JOURNAL OF AIRCRAFT
Vol. 36, No. 4, July–August 1999

Automatic Target Recognition Directed Image Compression

Charles D. Creusere* and Alan Van Nevel†
U.S. Naval Air Warfare Center, China Lake, California 93555

A novel synthesis is presented of two separate areas of image processing: automatic target recognition/cueing (ATR/ATC) and embedded image compression. To maximize the information content of a transmitted image, an ATR algorithm is used to detect potential areas of interest, and the compression algorithm regionally compresses the image, allocating more bits to the areas of interest. In this fashion, contextual information is retained, albeit at a lower resolution. Higher information content is achieved at a lower bit rate.

I. Introduction and Background

AS the number of operational uncrewed aerial vehicles (UAVs) and uncrewed aerial combat vehicles (UCAVs) increases, the availability of sufficient communications bandwidth will become a major concern. Whereas command and control signals require a small portion of the available bandwidth, high-resolution imagery and video can easily consume all of the available bandwidth and more. Unfortunately, tactical datalinks such as link 16 or SATCOM only support data rates of 57 kbit/s (with error correction) and 4.8 kbit/s, respectively, whereas other high-speed systems such as common datalink have significant limitations on the number of simultaneous users allowed. To increase the number of systems that can operate simultaneously using the limited rf bandwidth available, one must adopt digital compression. However, compression results in degraded imagery at the receiver for reasonable compression ratios (>2:1). With these severe limitations on the bandwidth in mind, we have developed a hybrid algorithm that draws on automatic target recognition/automatic target cueing (ATR/ATC) algorithms and regional embedded compression techniques.

The basic idea of the algorithm is relatively simple. Whereas an image may contain several different objects (trucks, buildings, tanks, etc.), only a few may be of potential interest to a military observer. An ATR algorithm is used to select potential areas of interest, which are then fed to a compression algorithm that regionally compresses the image, allocating more bits to the areas of interest. This process results in an image that has areas of high resolution (potential targets) and areas of low resolution in which contextual information is retained. Overall, a high-compression ratio can be achieved while maximizing information content.

UAVs are a platform for which this technology could be extremely useful. By placing an ATR/ATC directed compression system on a UAV, the reduction in bandwidth usage would enable more aircraft to effectively communicate over a limited channel. Furthermore, the ATR/ATC would also reduce the workload of UAV operators and imagery analysts, by providing areas of potential interest automatically. In a different context, the regional compression algorithms could be used to upload data to aircraft in flight, preserving features of interest in the imagery at high resolution, while still retaining contextual information, albeit at a lower resolution. A possible application in this sense would be for uploading imagery that provides funnel features for navigation. Currently within the military, there is a desire for imagery in the cockpit, and this goal can be achieved for preplanned missions. However, the communication channels available to tactical aircraft are ill suited for uploading imagery to an aircraft in flight. ATR directed regional compression can alleviate this problem.

In the first section, we will describe the ATR algorithm, developed by Carlson,[1] Mahalanobis et al.,[2,3] and Vijaya Kumar et al.,[4] which we use to determine the regions of interest. In the following sections, we will describe the embedded zerotree wavelet (EZW) algorithm and how it has been adapted for spatially variant resolution. In the last sections we present some results and discuss potential applications and future directions.

II. Maximum Average Correlation Height Filters

A. Mathematical Overview

To maximize the information content sent over the available bandwidth, we have chosen to identify potential regions of interest. To accomplish this, we have implemented a class of correlation filters as developed by Mahalanobis et al.[2,3] These filters have exceptional tolerance to scaling and rotation distortions. The tolerance of the filters is incorporated through the selection of an appropriate training set and can be tuned to provide high (generalization) or low (specificity) tolerance.

In the discussion of the maximum average correlation height (MACH) filters that follows, a bold lower case symbol indicates a column vector, whereas a bold upper case symbol represents a diagonal matrix. The filters result from maximizing the ratio

$$J(\boldsymbol{h}) = |\boldsymbol{h}^{+}\boldsymbol{m}|/\boldsymbol{h}^{+}\boldsymbol{S}\boldsymbol{h} \tag{1}$$

where $\boldsymbol{h}$ is the correlation filter and $\boldsymbol{m}$ is the average of the training images in the Fourier domain. Each image is lexigraphically ordered to form a vector. $\boldsymbol{S}$ is the average similarity measure matrix

$$\boldsymbol{S} = \sum_{k=1}^{N} (\boldsymbol{X}_k - \boldsymbol{M})(\boldsymbol{X}_k - \boldsymbol{M})^{+} \tag{2}$$

In Eq. (2) $\boldsymbol{X}_k$ are the individual training images, again in the Fourier domain. The training image is lexigraphically ordered, and its elements placed on the diagonal of $\boldsymbol{X}_k$, whereas $\boldsymbol{M}$ is the mean training image, arranged similarly to $\boldsymbol{X}_k$. Furthermore, all of the processing to generate the filters is performed in the Fourier domain to gain translational invariance. It is possible to perform the processing in other domains, for example, wavelet or spatial, but care must be taken to properly register the training imagery.

The optimal filter $\boldsymbol{h}$ is then given by

$$\boldsymbol{h} = \boldsymbol{S}^{-1}\boldsymbol{m} \tag{3}$$

Variants on the MACH filter can be achieved by varying the performance metric one wishes to maximize. For example, Refregier[5] has developed optimal tradeoff synthetic discriminant filters (OTSDFs) that attempt to minimize the energy functional

$$E(\boldsymbol{h}) = \boldsymbol{h}^{+}\boldsymbol{Q}\boldsymbol{h} - \delta|\boldsymbol{h}^{+}\boldsymbol{m}| \tag{4}$$

Presented at the AIAA Missile Sciences Conference, Monterey, CA, 17–19 November 1998; received 8 February 1999; accepted for publication 10 April 1999. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States. Approved for public release, distribution unlimited.

*Electrical Engineer, Computational Sciences Branch, Code 471600D; creuserecd@navair.navy.mil.

†Research Physicist, Signal Processing Technology Branch, Code 471600D; vannevelaj@navair.navy.mil.

where

$$\boldsymbol{Q} = \alpha \boldsymbol{P} + \beta \boldsymbol{D} + \gamma \boldsymbol{S} \tag{5}$$

$\boldsymbol{S}$ is as defined previously, $\boldsymbol{P}$ is the power spectral density of the expected noise, and $\boldsymbol{D}$ is the average power spectral density of the training set. The constants α, β, γ, and δ are nonnegative and must satisfy $\alpha^2 + \beta^2 + \gamma^2 + \delta^2 = k$, where k is any positive constant. Minimizing $E(\boldsymbol{h})$ results in

$$\boldsymbol{h} = (\delta/2)\boldsymbol{Q}^{-1}\boldsymbol{m} \tag{6}$$

By varying the parameters, one can optimize filter performance for the situation under study. If one sets $\alpha = \beta = 0$, the result is the MACH filter discussed earlier. Further variations can be made to the basic idea, including the extension to multiple class discrimination using distance classifier correlation filters, which are able to distinguish between multiple classes of similar objects, for example, T72s vs M1A1 tanks.

The class of MACH filters was chosen for the feature detection for several reasons. As discussed, the filters can incorporate varying degrees of distortion tolerance and can be built to generalize classes of targets. Another benefit of the algorithm is that the result is statistically optimum and depends on a realistic, mathematically rigorous optimaztion procedure as opposed to other heuristic methods. A final consideration is the computational efficiency. The MACH filters require no segmentation or edge detection preprocessing, and the correlation step can be performed rapidly using dedicated fast Fourier transform hardware.

B. MACH Implementation

To implement the MACH filters, one must first decide on a representative training set. Typically, the training set consists of $N < 20$ images collected from varying perspectives. A training set of one image will result in a filter similar to the matched filter with no distortion tolerance, whereas a training set having dozens of perspectives and scalings will produce a filter with a broad response and low discrimination properties. The filter $\boldsymbol{h}$ is first calculated offline from the training data. If one is using the OTSDFs, some parameter tuning can be performed at this point to maximize the correlation peaks for the training data.

Following correlation of an input test scene with $\boldsymbol{h}$, the correlation scene must be processed to determine the areas of interest. Previous correlation filters[6–8] had placed constraints on the correlation height, and classification was then accomplished by comparing the correlation height of the test scenes to the constraint. Generally, a threshold must be set when using the correlation height as a metric for detection and/or classification. By changing this threshold one can trade off between the probability of detection and the probability of false alarms, a lower threshold allowing more false alarms and a higher threshold reducing the probability of detection.

A second metric that can be used is the peak-to-sidelobe ratio (PSR). A square window (1×1, 3×3, 5×5, ...) is chosen and centered around the correlation peak, and the energy within this window is calculated. The energy in a larger square window is also calculated, and the ratio of the two energies is the PSR. This ratio can then be compared to a threshold for detection and classification. The local energy percentage can be modified to allow for multiple target possibilities by selecting multiple windows based on correlation height and excluding these energies from the global energy calculation. The benefit of this approach is that the ratio is independent of illumination or amplification effects. The overall peak height can be affected by constant amplification, but the ratio will remove this problem. This metric works well in rejecting false peaks due to clutter because most correlation surfaces for clutter images will not contain a high percentage of energy in a localized window.

In our hybrid algorithm, the second metric was chosen to determine regions of interest. No hard thresholding was used for detection. Instead, the top three energy percentage locations were selected as potential regions of interest to be compressed at a higher resolution than the background. The choice of three targets is somewhat arbitrary and can be changed based on the application. If a large number of areas are desired at high resolution, it may impact how the coding of the side information is performed. The choice may also be eliminated completely with the use of thresholding to eliminate false alarms and to increase the probability of detection. In this demonstration it was sufficient to designate a number of potential targets to effectively illustrate the concept.

III. EZW Compression

At the core of our feature-based approach to compression is an embedded coding algorithm. In this method of compression, data are transmitted to the receiver in order of importance, that is, the data that most reduces the error between the reconstructed image and the original image is sent first. This concept is shown in Fig. 1. There are a number of advantages to embedded compression algorithms: fixed bit rates, for example, compression ratios, are easily achieved; unequal transmission error protection is trivial; and inherently robust bit streams can be created.

The fundamental observation that inspired the EZW algorithm[9] is that there is a strong correlation between insignificant coefficients at the same spatial locations in different wavelet scales; that is, if a wavelet coefficient at a coarser scale is zero, then it is more likely that the corresponding wavelet coefficients at finer scales will also be zero. Figure 2 shows a three-level, two-dimensional wavelet decomposition and the links that define a single zerotree (the quadtree data structure containing all of the coefficients corresponding to a given region of the original image). If a wavelet coefficient at a given scale is zero along with all of its descendants (as shown in Fig. 2), then a special zerotree-root (ZTR) symbol is transmitted, eliminating the need to transmit the values of the descendants. Note that ZTR symbols can be created at any level of the wavelet coefficient mapping. Thus, if one of the three zerotree children at level s2 in Fig. 2 is significant for a given bit plane while the others are not, then the

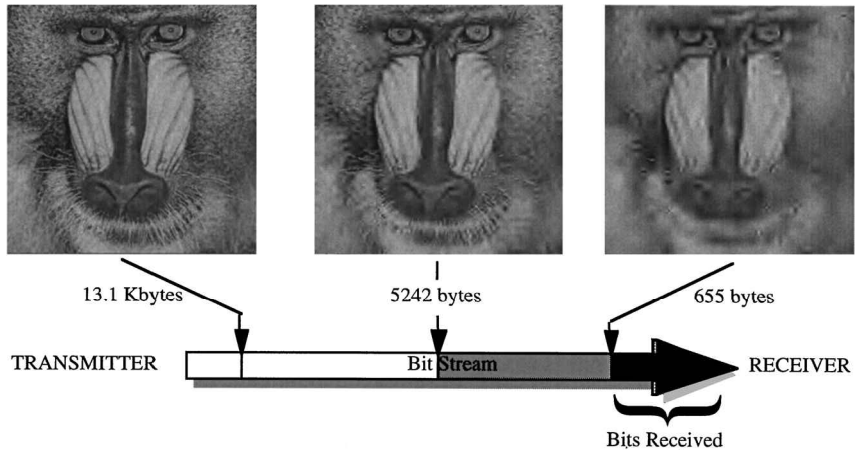


Fig. 1 Embedded image compression.

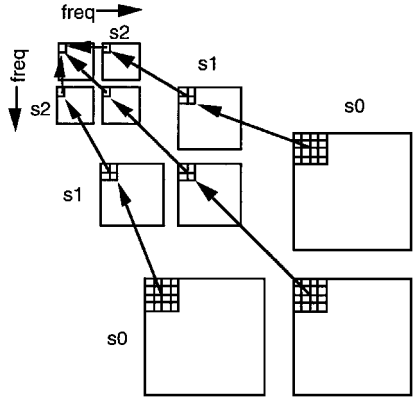


Fig. 2 Wavelet coefficient mapping with one complete zerotree shown.

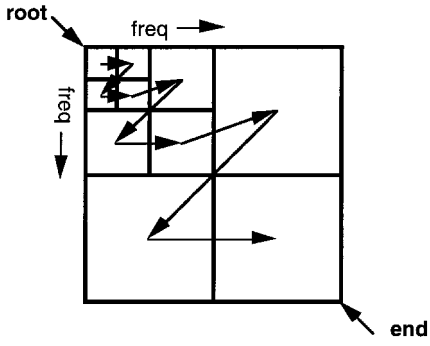


Fig. 3 Scanning order of wavelet coefficients.

low-frequency root coefficient must be represented by an isolated zero (IZ) symbol (assuming that it is also insignificant) while the two insignificant children are represented with ZTR symbols. Note that the wavelet scale (s0, s1, etc.) is inversely proportional to the spatial frequency. Basically, one can view this redundancy extraction process as a multiresolutional, self-terminating run-length encoder. Regardless of how one views it, zerotrees exploit the correlation of insignificance across scales and decrease the number of symbols that must be processed by the arithmetic encoder and transmitted to the receiver.

To generate an embedded code (where information is transmitted in order of importance), the algorithm scans the wavelet coefficients in what is basically a bit-plane fashion. First, a starting threshold is selected that is at least one-half as large as the magnitude of the largest wavelet coefficient. If the starting threshold is selected to be a power of 2, then a very fast approach can be used to compute all of the zerotree dependencies in one pass through the wavelet coefficients.[10] Starting with the appropriate threshold, the algorithm sweeps through the coefficients from low- to high-frequency subband as shown in Fig. 3, transmitting the sign (+ or −) if a coefficient's magnitude is greater than the threshold (i.e., it is significant), either a ZTR, if it is less than the threshold and the root of a zerotree at the coarsest possible scale, or an IZ, otherwise; this is the dominant pass. Next for the subordinate pass, all coefficients deemed significant in the dominant pass are added to a second subordinate list that is itself scanned. For each coefficient on this list during the pass, 1 bit is transmitted, decreasing its approximation error in the decoder by one-half (the coefficient's absolute error during a given pass depends on the value of the starting threshold). One iteration of this successive refinement process is shown in Fig. 4. The threshold is then halved, and the two passes are repeated with those coefficients having been previously found significant being replaced by zeros in the dominant pass (so that they do not inhibit the formation of future zerotrees). The symbol stream created by this scanning process is then passed through an arithmetic encoder to eliminate any remaining statistical redundancy before transmission to the decoder. A block diagram of the complete process is shown in

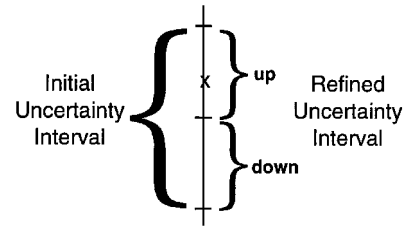
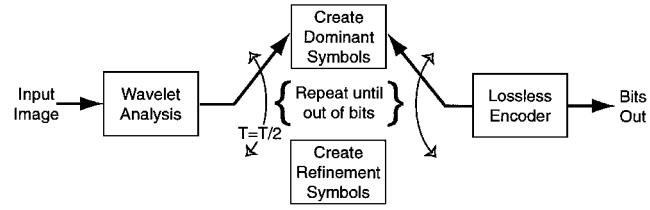
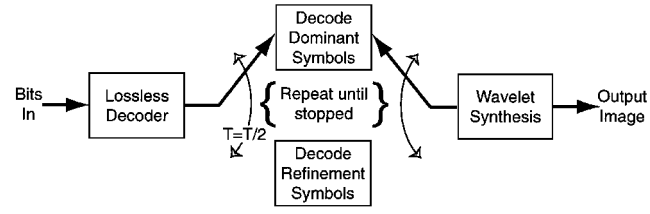


Fig. 4 One iteration of the successive refinement process.



a) Encoder



b) Decoder

Fig. 5 Embedded image.

Fig. 5a. To estimate symbol probabilities for the arithmetic encoder and decoder, we use the simple single-context, backward-adaptive model presented by Witten et al.[11] Slightly better compression can be achieved using the multicontext model proposed by Shapiro[9] at the cost of decreased execution speed. The routine of dominant pass, subordinate pass, and threshold reduction continues until the bit budget is exhausted or until some distortion criterion is reached; at that point, the encoder transmits a stop symbol and begins processing the next frame in the video sequence.

The image decoder simply inverts each operation performed by the encoder in reverse order: that is, it arithmetically decodes the bit stream to create symbols, and then it decodes the symbols to progressively refine its estimates of the wavelet coefficients. This process is illustrated by the block diagram shown in Fig. 5b. Because the arithmetic coding model is backward adaptive, we need not transmit it as side information. Furthermore, because the decoder's knowledge exactly mirrors that of the encoder at any given point in the processing of the bit stream, there is no need to transmit pass delimiters or synchronization signals (although these might be useful for resolution-scalable compression). Also, the resolution enhancement bits transmitted during the subordinate pass do not need any location specifiers; the decoder knows the exact transmission order of these bits because it has reconstructed the same subordinate list as the encoder had at that point in the processing.

IV. Feature-Based Compression

A. Feature-Selective Resolution Control

The conventional EZW algorithm allocates resolution uniformly across the image. To achieve this spatial uniformity, it actually distributes resolution to the wavelet coefficients nonuniformly across the wavelet scales, that is, frequency subbands. Specifically, a coarser scale is allocated twice as much resolution as the next finer scale; this allocation is implicitly controlled by the use of a unitary or unitarylike scaling in the wavelet decomposition. Such coefficient scaling has the effect of increasing the gain in each successive level of the two-dimensional wavelet decomposition by a factor of 2. To understand how a multiplicative factor implicitly controls resolution, consider the following example: Assume that the true value of a

wavelet coefficient is 85 and that the final dominant pass through the coefficients ends with threshold $T = 64$. Without rescaling, the final uncertainty interval for this coefficient in the decoder will be [64, 128), resulting in a reconstructed coefficient value of 96 ± 32. Now, assume instead that this coefficient is multiplied by 2 prior to coding; that is, we code the value 170. During the pass when $T = 128$, the coefficient will be declared significant and approximated in the decoder by 192 ± 64. After the refinement pass, however, the new approximation will be 160 ± 32. Because, the encoder stops after the dominant pass for $T = 64$, the coefficient will receive no further refinement bits. Dividing the coefficient approximation by 2 restores its original scaling and results in the final estimate of 80 ± 16. Thus, the uncertainty region of the new estimate, [64, 96), is one-half that of the original.

Whereas scaling is used in the classical EZW algorithm to implicitly control the bit allocations to coefficients in the different wavelet scales, it can also be used explicitly to weight features in the imagery. By our definition, a feature is anything in the imagery that can be localized in space and/or frequency. For example, a tank in a reconnaissance photo might be a spatial feature to which we want to allocate additional resolution. On the other hand, an orchard in the same photo could represent a space–frequency feature, that is, a feature that is defined by a frequency spectrum within a spatial region, whose resolution allocation should be reduced to increase the resolution of other more interesting portions of the image. Either way, the allocation of resolution (and thus bits) is most easily controlled by scaling coefficients up or down by the appropriate powers of 2 prior to encoding. Note, however, that rescaling only adjusts the resolution of coefficients relative to other coefficients: This process is a zero sum game. Thus, scaling all of the wavelet coefficients up by a factor of 2 will have no effect on the resolution of the reconstructed image.

B. Coding Scheme

By using coefficient rescaling, we create a single embedded bit stream in which different wavelet coefficients are represented with varying precision. If all of the coefficients corresponding to a single zerotree are coded at a specified precision, then the corresponding region of the reconstructed image will be reconstructed at the same precision (assuming that orthonormal wavelets are used). The "×"s in Fig. 6 correspond to the coefficients that form a complete zerotree, and it is these coefficients that must be multiplied by a power of 2 scaling factor to increase the resolution of the corresponding 16×16 image region. The size of the region in the image that corresponds to 1 zerotree (the minimally controllable region) for a depth L wavelet decomposition is $2^L \times 2^L$. Thus, to spatially vary the resolution across the image, one need only vary the coefficient scaling

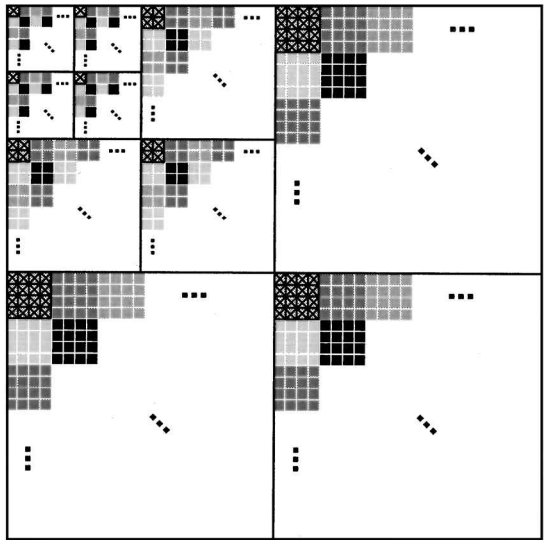


Fig. 6 All coefficients shaded in gray are rescaled together.

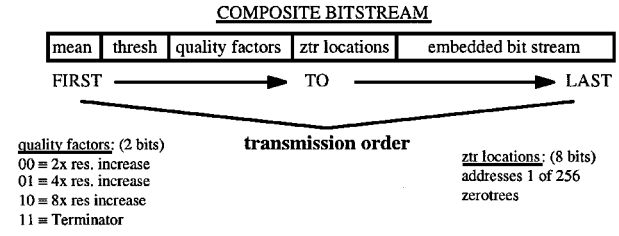


Fig. 7 Organization of compressed bit stream.

on a zerotree by zerotree basis. If, on the other hand, one wishes to increase or decrease the resolution of specific frequency bands within a given region, one must vary the scaling factor between wavelet scales corresponding to the same zerotree. Note that by using a fixed wavelet decomposition, we have limited the amount of frequency control that can be achieved, that is, the bandwidth of each image subband increases by a factor of 2 in each dimension as the wavelet scale decreases. If instead one used an adaptive wavelet packet decomposition,[12,13] one could also optimize the bandwidth of the subbands for the space–frequency features of interest. Because such wavelet packets greatly increase the complexity of the encoder, however, we have chosen not to use them in our system.

For the decoder to correctly reconstruct the wavelet coefficients, it must know which areas have been rescaled and by what scaling factor. The only exception to this general rule is when a coefficient has been scaled down so much that it will be reconstructed as 0 by the decoder. In the case where small areas of enhanced resolution are desired, the side information describing the rescaling to the decoder is very compact and does not require lossless compression. Figure 7 shows the organization of the compressed bit stream. First, the image mean (which was subtracted from the image prior to the wavelet decomposition) and starting threshold are sent; this is always done for any embedded compression algorithm. Second, a set of quality factors are transmitted, one for each zerotree whose resolution has been increased. In the current instantiation, each quality factor is a 2-bit quantity and allows for three levels of resolution increase (×2, ×4, and ×8). Because the number of zerotrees with resolution increases is not predetermined, a fourth parsing symbol is also allowed which terminates the quality factors section and tells the decoder how many ZTR locations to expect. Each ZTR location is a numeric value that uniquely indexes one zerotree; if the image is of size 512×512 and a depth 5 decomposition is used, then 1 B is sufficient to uniquely identify a zerotree. One zerotree location is transmitted for each quality factor, and the ordering is such that each zerotree location index corresponds exactly to a previously transmitted quality factor. In addition, no parsing symbol is needed because the decoder already knows how many zerotree location indices to expect. Finally, the embedded bit stream containing the compressed representation of the actual image is transmitted to the decoder.

One might note that the proposed method of transmitting the side information will not be particularly efficient if a large number of zerotrees are rescaled. In such a situation, it might be more efficient to simply send 1 bit for each zerotree indicating whether or not its scaling has been altered along with an ordered list containing the magnitudes of the rescaling. In the case where there are 256 total zerotrees, transmitting the ZTR location specifiers in this way would require exactly 32 B. Thus, the tradeoff is clear: If fewer than 32 zerotrees are rescaled, an explicit index to each should be transmitted; otherwise, the 1-bit/zerotree rescaling map should be sent.

V. Results

For the results presented in this section, we use a five-level decomposition based on a 5/3 biorthogonal wavelet transform [called (2, 2) in Ref. 14]. To increase its speed, we use lifting to implement this wavelet, which allows the high- and low-pass filters to share computations.[15] In addition, lifted transforms are in place and, thus, do not require large amounts of scratch memory during computation. Because a five-level decomposition has been selected, the

minimally controllable spatial region is 32×32 (not taking into account the overlapping basis functions of the transform). From a sequence of 800 frames, a training set of 10 images containing the group of 4 buildings indicated by the white arrow in Fig. 8a was selected. Note that in Fig. 8a squares have been added to highlight enhanced regions and that the arrow indicates the particular region of interest for which the system was trained.

From this training set, a MACH filter was constructed to recognize the buildings as the feature of interest. The size of the filter in this case was 64×64, which is larger than the minimally controllable region as determined by the depth of the wavelet transform. The regions with the top three correlation peaks, that is, the best three matches, are allocated resolution according to the following formula: region 1 always receives eight times the resolution of the background; regions 2 and 3 receive the same if their correlation peaks are within 5% of region 1, but they receive four times the resolution if the peaks are within 20%, and twice the resolution if less than 20%. The reasoning behind such an allocation scheme is that, although the MACH filter ideally will always return the highest correlation peaks for the true target, this is not always the case. If a false alarm has the highest peak value, presumably the true target has a similar value and, hence, correlation peaks within 5% of the top peak receive the same resolution. If the correlation scores are significantly different, the probability that the lower scores are true targets decreases and the region receives a lower bit allocation. Figures 8 and 9 show the results for compression at ratios of 80:1 and 160:1, respectively. Again, squares have been added to highlight enhanced regions. By looking at the error residuals (Figs. 8b and 9b), one can see the differences in the allocation of resolution to the different regions. In both examples, the area containing the group of four buildings has the highest resolution whereas the other two highlighted regions have lower resolution (but still higher than the background). As already mentioned, the choice of three high-quality regions is somewhat arbitrary but it does depend some on the design of the MACH filter. A tradeoff between the amount of compression and the probability of detection and false alarms needs to be considered when actually implementing this algorithm in practice.

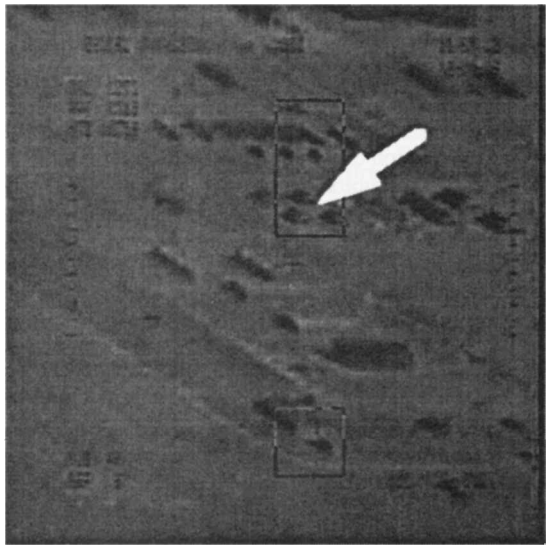


Fig. 8a Reconstruction of image compressed by 80:1 ratio.

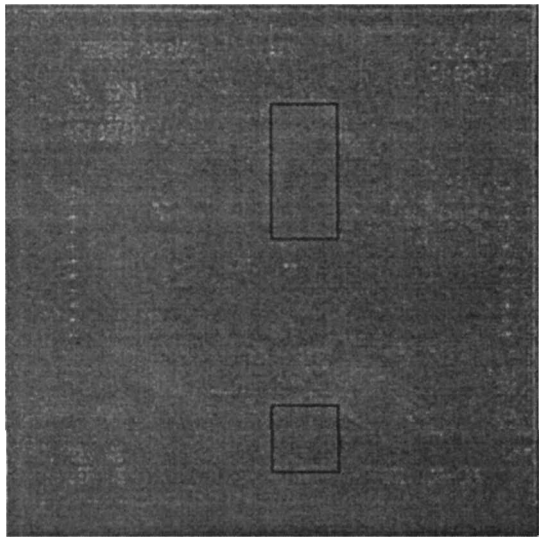


Fig. 8b Error residual between reconstructed and original image where white areas in residual denote large errors.

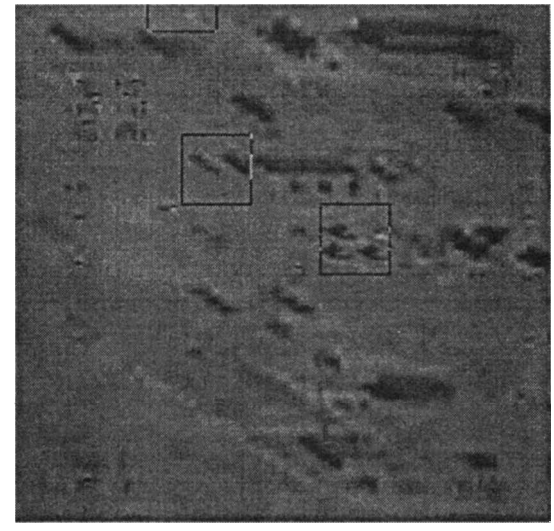


Fig. 9a Reconstruction of image compressed by 160:1 ratio.

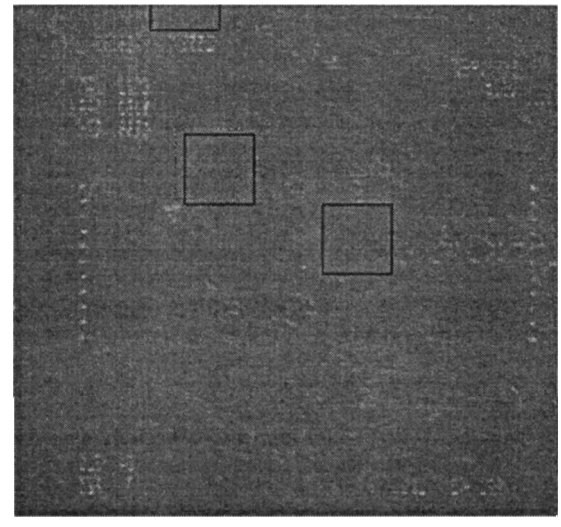


Fig. 9b Error residual between reconstructed and original image.

VI. Discussion and Conclusions

We have described a synthesis of two disparate areas of image processing: compression and ATR/ATC. The regional compression algorithm expands the effective bandwidth available by maximizing the information content of the transmitted imagery, using information from the ATR algorithm. One obvious area where this technique may prove valuable is in the UAV/UCAV arena. For example, with the use of the ATR/ATC driven compression, analysis by a human operator is simplified by allowing rapid identification of potential areas of interest. A second area where this work could be extended to is image database management. By utilizing regional compression driven by an ATR/ATC algorithm it is possible to achieve useful

compression ratios (>80:1) while preserving image quality in areas that may be of interest. The potential savings in communications bandwidth could enable more vehicles to operate simultaneously than would be possible with standard compression techniques currently in use.

References

[1]Carlson, D., "Optimal Tradeoff Composite Correlation Filters," Ph.D. Dissertation, Dept. of Electrical and Computer Engineering, Carnegie–Mellon Univ., Pittsburgh, PA, Oct. 1996.

[2]Mahalanobis, A., Vijaya Kumar, B. V. K., Song, S., Sims, S. R. F., and Epperson, J., "Unconstrained Correlation Filters," *Applied Optics*, Vol. 33, No. 17, 1994, pp. 3751–3759.

[3]Mahalanobis, A., Vijaya Kumar, B. V. K., and Sims, S. R. F., "Distance Classifier Correlation Filters for Multiclass Target Recognition," *Applied Optics*, Vol. 35, No. 17, 1996, pp. 3127–3133.

[4]Vijaya Kumar, B. V. K., Carlson, D., and Mahalanobis, A., "Optimal Tradeoff Synthetic Discriminant Function (OTSDF) Filters for Arbitrary Devices," *Optics Letters*, Vol. 19, No. 19, 1994, pp. 1556–1558.

[5]Refregier, P., "Optimal Tradeoff Filters for Noise Robustness, Sharpness of the Correlation Peak and Horner Efficiency," *Optics Letters*, Vol. 16, No. 8, 1991, pp. 829–831.

[6]Bahri, Z., and Vijaya Kumar, B. V. K., "Generalized Synthetic Discriminant Functions," *Journal of the Optical Society of America. A*, Vol. 5, No. 4, 1988, pp. 562–571.

[7]Vijaya Kumar, B. V. K., "Minimum Variance Synthetic Discriminant Functions," *Journal of the Optical Society of America. A*, Vol. 3, No. 10, 1986, pp. 1574–1584.

[8]Mahalanobis, A., Vijaya Kumar, B. V. K., and Casasent, D., "Minimum Average Correlation Energy Filters," *Applied Optics*, Vol. 26, No. 17, 1987, pp. 3633–3640.

[9]Shapiro, J. M., "Embedded Image Coding Using Zerotrees of Wavelet Coefficients," *IEEE Transactions on Signal Processing*, Vol. 41, No. 12, 1993, pp. 3445–3462.

[10]Shapiro, J. M., "A Fast Technique for Identifying Zeros in the EZW Algorithm," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, IEEE Publications, Piscataway, NJ, 1996, pp. 1455–1458.

[11]Witten, I. H., Neal, R. M., and Cleary, J. G., "Arithmetic Coding for Data Compression," *Communications of the ACM*, Vol. 30, No. 6, 1987, pp. 520–540.

[12]Coifman, R. R., and Wickerhauser, M. V., "Entropy-Based Algorithm for Best Basis Selection," *IEEE Transactions on Information Theory*, Vol. 38, No. 2, 1992, pp. 713–718.

[13]Ramachandran, K., and Vetterli, M., "Best Wavelet Packet Bases in a Rate-Distortion Sense," *IEEE Transactions on Image Processing*, Vol. 2, No. 2, 1993, pp. 160–175.

[14]Daubechies, I., *Ten Lectures on Wavelets*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992, Chap. 8.

[15]Sweldens, W., "The Lifting Scheme: A New Philosophy in Biorthogonal Wavelet Constructions," *Wavelet Applications in Signal Processing, III*, Society of Photo-Optical Instrumentation Engineers, Vol. 2569, 1995, pp. 68–79.