GEOFFREY B. IRANI and JAMES P. CHRIST

IMAGE PROCESSING FOR TOMAHAWK SCENE MATCHING

To navigate precisely to a target, the Navy's Tomahawk cruise missile measures its position en route. One positioning technique matches optical images of selected scenes, taken from the missile, to reference maps produced at a mission planning site before launch. The Laboratory has participated in the design, development, and evaluation of this image matching, which enables both the mission planning and the flight mechanics to implement the positioning technique. This article describes Tomahawk's scene selection, image matching, and match reliability, and it illustrates the Laboratory's contributions.

INTRODUCTION

The Tomahawk missile is an autonomously guided strike weapon. This cruise missile flies a planned route from launch to target using an inertial guidance system. Errors in the inertial guidance data exist at launch, increase as the missile flies, and would cause the missile to miss its target if not reduced by en route position measurements. Tomahawk uses three systems to measure positions en route: terrain elevation matching with the Terrain Contour Matching System (TERCOM), satellite positioning with the Global Positioning System (GPS), and optical scene matching with the Digital Scene Matching Area Correlator (DSMAC). Of these three systems, DSMAC produces the most precise positions; its use is required to attack most targets effectively with a conventional warhead and to minimize collateral damage. This article discusses the Laboratory's role in developing DSMAC's operational utility.

The design of the DSMAC system is shown in Fig. 1, along with notes on some of the factors affecting its reliable operation. For each scene selected for a DSMAC update, an optical gray-scale image that shows the scene from high altitude is geodetically positioned; transformed, typically from an oblique perspective to a view along the nadir; and converted into a binary map of local brightness patterns. Because reconnaissance images are generally acquired weeks or months before the DSMAC flight unit images the scenes, changes in time, lighting, atmospheric scattering, and imaging perspective may cause DSMAC images, called frames, to differ greatly from the reconnaissance images. Using image processing and correlation, DSMAC routinely computes the correct position even when differences are substantial between the images taken en route and the stored reconnaissance maps. There are limits to the differences that can be tolerated, however. An important function of mission planning is to estimate these scene-dependent limits and to specify the conditions that will enable DSMAC to determine Tomahawk's position.

Image processing, optical feature modeling, and correlation modeling all are used to forecast DSMAC performance. This forecasting uses analytical mathematical modeling rather than simulation to reduce delays from the computations associated with image and scene analysis. (Analytical modeling represents the typical effects on DSMAC performance of optical features and image processing. Simulation would require specific sets of conditions to be varied over many runs to determine typical effects.) A DSMAC analyst makes these predictions using specialized image display and processing equipment called the Digital Imagery Workstation Suite (DIWS). The analyst identifies the features in the reconnaissance images displayed monoscopically and stereoscopically by the DIWS and measures feature characteristics required by the models. The DIWS software forecasts DSMAC reliability and displays it as false colors superimposed on a binary image of the scene. The analyst reviews these reliability images and adjusts the conditions under which a binary reference map can achieve the broadest use that provides acceptable reliability.

Mission data incorporating DSMAC maps are loaded into a Tomahawk missile shortly before launch and direct the inertial guidance from launch to target. Nearing a scene, DSMAC takes a sequence of short-exposure frames from low altitude, nominally along the nadir, with a video camera. Each frame in the sequence is processed into binary features and is correlated over the map. The resulting sequence of correlation surfaces is further processed to locate the most likely position of the best correlating frame. After one or several successful DSMAC position updates, which follow TERCOM or GPS updates, the inertial navigation is well tuned to strike its target.

Many DSMAC operations apply digital image processing to a large quantity of image data; several image operations in map preparation and reliability forecasting merit note. Data are routinely compressed to store images compactly while preserving the detail required by photogrammetry, forecasting, and correlation. Photogrammetric control points within a geodetically controlled stereo image database (called the Point Positioning Data

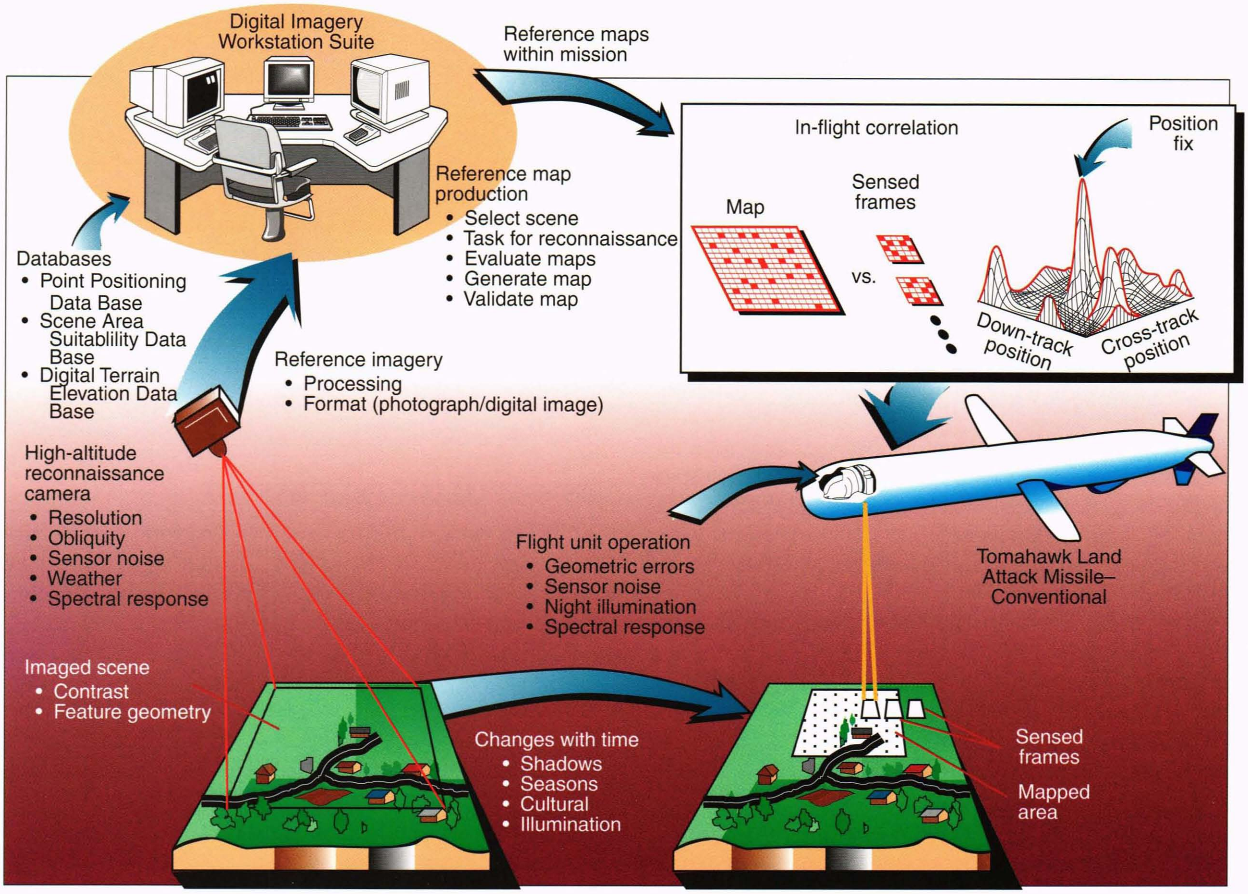


Figure 1. Overview of the Digital Scene Matching Area Correlator (DSMAC) System.

Base) enable DIWS to transform, or "warp," reconnaissance images into a nadir view and create pixels with a manually selected resolution. For computational speed, much of the forecast modeling is implemented by means of look-up tables. Analysis of correlation sidelobes (correlation away from the match point) uses special hardware to carry out the large number (10^6 to 10^{10}) of binary comparisons of frame-sized map subsets and to estimate the correlation noise within which the correct correlation position must be identified.

Aboard the missile, DSMAC performs fewer processing operations but in much less time than was allowed during mission planning. An intensified video camera acquires the missile images and outputs standard video signals. The intensifier gain and gating, which control the effective exposure of the sensor, adjust automatically to maintain a nominal video signal level day or night. Short exposure intervals prevent missile motion from significantly blurring the scene features. In the vicinity of a scene, each frame is sequentially low-pass filtered, reduced in resolution, band-pass filtered, converted to binary brightness, and correlated with the reference map. Correlation surfaces are spatially related arrays of numbers, just as images are. When DSMAC acquires an image within the mapped area, the corresponding correlation surface will include one value for the true (match) position and many values at false positions. DSMAC processing must identify the true position and pass it to the missile's inertial navigator while rejecting all false positions.

In a sequence of frames, the true correlation positions are related by missile altitude, attitude, and velocity. These relationships can be computed accurately from the missile inertial data. They are used differently by the two operational variants of DSMAC, Block II and Block IIA. The earlier variant, Block II, uses a process called voting to compare the computed inertial relationship to the positions of maxima from each pair of frames in each sequential three-frame set. In contrast, Block IIA uses the inertial data to shift each correlation surface so that the true correlations can be summed, even though their locations within the surfaces have not yet been identified. Correlation levels away from the true positions are random; their sums approach the 50% level of random binary matches. Through this superior Block IIA processing scheme, invented at APL, a true position can be found from a sequence of frames that individually have no detectable true correlation.

THE LABORATORY'S ROLE IN DESIGNING DSMAC

The Naval Air Warfare Center's Aircraft Division in Indianapolis, Indiana, invented and initially developed DSMAC and produced the first set of flight units designated Block I. Rudimentary selection and mapping of scenes supported this flight capability. McDonnell Douglas Aeronautics repackaged the flight electronics, creating Block II. This variant retained most of the initial algorithm design, and it structured scene selection and mapping into an operational procedure. As the Technical Direction Agent for Tomahawk, APL worked with these organizations to understand, model, and improve both the flight elements and the mission planning elements of DSMAC. Throughout its involvement with this program, the Laboratory has emphasized the system perspective depicted by Fig. 1. This perspective has guided our analysis, design, and testing to enable us to focus on the topics offering the greatest gains in DSMAC employability and reliability. Our understanding of DSMAC stems from theoretical analysis, laboratory tests, and analysis of flight data.

Three areas in which the Laboratory's efforts have been operationally implemented are DSMAC processing and correlation in the flight unit, performance prediction (that is, forecasting how reliably DSMAC will provide a correct position update for a selected scene), and analysis of complex operational scenes. In 1984, Thomas B. Criss devised and programmed the core of the image processing models that predict Block II's probability of reporting a correct position. McDonnell Douglas incorporated this software in its planning system, which, with refinements, was subsequently used to plan the strikes against Iraq. One refinement was Jeffrey D. Jordan's model of the correlation level lost to perspective distortions. This model and some of Criss's algorithms remain in use. Geoffrey B. Irani, with Kim T. Constantikes and Gary D. Shiflett, invented and demonstrated the Block IIA algorithm. To aid planning for the Gulf War, James P. Christ devised and programmed algorithms to enable operational DSMAC analysts to measure and evaluate scenes with more reality and precision than possible with standard methods. Irani and Christ worked closely with operational analysts to evaluate DSMAC use under marginal conditions.

DSMAC IMAGE PROCESSING

Image processing in Block IIA differs significantly from that in Block II, as Fig. 2 indicates. The newer electronics in Block IIA enable entirely digital processing to correct nonuniform response of the sensor, spatially filter in two dimensions, add correlation surfaces, and adaptively detect correct missile positions. When DSMAC was designed in the late 1970s, the only practical approach was to process the video with analog electronics until the video image was compressed in

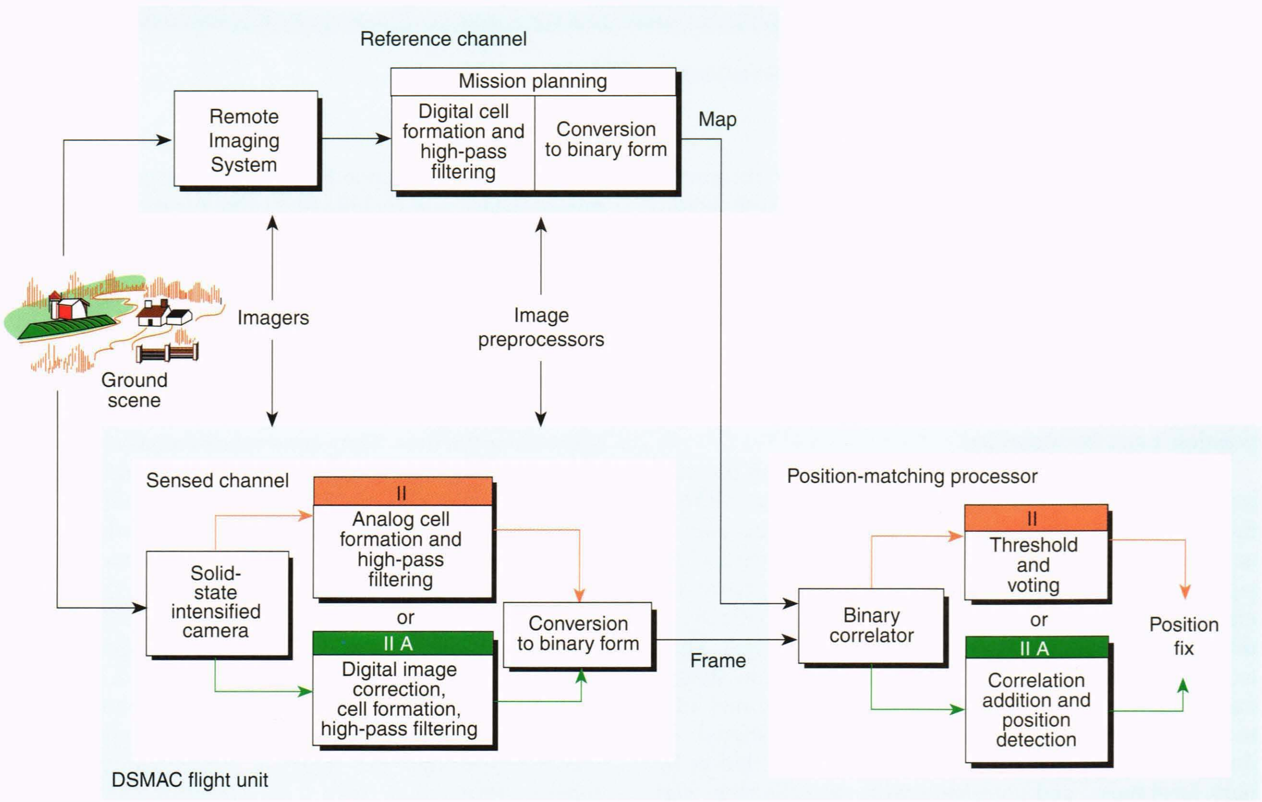


Figure 2. Dual-channel designs of DSMAC for Block II and IIA. The reference channel consists of a reconnaissance camera and elements of the Tomahawk Mission Planning System. The sensed channel is the DSMAC unit itself.

both spatial and brightness dimensions to a small binary array. This analog processing performs some of the operations now realized in digital hardware in Block IIA. However, the filters in Block II are essentially one-dimensional and aligned along the video raster line. The vertical averaging that reduces the down-track resolution operates on binarized video lines. These processes introduce substantial distortion between gray-scale and binary features. The distortion reduces correlation levels somewhat, but mostly it interferes with visual recognition of the stable binary features when maps are being selected. In Block II, as in Block IIA, each binary frame is correlated over the map. Unlike Block IIA, however, Block II requires a correlation level to exceed a fixed threshold that must be selected during mission planning. A fixed-size buffer is used to store correlation levels above the threshold; once this buffer is full, any additional values above the threshold are lost. As noted in the Introduction, the two DSMAC variants also use different methods to determine if the correlation positions are consistent with inertial data. Analysis and measurements show that Block IIA far exceeds Block II in its capacity to correctly identify missile position.

The DSMAC functions as a dual-channel match process. The reference channel consists of a reconnaissance sensor and the Tomahawk Mission Planning System, which constructs the reference map; the sensed channel is the DSMAC flight unit. Time always separates information in the two channels; reference processing may be completed months before the map is flown.

The DSMAC maps are produced during mission planning. Mission planning has three objectives: (1) to identify areas suitable for DSMAC operation, (2) to create DSMAC maps for the selected areas, and (3) to estimate how reliably DSMAC will find the correct position of the missile using these maps.

The initial identification of suitable DSMAC scene areas is primarily a manual process, potentially aided by the Scene Area Suitability Data Base. A skilled analyst visually reviews the reconnaissance imagery to find areas that have suitable characteristics for DSMAC operation. During this review, the analyst uses image processing hardware to enhance image contrast and control display resolution.

After a scene has been selected, a DSMAC map must be created from the reconnaissance image; the original reconnaissance image is not used. Instead, the image is spatially averaged and subsampled to reduce its resolution, digitally filtered to remove the local average brightness, and converted from a multibit gray-scale image to a single-bit (binary) image, in which the binary state corresponds to the sign of the filter output. A sample gray-scale reconnaissance image is shown in Fig. 3a; the corresponding binary image is shown in Fig. 3b. A subset of the binary image is extracted to obtain a DSMAC map.

The DSMAC uses reduced-resolution binary images for two reasons. First, the lower resolution reduces the flight unit's data and computational loads. Reconnaissance images contain 50 thousand to 25 million pixels, and DSMAC video images contain 50 to 100 thousand pixels. Correlation is a computationally intensive process that scales as the fourth power of the image resolution. Even with modern electronics, processing full-resolution images in real time is impractical. Second, the appearance of scene features changes over time. Detailed information about shape or local contrast rarely contributes to a correlation match. Rather, the pattern formed by features distributed throughout a sensed frame enables correlation to identify the correct position.

(a) (b)

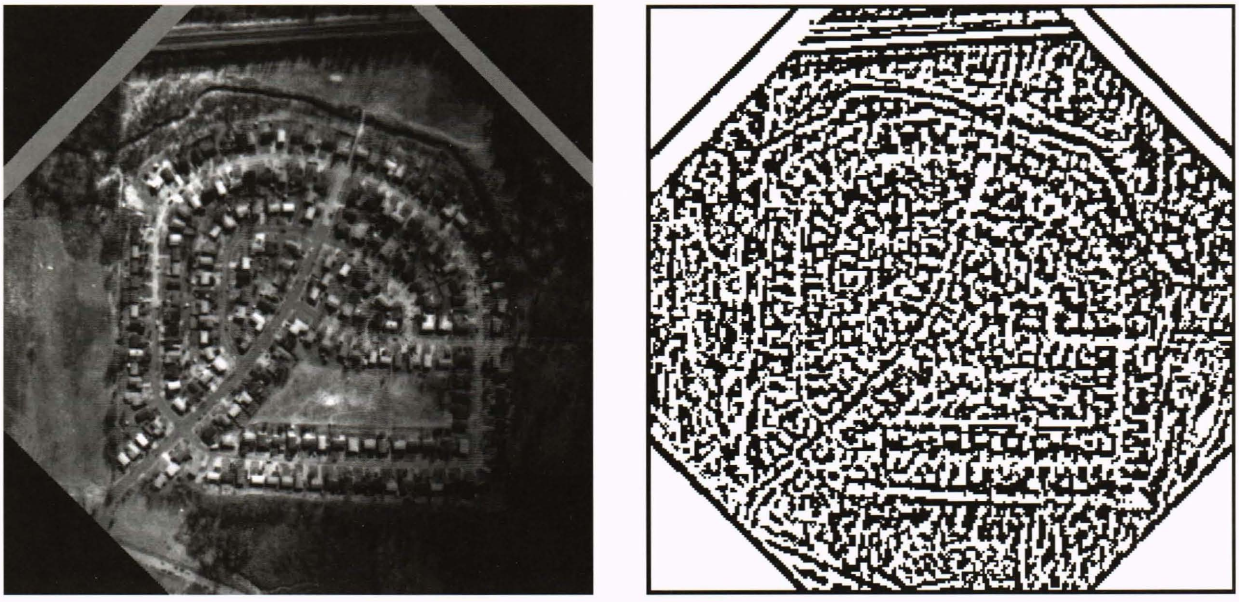


Figure 3. (a) Sample reference image of an urban area and (b) the same image after spatial averaging, filtering, and conversion to binary form.

During flight, the DSMAC unit uses an intensified charge-coupled device video camera to acquire images of the ground. A high-intensity strobe provides illumination at night. As in the reference channel, the sensed channel spatially averages the video and subsamples, filters, and binarizes the images to produce binary frames. (With Block IIA the process is all digital; with Block II, analog.) Figure 4 shows the gray-scale and corresponding binary image for a maximally sized frame acquired over the scene of Fig. 3a.

As the missile acquires each frame, the DSMAC unit correlates the frame with the reference map to determine where the two images correctly match. Correlation compares the frame at every possible location, counting the number of binary pixels that match between the frame and the map. The higher the correlation value, the more likely that this position is the correct location of the frame within the map; the highest correlation value obtained is called the correlation peak. Figure 5 is a plot of the correlation values around the peak location for the frame in Fig. 4a. The smaller peaks around the correlation peak are called sidelobes. They are effectively noise in which the correct correlation must be identified. Mission planning predicts the likelihood that the true peak will be larger than all the sidelobes.

(a)

(b)

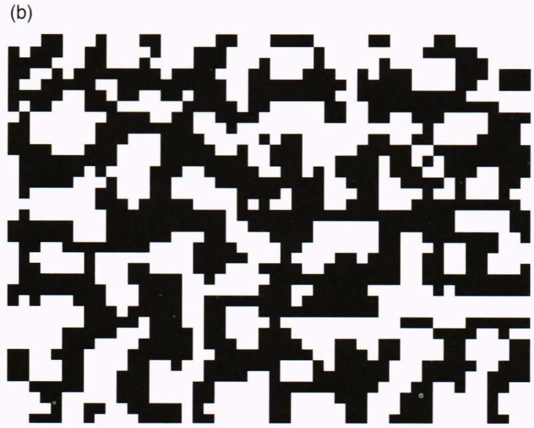


Figure 4. (a) A gray-scale video frame from DSMAC during flight and (b) its corresponding maximally sized binary image.

The maximum correlation value could occur at the wrong position within the reference map, leading to a substantial error in the measured position. To prevent this possibility, several frames are acquired and correlated with the map. Block II and IIA use different methods to determine if a correct position is detected.

In Block II, the reference map is sized so that at least three frames are acquired over the map. The DSMAC compares the relative peak positions of three consecutive frames with relative positions computed from the known velocity, altitude, and attitude of the missile. If at least two of the three positions are consistent with this information, the frame with the highest correlation level is used for the position update; otherwise no position update is provided.

In Block IIA, the individual correlation surfaces are shifted to compensate for missile motion. The shifted surfaces are then added together to produce a summed correlation surface. These shifts align the correct correlations independently of their levels and locations to produce a peak in the summed correlation surface, even when individual surfaces do not have correct peaks. Sidelobe structures are uncorrelated with missile motion and will not line up after the shift. Thus, the summed surface has smaller sidelobe structures than the individual surface. As more surfaces are summed, the sidelobe structures approach a constant level, which is the mean sidelobe level. The location of the highest value in the summed correlation surface provides the position update.

Predicting Performance

During mission planning, an analyst uses several computer algorithms to estimate how reliably DSMAC will provide a correct position update for a selected scene. These algorithms model the behavior of scene features and the effects of feature changes on DSMAC processing. Some of the algorithms simply simulate calculations performed in the missile. Others approximate computationally intensive processes. Temporal changes in features, such as moving shadows or seasonal changes in foliage, affect reliability and must be forecast to predict periods of reliable operation. Other characteristics that must be represented are imaging noise, errors in geometry, and similarity of feature patterns. Rather than discussing each model separately, we illustrate the processing that APL performs by describing a single example in detail—the analysis of errors in horizontal geometry.

Analyzing Horizontal Geometric Errors

Differences between the geometry of scene features in a sensed frame and the geometry of the corresponding features in the DSMAC map significantly decrease the correct correlation level. These geometric differences include scale and rotational errors, as well as a small translational error called phasing. The amount of geometric error varies from frame to frame, map to map, and flight unit to flight unit. Consequently, performance prediction models only the average effect. The current model is based in part on a theoretical result, which shows that small geometric errors affect correlation as if the images were low-pass filtered (Mostafavi and Smith[1]). In a

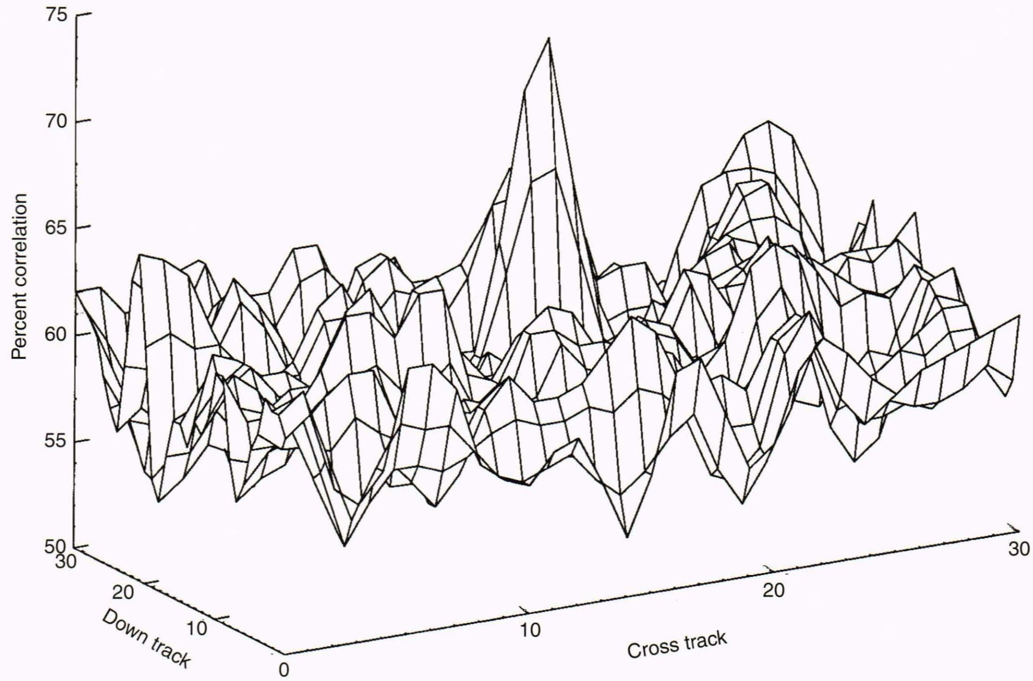


Figure 5. A subset of a DSMAC correlation surface, centered around the peak correlation position.

subsequent empirical study, Criss[2] developed a 3 × 3 finite impulse response filter that represents the losses caused by geometric differences.

The work to determine the filter coefficients was based on a limited data set. In addition, the data included several other sources of correlation loss. Recently, data more suited for evaluating the accuracy of this model were obtained, and a more accurate process was developed. The new evaluation is based on comparing the correlation losses caused by horizontal geometric errors experienced during a test flight to the correlation losses predicted by the model.

Conceptually, the correlation loss due to geometric errors can be measured by comparing the correlation level for a frame acquired by the DSMAC unit with the correlation level that would be obtained if there had been no geometric errors. The difference between the two correlation levels would provide a single measure of the loss due to geometric errors. Because the horizontal geometry model estimates the average expected correlation loss for each possible frame location, rather than predicting the actual loss for any single frame, evaluating the model requires a statistical comparison of the actual losses with the predicted average losses.

Computing the actual correlation loss requires a video frame without the geometric errors. In practice, acquiring such a frame is difficult. A large number of frames are involved with varying geometric errors. (A single sequence of test flights in 1991 yielded nearly 10,000 frames.) The geometric errors can be measured by adjusting the geometry of a recorded DSMAC flight video to match that of a reconnaissance image. In captive test flights, but not aboard a missile, the gray-scale video for each frame is digitally recorded for processing in APL's Mission Planning Development Laboratory. The geometric errors in each frame are measured through repeated computer trials. A set of geometric corrections to the frame geometry is computed, and the resulting frame is compared through correlation to the reconnaissance image. This process is iterated using different geometric corrections until a best correlation is obtained. The correction providing the best match measures the geometric error for that frame. For some frames the process does not converge, and no useful geometric measurement is produced.

Three problems had to be solved for the correction process to be automated. Each is discussed in detail in the sections to follow. We had to determine how to compute a corrected frame, given a measurement of the geometric errors; design an algorithm that allows the computer to determine when the sensed and reference images are properly aligned; and decide how to efficiently search the space of possible geometric errors to find the errors that actually occurred.

Computing a Corrected Frame Given the Geometric Errors

Producing a geometrically corrected sensed frame given a set of geometric errors is a relatively simple task. The video digitizer measures the sensed brightness throughout the video frame, producing a rectangular array of pixels. Geometric errors present when the frame was acquired move the pixels from their ideal locations without changing the measured brightnesses. If we knew where the pixels belonged, we could construct a corrected frame by moving each pixel value to its proper location within the ideal frame.

This approach transforms a sensed frame with geometric errors into a new sensed frame without geometric errors. However, correcting some scale and rotation errors could require pixels that are not present in the distorted sensed frame. In practice, we prefer to transform a subset of the reference image into the (distorted) geometry of the sensed frame, since the reference image provides pixels surrounding the frame. The transformation is defined by a set of equations derived by Ratway.[3] For displays, however, a corrected sensed frame is used instead of the transformed reference image, which requires inverted transformation equations.

Conceptually, transforming a subset of the reference image is simple. The transformed image will be the same size as a sensed frame, since evaluation of the transformation is based on correlation. The origin of the coordinate system for the transformed image is placed at the center of the frame. The computer then steps through each pixel location in the transformed image, computes where that pixel lies in the original reference, and uses bilinear interpolation to estimate the gray level from the four neighboring reference gray levels.

Figure 6 shows uncorrected and corrected frames: Fig. 6a is the original gray-scale frame acquired during flight, and geometric correction produced the image of Fig. 6b. The original image did not cover a large enough ground area; the gray band around the corrected image corresponds to the area not imaged. The frame is rotated slightly within the gray band; this corresponds to a yaw error in the original frame.

Determining When the Sensed and Reference Images Are Properly Aligned

The second problem is to design an algorithm that allows the computer to determine when the sensed frame and reference image are properly aligned. One can visually compare feature edges in the two images. Computer vision techniques to locate edges and segment features from an image are complex and do not apply well to this situation, particularly because perspective differs between the images. It is difficult or impossible to have a computer consistently find suitable alignment features. The simplest way to automatically compare two images, I and J, is to compute the average gray-level difference between the images. We could use the mean square difference between the images:

$$\mathrm{diff}(I, J) = \frac{\sum_{m,n} (I_{m,n} - J_{m,n})^2}{\sum_{m,n} 1}, \tag{1}$$

where m, n denotes individual pixel locations. As the geometry is corrected and the two images I and J become more alike, this difference should decrease.

Unfortunately, the difference function defined in Eq. 1 is not sufficient. The reference image and sensed frame were acquired by different sensors at different times. The lighting for the two images is different, leading to differences in the average brightness of the image as well as differences in the image contrasts. If these differences are severe enough, Eq. 1 provides a poor measure of image similarity.

We need to suppress gray-level differences that do not represent horizontal geometry. This is the same challenge that must be met for proper DSMAC operation. The DSMAC filter removes the local average gray level and, thereby, suppresses differences in average brightness. Binarization effectively suppresses differences in image contrast. DSMAC has demonstrated the effectiveness of this approach, so the same techniques are used to compare the reference image and sensed frame.

(a) (b)

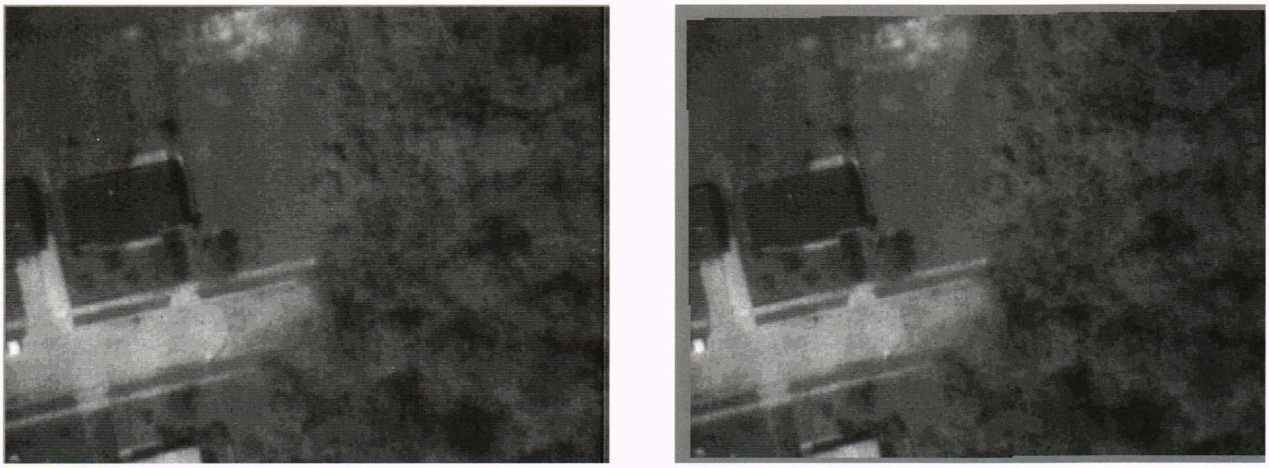


Figure 6. A gray-scale video frame from DSMAC during flight. (a) Original, uncorrected frame; (b) frame corrected to remove geometric errors.

After the two images have been filtered and binarized, the difference between them can be easily measured using binary correlation with exclusive-or logic, indicated by the symbol $\otimes$. Because relative position is a parameter in the geometric correction, only the difference at a prescribed location need be computed. This difference is the fraction of pixels in the two images that have different binary states:

$$\mathrm{diff}(I, J) = \frac{\sum_{m,n} I \otimes J}{\sum_{m,n} 1} . \qquad (2)$$

Performing this calculation at full resolution provides the best sensitivity to geometric distortion, so the images are not averaged and subsampled before filtering or binarization.

Searching the Space of Possible Geometric Errors

The remaining task is determining the transformation parameters that provide the minimum difference between the sensed frame and the reference image. The geometric transformation depends on six variables: roll, pitch, yaw, altitude, cross-track position x, and down-track position y. Trying just five values for each would require us to test more than 15,000 possible transformations, which would consume too much time. The telemetry information that is acquired with the sensed frames includes measures of roll, pitch, yaw, and altitude. Two of these parameters, roll and pitch, are relatively small, and small errors in their values have a fairly small effect on the sensed frame geometry. Therefore, the roll and pitch values from the telemetry are used without modification, leaving four parameters to be determined by computer trials.

The difference measure defined in Eq. 2 produces numerous local minima as the transformation geometry is varied. These minima prevent the use of a classical gradient-based search technique, which would find a local minimum rather than the true global maximum. To find the global minimum, the transformation parameters are divided into two pairs: x and y, and yaw and altitude. For each pair, the difference measure is evaluated over a grid of points centered at the current estimate of the transformation. The point in the grid that provides the minimum difference is used to update the current estimate. After this has been done for both pairs, we have a new estimate of the true transformation. This process is repeated several times, using a more closely spaced search grid at each iteration.

This grid-based search finds the approximate location of the global minimum as long as the minimum is within the region being searched. Once the approximate location of the minimum has been found, however, a gradient search is more efficient. Thus the final transformation is determined using a gradient-based search.

Evaluating the Horizontal Geometric Errors

As part of the effort to evaluate the accuracy of the performance prediction models, a large number of flight frames were run through the geometric correction described above. The geometric parameters obtained through this process provide the best available measure of the geometric errors encountered during flight. The corrected frames are also useful for other analyses.

The most obvious use for the frames is to determine how much correlation loss occurs due to geometric errors in flight. The increase in correlation level after geometric correction is plotted in Fig. 7. This figure shows that increases of approximately 7% of perfect match are obtained. If this correlation improvement could be realized in the missile, it would represent a significant increase in DSMAC capability. A comparison of the measured losses due to geometry with the predicted losses shows that geometric losses are consistently underpredicted. For one scene studied, the average predicted loss was 2%, whereas the average actual loss was 8%. To compensate for the underpredicted geometric losses, the predictions appear to overstate the loss due to system noise.

A second use for the corrected frames is to create a mosaic image, where the individual frames have been combined to provide one image of the entire scene. Figure 8 shows a frame-mosaic image of Duke airfield. Each pixel in this image is the average of all corrected frames that include that pixel. Changes in the scene features can be identified by comparing the frame-mosaic image to the reference image from which the DSMAC binary map was produced.

Other useful images can be produced from sets of corrected frames. If the frames are binarized and then combined to form a mosaic, each pixel estimates the probability P_{white} that a given pixel will binarize white. The P_{white} is an intermediate figure of merit computed by the prediction models and can be evaluated by comparison with this mosaic image.

A more important parameter than P_{white} for predicting the correlation peak level is the probability of correct binary assignment, P_{cba}; that is, the probability that after

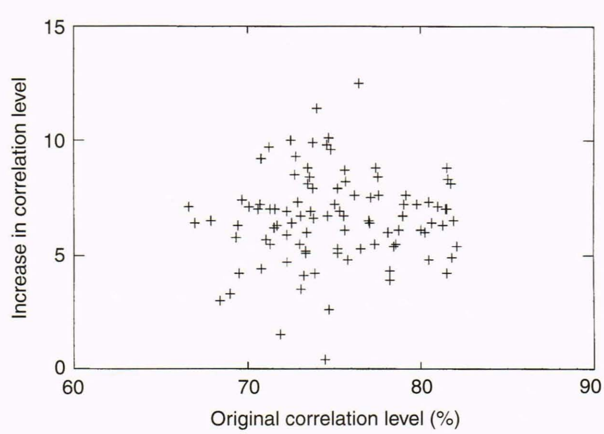


Figure 7. Increases in correct correlation levels achieved by removing geometric errors.

Figure 8. A gray-scale mosaic of DSMAC frames from a test flight. Individual sensed frames are geometrically corrected and pieced together to form the mosaic image.

a frame is converted to binary form, a given pixel will be in the same state (black or white) as in the reference map. We can combine the corrected frames with the reference map to produce a P_{cba} mosaic. Figure 9a shows a P_{cba} mosaic, and Fig. 9b shows the predicted P_{cba}. A comparison of these two images measures how well we predict the correlation peak level. One striking difference between the two is that a large area in the measured P_{cba} mosaic is white, corresponding to a P_{cba} of 1.0. In contrast, the predicted P_{cba} is almost never white. A characteristic of the model used is that the predicted P_{cba} is never less than 0.5. In contrast, the measured P_{cba} also includes values that are less than 0.5; in some areas the measured P_{cba} is even zero.

Once a P_{cba} mosaic and a predicted P_{cba} image have been produced, comparison between the two is easy. A simple form of comparison is a two-dimensional histogram (sometimes called a co-occurrence histogram). The two-dimensional histogram $h(x, y)$ is generated by counting all the locations where the first image is equal to x and the second image is equal to y. Figure 10 shows a two-dimensional histogram produced from the two images in Fig. 9. The horizontal streaks in this figure are caused by the quantization of the measured P_{cba} values. (The number of distinct P_{cba} values that can be observed depends on the number of frames that image each map pixel projected onto the scene. Since the number of frames is limited, the number of possible P_{cba} values is small.) Because the predicted P_{cba} is never less than 0.5, the left half of the plot is completely black. The histogram shows a large spread in the measured P_{cba} for all values of predicted P_{cba}.

Comparing the average measured P_{cba} with each predicted P_{cba} value provides another indication of prediction accuracy. Figure 11 compares the two graphically. On average, there appears to be a relationship between predicted and measured P_{cba}. For most values of predicted P_{cba} the curve lies above the diagonal, implying that the predicted P_{cba} is less than the measured P_{cba}. This result agrees with the earlier observation that the predicted loss due to geometric errors is lower than the measured loss and that the predicted loss due to system noise is higher than actual. Since the loss from geometry has been removed from this comparison, the net result is more predicted loss (smaller predicted P_{cba}) than measured.

We have illustrated image processing for DSMAC performance prediction by discussing our analysis of horizontal geometric errors. This is only one of the characteristics modeled for performance prediction. Although other models apply image processing in different ways, the goals are the same—to predict DSMAC performance, and to measure the accuracy of the predictions.

SPECIAL SOFTWARE FOR OPERATIONAL SCENE ANALYSIS

Tomahawk uses image processing for purposes other than the development and application of standard predictive algorithms. In certain circumstances, special analyses must be done of images showing operational scenes having marginal predicted performance. At APL, J. P. Christ developed two menu-driven sets of software, Analyst and Compare, for such analyses.

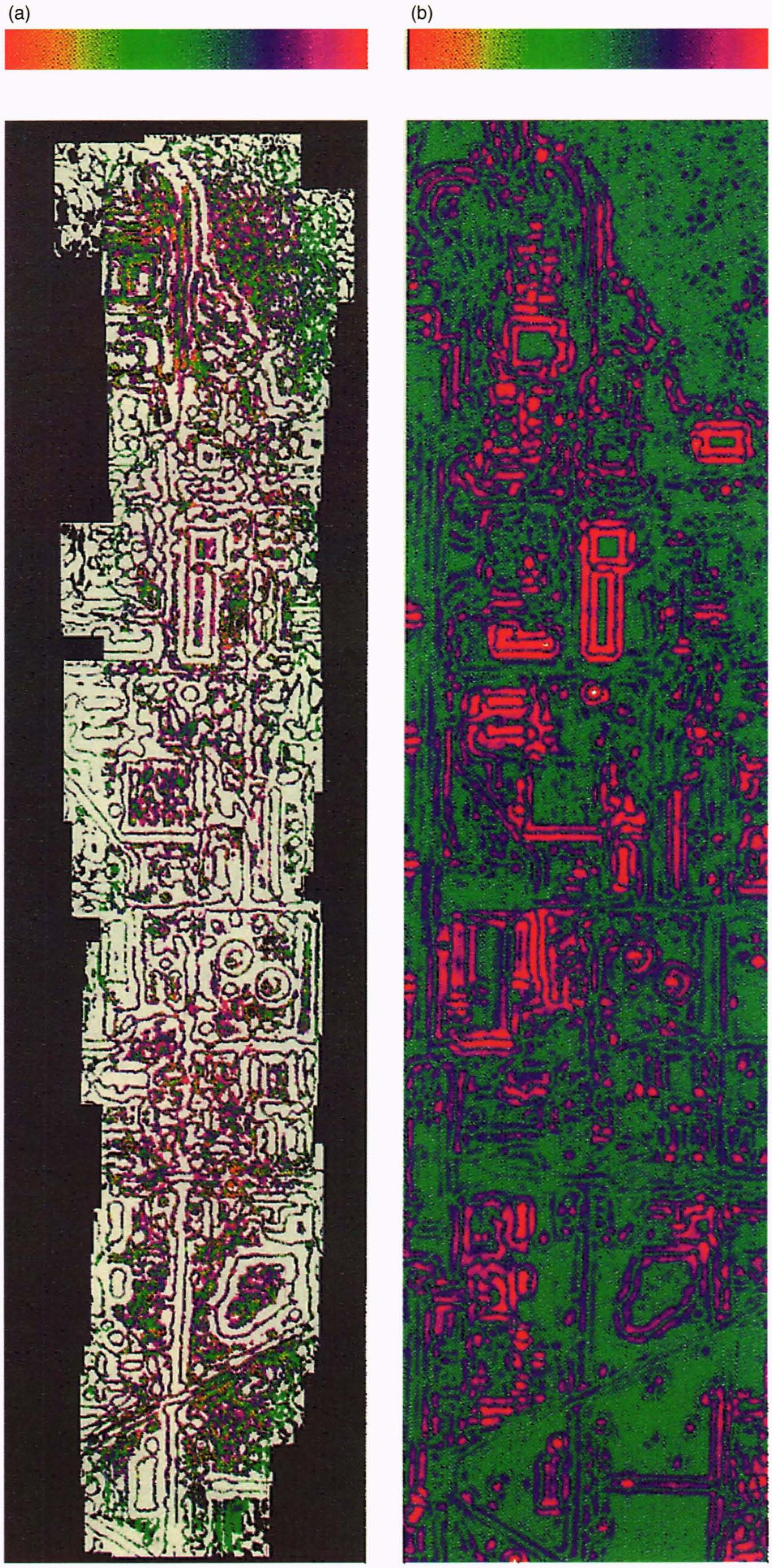


Figure 9. The probability of correct binary assignment P_{cba} for a test scene. (a) The P_{cba} measured during a series of test flights; (b) predicted P_{cba}.

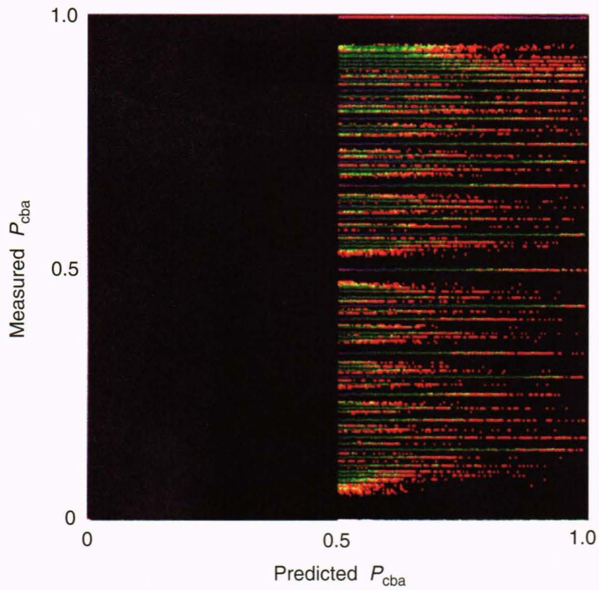


Figure 10. A two-dimensional histogram comparing predicted and measured P_{cba} values from the images in Fig. 9.

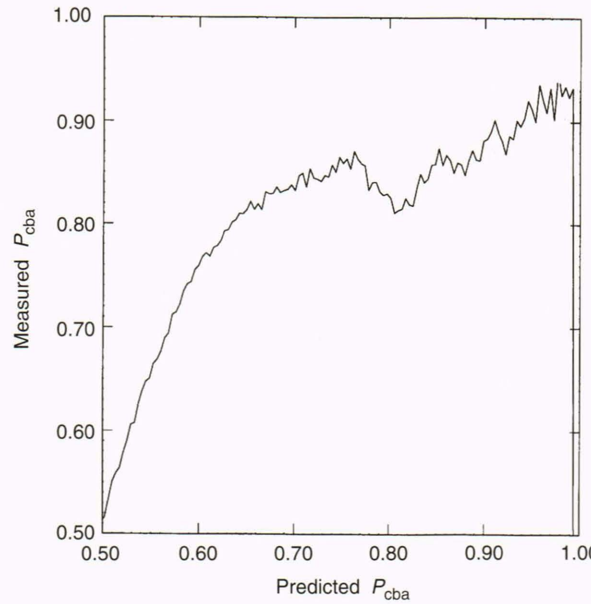


Figure 11. The average measured P_{cba} as a function of predicted P_{cba}.

The Analyst Program

Analyst, which is written in C, aids analysis of scene instabilities. Instabilities are features whose appearance changes between the time the reconnaissance image is acquired and the time the missile flies over the scene. Analyst provides tools for two activities: measurement of feature composition in areas of interest and gray-level transformation of selected image features.

Instability analysis uses the fraction of the scene area affected by certain visually recognized instability classes to predict correlation losses from instabilities. Two examples of these classes are shadows that move during the day and mapped shadows that disappear at night. To help the user measure areas affecting correlation for each class, Analyst color-codes interactively selected gray-scale ranges (slices) of a displayed image. For each slice within a selected image area, the user selects minimum and maximum gray levels. The software colors each slice uniquely in real time as the user adjusts the gray-scale limits. When up to 10 slices have been defined, the software counts the number of pixels within the slices and reports it as a fraction of the scene area.

Gray-scale values mean little to a DSMAC analyst. Rather, the features in the area to be mapped and their instability classes matter. Consequently, Analyst allows slices to be defined visually. The Analyst program displays the image on a monitor and colors the pixels that fall within the selected slice. The bounds on the slice can be varied using a mouse or track ball; the color-coded display is updated in real time to reflect the new values. While watching the display and varying the slice bounds, the user can generally isolate the instabilities or their sources (for example, tall objects) well enough to predict performance. Performance prediction assumes that only one instability affects a single point in the scene, so an image area should be assigned to one instability at most. Analyst can adjust the limits of two slices so that their gray levels do not overlap and the same area is not counted twice.

Two or more instability classes sometimes share the same gray-scale range. In this case Analyst allows polygonal subdivision of the image; each defined region must then be separately classified. After gray-scale slicing has been done satisfactorily within each area of interest, Analyst reports the total area assigned to each instability class within all areas of interest. Analyst then computes the fractional areas of all classes across all regions.

Two sources of correlation loss, moving shadows and verticality, require measurement of feature size and height. The images used to produce DSMAC maps are geometrically controlled so that each pixel represents a fixed distance on the ground. Analyst allows the user to draw a line along a horizontal feature boundary within the image and then reports the length of the line in feet. This allows measurement of horizontal dimensions. Analyst also enables height to be measured using one of two approaches. The obliquity of the reconnaissance image is always known. If a line along a vertical feature edge is drawn, Analyst transforms this length to height using trigonometry. The second principle is also trigonometric. If the length of a shadow of an object can be marked with a line and the elevation of the Sun is known, Analyst will calculate the object's height. If the Sun's elevation is not known, Analyst asks the user to measure both a vertical edge and the corresponding shadow; this measurement allows Analyst to calculate the Sun's elevation.

DSMAC performance prediction includes a model for correlation loss when the shadows shown in reconnaissance are absent in DSMAC frames, as occurs at night or with a high overcast. The model assumes the shadows are not a major source of image contrast within the scene. This assumption is convenient, but often does not apply. Analyst offers a more accurate measure of shadow loss for such cases.

The most accurate way to measure the loss due to the disappearance of shadows would be to acquire a second reconnaissance image without the shadows, and then measure the binary change between the images. (The Compare program described later enables the measurement of binary change.) Unfortunately, acquiring an image without shadows is rarely possible, since a cloud must block the Sun but not the reconnaissance perspective.

Analyst circumvents the need for a second reconnaissance image by allowing the user to suppress shadows in the original image. The user first identifies the shadows, either by gray-scale slicing or by using polygonal areas of interest. Analyst supports a linear transformation of the gray-scale values within each area of interest or gray-scale slice. The user interactively adjusts the transformation with track ball or mouse while viewing the modifications in real time. By using multiple gray-scale slices for a shadowed area, the user can often blend the shadows into their surrounding features. This is an inherently subjective process, but it produces fairly realistic representations of scenes without visible shadows.

Once the user has suppressed the shadows within the image, Analyst processes both the original and the modified images through the DSMAC filter to obtain binary images for the scene with and without shadows. To show how the shadows influence the binary image, Analyst alternately displays (flickers) these two binary images with either of the two gray-scale images. Analyst also displays the area where the binary images differ and flickers this difference with either gray-scale image. These displays help the user determine what features are affected by the shadow changes. Analyst also measures the impact of these changes on DSMAC correlation by computing the average binary change within each possible DSMAC frame. Analyst displays the average change as a pseudo-color image overlaid upon the binary features in the shadowed image. The user examines this display for local areas with unacceptably high correlation loss and selects an average loss to enter into the standard performance prediction software.

The Compare Program

The performance prediction algorithms estimate the future performance of a DSMAC reference map. The map has a limited operational lifetime that depends on the instabilities affecting the scene. At planned intervals, a current reconnaissance image is compared with the image used to map the scene. If significant changes have occurred, new maps are created. This process is called map maintenance.

The Compare program replaces visual comparison of printed images during map maintenance with a quantitative analysis of digital images. This software measures binary changes between two images of the same scene. The resulting measurement can be entered into the standard DSMAC performance prediction software to determine if the map still provides adequate reliability.

Compare assesses pairs of images pixel by pixel. For meaningful results, the geometry of the two images must be nearly identical; that is, features must occupy corresponding pixels within the two images. Even a small geometric difference causes an artificially large change to be measured. Therefore, Compare must first remove any geometric difference between the two images. The user measures the geometric differences by identifying a set of identical ground-level points in both images called tie points. Figure 12 shows an example of the display that Compare provides to guide selection of tie points. The left side of the display shows the two images being compared. The right side shows a portion of each image at full resolution. The upper images are an urban area with a large number of trees. The lower images show the same area after the leaves have fallen. The user placed the two green squares on the right side of the display to define a tie point. After the user has marked corresponding areas in this manner, the program performs a gray-scale correlation in the vicinity of the square to more accurately locate the tie point. This correlation reduces the accuracy with which the user must mark the tie points. The colored squares on the left side of the display indicate previous tie points, with the color-coding indicating how well the areas matched during the local correlation.

When enough tie points have been selected, the program automatically determines and applies a transformation to one image to suppress the geometric differences. Compare allows the user to flicker the two images on the display to observe how well the geometry has been corrected. If geometric differences remain, Compare provides a tabular display that measures how well each tie point fits the geometric model, which helps the user choose tie points to position or replace.

Once the geometry has been adequately matched, Compare provides several ways to examine change within the scene. The first tool is a colored overlay that identifies where the DSMAC binary image has changed. If the changes are substantial and affect features that were considered stable when the map was originally prepared, then a new map is required.

A second measure of change is the average binary change for each possible DSMAC frame over the scene. This measure indicates how much correlation loss will occur if the map is used, and is displayed as a false-color overlay. The user checks the display for any local areas with unacceptably high correlation loss and to determine the average correlation loss over the scene area. The average loss can then be used as an additional instability value for the DSMAC performance prediction software; the predicted probability of correct update P_{cu} provides a familiar indication of map reliability. Figure 13 shows the frame-averaged binary changes between the two images in Fig. 12. The number of changes is color-coded and placed at the center of each possible frame. Green and blue represent little or no change; red and yellow represent substantial change. The difference in foliation between the two reconnaissance images led

Figure 12. A screen image of tie-point generation using the Compare program. (The images on the right are portions of those on the left shown at full resolution.) The upper two images are an urban area with a large number of trees. The lower two show the same area after the leaves have fallen. The green squares on the right are the tie points currently being selected; the green and red squares on the left are previously generated tie points.

to substantial binary changes. A DSMAC map of this scene was produced from summer imagery and is unlikely to correlate reliably during winter.

The frame-average loss just described is calculated at the correct geometric match point between the two images. However, DSMAC does not know the correct match point, so it uses the point at which maximum correlation occurs. For some types of instability (shadows, for example) the maximum correlation is not at the true match point. When shadows move from one side of an object to the other, the correlation peak is also likely to move. Compare measures the shift in correlation position as well as the correlation loss that occurs at the peak location. To do this, Compare selects frame-sized subsets from one image and correlates each subset over a neighborhood around the true match point. Computing this correlation for every possible frame within the map would take too long, so Compare selects these subsets using a grid pattern spanning the map. If images are available for different times of day, this analysis can be used to estimate shadow losses.

CONCLUSION

Tomahawk effectiveness requires DSMAC to be reliable in a wide variety of scenes and environmental conditions. To meet this requirement, the Laboratory has

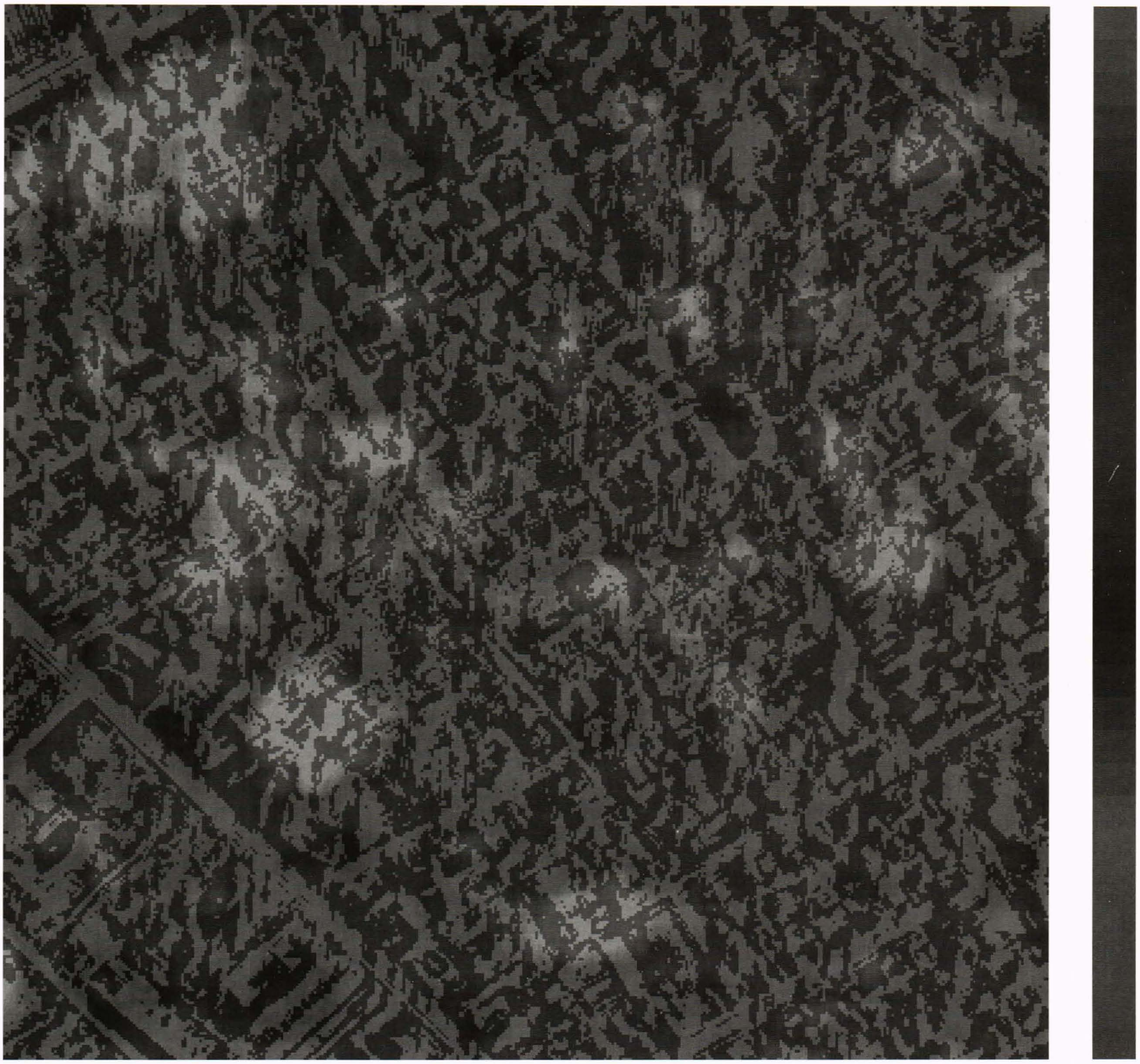


Figure 13. The average binary change within a DSMAC field of view. The color bar on the right indicates the magnitude of the change, with green representing no change, blue a small change, and red and yellow large changes.

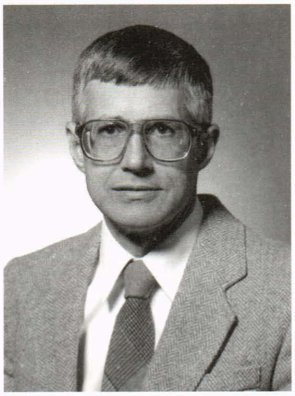
adopted, invented, and applied a variety of image processing techniques to advance DSMAC's potential and realized capabilities. Many image processing techniques and empirical relationships are applied in selecting scenes, mapping reconnaissance information showing the scenes, and forecasting DSMAC performance with the resulting maps. In flight, DSMAC relies on the robustness of correlation in combination with inertial data to determine position in real time.

Much refinement of DSMAC has been achieved since its initial development. Even so, evolving technology and the complexity of optical features provide incentive for more improvements. More detail in DSMAC planning calculations, for example, may improve forecasts made for complex scenes and environments. In addition, other algorithms to run in the reprogrammable DSMAC Block IIA flight processor have been proposed.

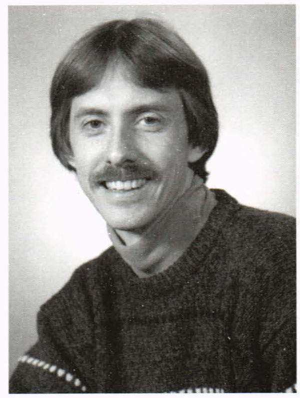
REFERENCES

[1] Mostafavi, H., and Smith, F. W., "Image Correlation with Geometric Distortion Part I: Acquisition Performance," *IEEE Trans. Aerospace Electron. Sys.* **AES-14**(3), 487–493 (1978).
[2] Criss, T. B., *Calculation of Optimum Value of the Phasing Parameter for the New Scene Evaluation Algorithm*, JHU/APL F1A(3)-86U-11 (1986).
[3] Ratway, V., *Baseline DSMAC IIA Horizontal Geometry Error Model*, N40MP, Naval Surface Warfare Center, Dalghren Division (1992).

THE AUTHORS



GEOFFREY B. IRANI, a member of APL's Principal Professional Staff, holds a B.S. degree in physics from the University of Maryland and M.S. and Ph.D. degrees in engineering/applied science from the University of California. During graduate studies at the Lawrence Livermore Laboratory, he specialized in low-pressure shock waves, the optical properties of solids, and ultra-high vacuum reflectometer design. After joining APL in September 1970, he assessed detection of ballistic missile submarines, specializing in ocean surface and internal waves and optical remote sensing of the ocean surface. In 1982, he shifted to optical imaging for the Digital Scene Matching Area Correlator, becoming lead DSMAC engineer in 1984. In 1985, he conceived the correlation addition processing that DSMAC Block IIA now uses. Dr. Irani was granted a U.S. patent and received the American Society of Naval Engineers' Solberg Award for this invention.



JAMES P. CHRIST received a B.S. degree in electrical engineering and a B.S. degree in mathematics in 1980, an M.S. degree in electrical engineering in 1982, and a Ph.D. in electrical engineering in 1987 from Carnegie Mellon University. While in school, he worked on tactile sensing in robotics. Since joining APL in 1987, he has worked on the Digital Scene Matching Area Correlator navigation system for the Tomahawk cruise missile. He is a member of the APL Senior Professional Staff.