matrix for camera **j** (3 unknowns/camera).
$\mathbf{P_{wi}} = \mathbf{[Xw_i, Yw_i, Zw_i]}$ = world position of a point **i** (3 unknowns/point)
$\mathbf{P_j} = \mathbf{[X_j, Y_j, Z_j]}$ = world position of camera **j** (3 unknowns/camera)..

Define the pixel coordinates of point **i** as seen by camera **j** as

$$\mathbf{U_{ij} = S \cdot Xc_{ij} / Zc_{ij}}$$

and

$$\mathbf{V_{ij} = S \cdot Yc_{ij} / Zc_{ij}},$$

where $\mathbf{S = N \cdot f / w}$, in which **f** is the lens focal length, **N** is the number of pixels across the sensor, and **w** is the width of the sensor.

A steepest descent optimization is used to determine the unknowns in $\mathbf{Pc_{ij}}$. An error function is defined for the pixel coordinates as follows:

$$\mathbf{eu_{ij} = U_{ij} - S \cdot Xc_{ij} / Zc_{ij}}$$

and

$$\mathbf{ev_{ij} = V_{ij} - S \cdot Yc_{ij} / Zc_{ij}}.$$

The global performance index to be minimized in the steepest descent procedure is the sum of the squared errors above over all points in all cameras. That is,

$$\mathbf{E = \sum\sum(eu_{ij}^2 + ev_{ij}^2)}$$

The result of this process is location and position angles for each camera. At this point it is now possible to select world points in the scene and map them to a pixel coordinate in each camera frame. This provides a correspondence between all pixels in the different camera views and will allow a complete mapping of a three-dimensional space. The images in Figure 7 show views from four different cameras that observe a common area. Corresponding pixels with $\mathbf{Xw_i}$, $\mathbf{Yw_i}$, with $0 < \mathbf{i} < 96$ and $\mathbf{Zw} = 0$ are shown in yellow.

Figure 7. Corresponding pixels in four image views of a common area.

4. Summary

Fusing multi-spectral imagery into a common 3-D coordinate space allows for a high level of intelligence for inferring and interpreting activities within the region of interest. Detection and assessment from single 2-D imagery can often lead to an unacceptable number of nuisance alarms and very little information for assessment decisions. Using multi-spectral imagery provides additional information about the scene that may not otherwise be available. Combining the imagery from several sensors, either from the same or from different spectrums, will allow increased intelligence for faster assessment decisions. The advantages of combining multi-spectral imagery are often capitalized upon by using collocated, aligned sensors. This allows for composite images with detailed information from each sensed spectrum, which can then be processed for further information. Sometimes sensors cannot be collocated due to terrain restrictions, sensor limitations, or the method of set-up. In these situations, the combined use of the different imagers is still desired, but the fusion methods change. An approach to accomplishing image fusion of multiple imagers that are not collocated is to map each individual two-dimensional image into a common three-dimensional coordinate space. The individual images may be from a single spectrum or the result of fusing images from several

spectrums. The resulting three-dimensional space can be further processed to extract assessment information that is based on multiple multi-spectral image sensors. The use of the methodologies described in this report leads towards the rapid deployment of several imagery sensors without the need for pre-determined sensor locations. The image fusion architecture allows for the use of the best features of each spectrum for increased detection, decreased nuisance alarms, and intelligent assessment capabilities.

References

[1] Fluke IR Fusion Technology, http://us.fluke.com/usen/products/xlink?pid=35656&type=16&trck=irfusion

[2] *Multi Sensor Fusion*, Sarnoff Corporation, Data sheet http://www.sarnoff.com/downloads/research-and-development/vision-technologies/embedded-vision/multi_sensor.pdf

[3] *Arms control verification, sensors, and monitoring systems*, Lab News Labs Accomplishments, http://www.sandia.gov/LabNews/LN02-12-99/la99/arms_story.htm, Vol. 51, No. 3, February 12, 1999

[4] Hall, David L., Llinas, James, Handbook of Multisensor Data Fusion, CRC Press, 2001

APPENDIX A

Selecting a Distortion Model and the Optimal Distortion Coefficients

When long focal length lenses are used (25mm or greater) the image distortion is minimal and can often be ignored. However, with shorter focal length lenses there is generally too much distortion to perform direct mappings between world points and camera points. In order to map camera images from different cameras onto each other or to map targets in world positions onto camera images (or vise versa) an appropriate distortion model must be applied to the data. It is not always obvious which distortion model will provide the best results. This section provides a methodology for selecting a distortion model. The selection of a distortion model requires selection of optimal distortion parameters, $\{k_1, k_2, k_3, k_4, \ldots, x_o, y_o\}$.

1. Defining a Measure of Error for Selecting Distortion Coefficients

In order for a distortion correction model to be effective, it is necessary to optimize the model by selecting good distortion coefficients. This can be done by minimizing the cost based on the deviation from a linear mapping. In other words, the goal is to minimize the error between the true undistorted pixel locations of the measured world points and the model predicted undistorted pixel position of the world points. This error can be represented as the following:

$$E = \sqrt{\sum_{i}^{N} (e_{xi}{}^2 + e_{yi}{}^2)}$$

where

$$e_{xi} = x_{ui} - \frac{a_{11}x_{wi} + a_{12}y_{wi} + a_{13}}{a_{31}x_{wi} + a_{32}y_{wi} + 1} \tag{6}$$

and

$$e_{yi} = y_{ui} - \frac{a_{21}x_{wi} + a_{22}y_{wi} + a_{23}}{a_{31}x_{wi} + a_{32}y_{wi} + 1}$$

Here, (x_u, y_u) = the undistorted pixel positions of measured points obtained using a distortion model with selected distortion coefficients, and

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{7}$$

is a parameter matrix that is used to apply the perspective model transform of world positions of the measured points to their undistorted pixel positions. The derivation of this parameter matrix comes from the definitions of x_p and y_p in equation (4), define in the main document. We assume that we are working in a plane and, therefore, z_w is set to 0. These variables can be rewritten as

$$x_p = \frac{sr_{11}x_w + sr_{12}y_w - s(r_{11}x_j + r_{12}y_j + r_{13}z_j)}{r_{31}x_w + r_{32}y_w - (r_{31}x_j + r_{32}y_j + r_{33}z_j)}$$

and

$$y_p = \frac{sr_{21}x_w + sr_{22}y_w - s(r_{21}x_j + r_{22}y_j + r_{23}z_j)}{r_{31}x_w + r_{32}y_w - (r_{31}x_j + r_{32}y_j + r_{33}z_j)}$$

Since the only variables in these equations are x_w, y_w, and z_w, the rest of the terms can be written as constants, c_{ij}:

$$x_p = \frac{c_{11}x_w + c_{12}y_w + c_{13}}{c_{31}x_w + c_{32}y_w - c_{33}}$$

and

$$y_p = \frac{c_{21}x_w + c_{22}y_w + c_{23}}{c_{31}x_w + c_{32}y_w + c_{33}}$$

To further simplify, we divide both the numerator and denominator by c_{33}. This, in turn, creates a new set of constants, which we label as a_{ij}:

$$x_p = \frac{a_{11}x_w + a_{12}y_w + a_{13}}{a_{31}x_w + a_{32}y_w + 1}$$

and

$$y_p = \frac{a_{21}x_w + a_{22}y_w + a_{23}}{a_{31}x_w + a_{32}y_w + 1}$$

This is the x_p and y_p expressed in (6). Recall that this model for x_p and y_p only works for a pinhole camera in which there is no distortion. Since the measured pixels (x_p, y_p) are most likely distorted, but we want to use this perspective transform in finding the optimal distortion coefficients, we will change the notation of (x_p, y_p) to (u, v). With no distortion correction $(u, v) = (x_p, y_p)$, but as the perspective parameters (a_{ij}) become optimized, (x_p, y_p) will approach their undistorted positions (x_u, y_u) that is expressed in the error function in (6). The pixel positions (u, v) are rewritten as shown below for use in the next step.

$$u = \frac{a_{11}x_w + a_{12}y_w + a_{13}}{a_{31}x_w + a_{32}y_w + 1}$$

and (8)

$$v = \frac{a_{21}x_w + a_{22}y_w + a_{23}}{a_{31}x_w + a_{32}y_w + 1}$$

These equations effectively perform a perspective transform of the world positions of the measured points to pixel positions, assuming that all measured points are on a plane (z_w = 0). We also assume that any rotation of the camera is about its optical center since other rotation is not accounted for.

Remember that the goal is to optimize the distortion coefficients for the distortion model that will be applied to the data. An effective approach for doing this is to apply the distortion model, with default coefficients, to the data and then optimize the coefficients by minimizing the error expressed in (6). The error is best minimized using an optimization algorithm such as steepest descent. Before we can apply the optimization procedure we must first define the perspective model parameters, a_{ij}, that are used in the error function. This step is described in the following section.

2. Using Linear Least Squares to Solve for the Perspective Model Parameters

The perspective model parameters are computed using the world positions of the measured points (x_w, y_w) and the (u, v) pixel positions that result after applying distortion correction to the measured points (x_p, y_p). A linear least squares approach is used to find these parameters. In linear least squares we are trying to solve for the matrix **x** in the equation

$$\mathbf{Ax} = \mathbf{b}.$$

The assignment of **A** begins with the equations in (8). Multiplying both sides of the equation(s) by the denominator and rearranging terms yields

$$u = a_{11}x_w + a_{12}y_w + a_{13} - a_{31}x_w u - a_{32}y_w u$$

and

$$v_p = a_{21}x_w + a_{22}y_w + a_{23} - a_{31}x_w v - a_{32}y_w v$$

These can be expanded further by including the terms for all eight a_{ij} terms (a_{33} = 1 is the 9th term).

$$u = a_{11}x_w + a_{12}y_w + a_{13} + a_{21}\cdot 0 + a_{22}\cdot 0 + a_{23}\cdot 0 - a_{31}x_w u - a_{32}y_w u$$

and

$$v = a_{11}\cdot 0 + a_{12}\cdot 0 + a_{13}\cdot 0 + a_{21}x_w + a_{22}y_w + a_{23} - a_{31}x_w v - a_{32}y_w v$$

Finally, this can be put in matrix form by listing all points i to N of the measured pixels (u, v) and their corresponding world positions (x_w, y_w). The result is

$$A = \begin{bmatrix} x_{w1} & y_{w1} & 1 & 0 & 0 & 0 & -x_{w1}u_1 & -y_{w1}u_1 \\ 0 & 0 & 0 & x_{w1} & y_{w1} & 1 & -x_{w1}v_1 & -y_{w1}v_1 \\ x_{w2} & y_{w2} & 1 & 0 & 0 & 0 & -x_{w2}u_2 & -y_{w2}u_2 \\ 0 & 0 & 0 & x_{w2} & y_{w2} & 1 & -x_{w2}v_2 & -y_{w2}v_2 \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ x_{wN} & y_{wN} & 1 & 0 & 0 & 0 & -x_{wN}u_N & -y_{wN}u_N \\ 0 & 0 & 0 & x_{wN} & y_{wN} & 1 & -x_{wN}v_N & -y_{wN}v_N \end{bmatrix}$$

This is a 2N x 8 matrix, where N is the number of measured points. The parameter matrix **x** is a 8 x 1 and **b** is 2N x 1. These are shown below:

$$\mathbf{x} = \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{21} \\ a_{22} \\ a_{23} \\ a_{31} \\ a_{32} \end{bmatrix} \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ . \\ . \\ . \\ . \\ u_N \\ v_N \end{bmatrix}$$

Linear least squares can now be used to solve for **x**. This matrix along with initial values of distortion coefficients can be used in the steepest descent algorithm, which will minimize the error specified in (6) and find the optimal distortion coefficients. These coefficients will optimize the selected distortion model.

3. Selection of a Distortion Model

A challenge presents itself in determining the appropriate distortion model. There are several distortion models in use today, but it is not entirely straight-forward as to the appropriate model for an application. A study was conducted to compare the distortion models by applying a specific distortion model to measured points, optimizing the distortion parameters for that model, and then calculating the final error between the measured points the optimized distorted points.

Several radial distortion models were considered:

A. $r_u = r_d \cdot (1 + k \cdot r_d^2)$

B. $r_u = r_d \cdot (1 + k_1 \cdot r_d^2 + k_2 \cdot r_d^4)$

C. $r_d = r_u \cdot (1 + k \cdot r_u^2)$

D. $r_u = (-f / 2) \cdot (e^{-2r_d / f} - 1) / e^{r_d / f}$

A fifth distortion model was also included in which both radial and tangential distortion was considered, but it was found that tangential distortion was small enough to be ignored.

A procedure was defined to compare the different models. It is explained in the following steps:

1. Create a point grid in a plane to use for target points. The points should be equidistant apart. Capture images of the point grid with the points filling up the field-of-view of the camera as much as possible. These points represent the world coordinates of the measured points. An example image of a point grid is shown in Figure A-1.

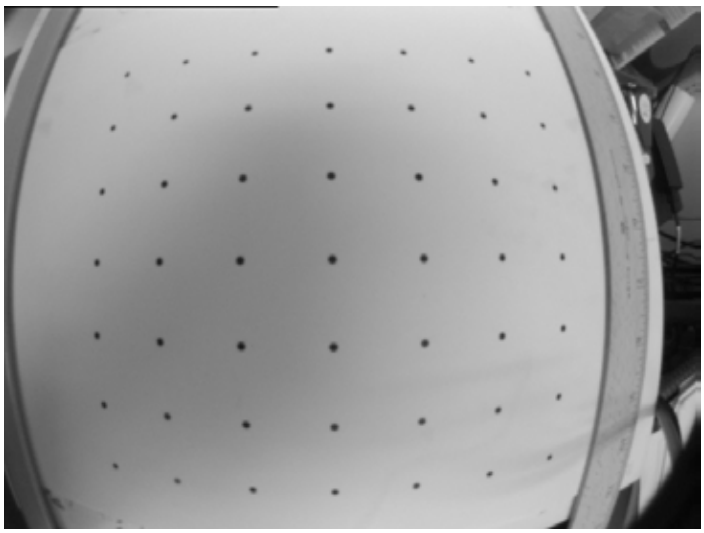

Figure A-1. Image of a 7x7 point grid captured with a 2.6mm lens.

2. In software, segment the points using a simple threshold routine. We used 0.6*(MaxGray-MinGray) as the threshold for each point. This was calculated over a 20 pixel box that surrounded each point. Points are selected in order to specify the world coordinates, with x_w incrementing fastest and then y_w (recall that $z_w = 0$ since this is a planar mapping). Compute the center pixel of each dot on the grid. These will make up the measure pixel values. After this step, we have a list of measured pixel positions (x_{pi} , y_{pi}) and their corresponding world positions (x_{wi} , y_{wi})

3. We found that by normalizing the measured pixel positions, the performance of the distortion models improve. Also, the world coordinate grid should be centered on the middle dot on the point grid in order to minimize bias that is otherwise introduced.

4. Initialize distortion model coefficients (e.g., $k = 0$, $u_o = 320$ $v_o = 240$ for a 640x480 image). Use the steepest descent algorithm to optimize these coefficients. The performance function for steepest descent will use steps 5 – 7 until the error function in step 7 is minimized.

5. Use distortion model with current coefficients to solve for normalized undistorted pixel positions of points.

   Example:

   Using distortion model $r_u = r_d \cdot (1 + k \cdot r_d^2)$ and the relationships in (5):

   $u^* = u + k(u - u_o)\,[(u - u_o)^2 + (v - v_o)^2)]$

   $v^* = v + k(v - v_o)\,[(u - u_o)^2 + (v - v_o)^2)]$

   where

   (u, v) = distorted measured pixel

   (u^*, v^*) = undistorted position of (u, v)

6. Use normalized point/world positions resulting from step 5 to solve for the eight perspective model parameters using linear least squares.

7. Computer the error function from (6), which can be expressed as

$$E = \sqrt{\sum_{i}^{N} (nu_1^{*} - n\hat{u}_i)^2 + (nv_1^{*} - n\hat{v}_i)^2}$$

where

$$n\hat{u}_i = \frac{a_{11} nx_{wi} + a_{12} ny_{wi} + a_{13}}{a_{31} nx_{wi} + a_{32} ny_{wi} + 1}$$

and

$$n\hat{v}_i = \frac{a_{21} nx_{wi} + a_{22} ny_{wi} + a_{23}}{a_{31} nx_{wi} + a_{32} ny_{wi} + 1}$$

The minimized error can be used to evaluate different models. In an ideal case, the model-predicted undistorted measured pixels would coincide perfectly with the undistorted measured pixels. Another measure of comparison is to distort the model-predicted undistorted pixels using the optimized coefficients. In this case the distorted points should overlay the original measured points exactly. Results from this optimization process are shown in Figures A-2 and A-3. In Figure A-2, the green dots represent the undistorted pixel positions using the optimized coefficients in the distortion model, the red dots represent the model-predicted undistorted pixel positions using the

optimized perspective model from (8), and the blue dots represent the distorted position of the model predicted undistorted position using optimized coefficients in the selected distortion model. Notice that the red dots almost completely cover the green dots, indicating a near perfect fit. The image in Figure A-3 was created using the optimized distortion model.

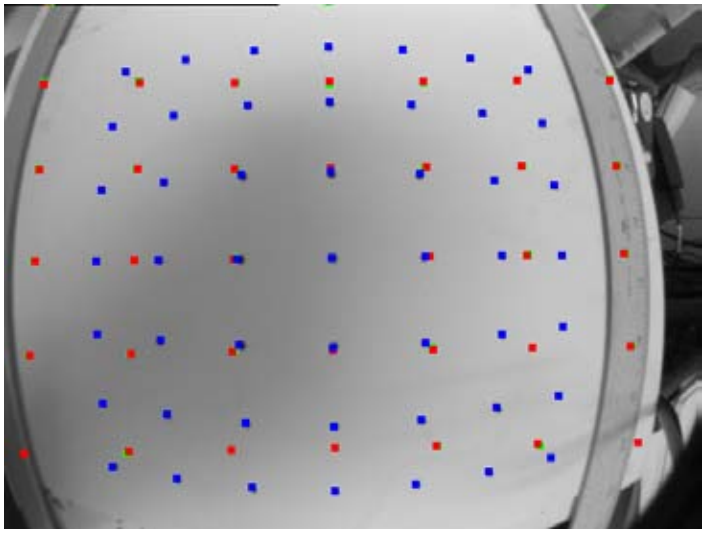

Figure A-2. Results of applying distortion model A to distorted image in Figure A-1.

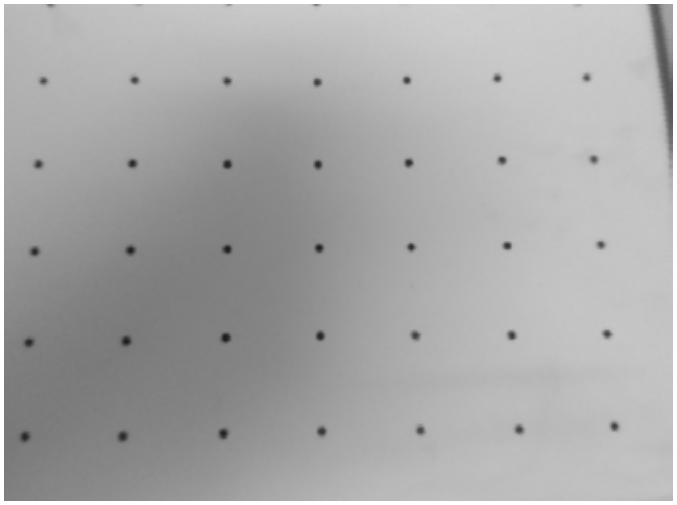

Figure A-3. Undistorted image from Figure A-1 using distortion model A.

Distribution

1	MS	0123	D. L. Chavez (LDRD Office), 1011
1	MS	0780	Stephen Ortiz, 6484
5	MS	0780	Cynthia L. Nelson, 6474
2	MS	9018	Central Technical Files, 8944
2	MS	0899	Technical Library, 9536