Estimating performance limits for automatic target recognition in compressed video

Ryan A. Kerekes[*a], B. V. K. Vijaya Kumar[a], S. Richard F. Sims[b]
[a]Dept. of Electrical and Computer Engineering, Carnegie Mellon Univ., 5000 Forbes Ave.,
Pittsburgh, PA USA 15213
[b]U.S. Army RD&E Command, Aviation & Missile RDEC, Redstone Arsenal, AL 35898

ABSTRACT

Robust real-time recognition of multiple targets with varying pose requires heavy computational loads, which are often too demanding to be performed online at the sensor location. Thus an important problem is the performance of ATR algorithms on highly-compressed video sequences transmitted to a remote facility. We investigate the effects of H.264 video compression on correlation-based recognition algorithms. Our primary test bed is a collection of fifty video sequences consisting of long-wave infrared (LWIR) and mid-wave infrared (MWIR) imagery of ground targets. The targets are viewed from an aerial vehicle approaching the target, which introduces large amounts of scale distortion across a single sequence. Each sequence is stored at seven different levels of compression, including the uncompressed version. We employ two different types of correlation filters to perform frame-by-frame target recognition: optimal tradeoff synthetic discriminant function (OTSDF) filters and a new scale-tolerant filter called fractional power Mellin radial harmonic (FPMRH) filters. In addition, we apply the Fisher metric to compressed target images to evaluate target class separability and to estimate recognition performance as a function of video compression rate. Targets are centered and cropped according to ground truth data prior to separability analysis. We compare our separability estimates with the actual recognition rates achieved by the best correlation filter for each sequence. Numerical results are provided for several target recognition examples.

Keywords: automatic target recognition, video compression, correlation filters, scale distortion

1. INTRODUCTION

Data compression is becoming an increasingly important part of defense-oriented sensor networks as the number of sensors and the volume of acquired data grow faster than the bandwidth of the links that support them. Typically the groups of sensors generate much more data than can be processed by humans, making robust automatic processing and analysis of this data critically important. In many applications the computational loads incurred by real-time processing algorithms are prohibitively heavy for on-site data processing, and therefore data acquired by the sensors is transmitted to a remote facility to be processed. In the case of imaging sensors, such as electro-optic (EO) sensors and synthetic aperture radar (SAR), the use of video compression algorithms can greatly increase the amount of data that can be transmitted across a given link, permitting for example higher resolution imagery, higher frame rates, or more sensors sharing the link.

International video compression standards (e.g. MPEG, H.26x) are preferred in many applications over proprietary solutions. The MPEG-2 standard has been particularly important in industry as the established format for DVD video and digital television. In 2003, a joint effort between the two leading standards organizations, International Telecommunications Union (ITU) and International Standards Organization (ISO), produced a standard called ITU-T/H.264 | ISO/IEC MPEG-4 AVC, which offers significant improvement in video compression efficiency over MPEG-2[1]. This new standard is referred to as H.264.

In this paper we investigate the effects of H.264 video compression on automatic target recognition (ATR), which refers to the task of locating and identifying military targets in an image or a sequence of images. Our primary focus is the use

[*] rkerekes@andrew.cmu.edu

of correlation filters to perform robust target recognition on individual frames of several H.264-compressed/decompressed video sequences. The video sequences are part of a database of infrared (IR) video imagery acquired using a missile seeker camera onboard a helicopter. Each sequence is recorded as the helicopter flies an approach trajectory toward one or more ground targets. This type of flight path introduces large amounts of scale distortion in the target imagery. Seven decompressed versions of each sequence are stored, each one having a different compression quality.

Many approaches have been proposed for robust ATR on IR imagery. Popular methods include feature-based methods[2], neural networks[3], and model-based approaches[4], all of which require image segmentation. We focus on the use of correlation filters, which do not require any segmentation and have been shown to be effective for ATR[5,6]. We use two different types of correlation filters to perform target detection and recognition on the decompressed sequences: optimal tradeoff synthetic discriminant function (OTSDF) filters[7], which handle scale distortions by training on several target images at different scales, and fractional power Mellin radial harmonic (FPMRH) filters[8], which have built-in scale tolerance and require only a single training image. We compare the recognition performance of these filters across the range of compression levels provided in our database and show the performance degradation as a function of compression ratio. Since we typically use banks of several correlation filters to perform target recognition, we also consider the performance effects of increasing the number of filters per bank.

A secondary focus of this paper is to evaluate the effects of video compression on target imagery without explicitly performing recognition. We desire to quantify the loss of target information in a manner independent of the choice of recognition algorithm and predict the resultant degradation in performance. Previous work in this area has involved the use of information metrics, such as mutual information and Kullback-Leibler distance[9,10], to measure clutter and target distortion. We propose the use of Fisher ratios to quantify compression effects by measuring inter-class separability as a function of compression ratio. We show several results from applying Fisher ratios to groups of target images, and we compare these results to the actual performance of the correlation filter classifiers across the range of compression levels.

The rest of this paper is organized as follows. Section 2 reviews the correlation filtering process, including several concepts specific to OTSDF and FPMRH filters. Section 3 describes the IR video database and the procedure used to perform recognition simulations on the database. Section 4 introduces the idea of using Fisher ratios for performance estimation, and Section 5 shows the results of such as well as actual performance results. We summarize our conclusions in Section 6.

2. CORRELATION FILTER REVIEW

In this section we first briefly review some basic concepts of correlation filtering. We then review some important properties of OTSDF and FPMRH filters, the two correlation filter designs used in our target recognition experiments.

2.1 Review of correlation filter basics

Correlation filtering refers to the process of locating a specific pattern in an input image by computing its cross-correlation with some pre-computed template image. The correlation filter itself is simply a template that is designed to have high cross-correlation with the pattern of interest. The simplest and most well-known correlation filter is the matched filter, which, as the name suggests, is identical to the pattern of interest. The matched filter is ideal for detecting a pattern in the presence of additive white Gaussian noise (AWGN); however, it is typically unsuitable for pattern recognition in most real-world applications in which the noise cannot be accurately modeled as AWGN. Advanced correlation filter designs such as OTSDF filters and maximum average correlation height (MACH) filters optimize certain performance criteria and have demonstrated excellent performance in many ATR applications[5].

It is important to differentiate between the "design" stage and the "use" stage of correlation filtering. The design stage refers to the process of computing the template image of the filter. This process is often computationally intensive due to various coordinate transformations, matrix inversions, training image segmentation, or other operations; however, the filter template need only be designed once. Using the filter requires much less computation. Correlation is carried out in the frequency domain by pointwise multiplication of the Fourier transform (FT) of the filter template by the FT of the input image and computing the inverse FT of the resulting product. This output is called the “correlation plane” and is subsequently inspected for peaks, which indicate locations of the pattern in the input image. Thus, no complicated matrix inversions or transformations are required during the use stage of correlation filtering.

2.2 OTSDF filters

The OTSDF filter design is a composite filter design that offers a user-controlled tradeoff between noise tolerance and peak sharpness. The user specifies the desired peak value for each of *N* training images as well as the value of the tradeoff parameter *α*. The equation for the OTSDF filter is as follows:

$$\mathbf{h} = \mathbf{T}^{-1}\mathbf{X}\left(\mathbf{X}^{+}\mathbf{T}^{-1}\mathbf{X}\right)^{-1}\mathbf{u} \tag{1}$$

where

$$\mathbf{T} = \left(\alpha\mathbf{D} + \sqrt{1-\alpha^{2}}\,\mathbf{C}\right). \tag{2}$$

In Eq. (1), **h** is the filter vector in the frequency domain, **X** is a matrix of frequency-domain training image column vectors, and **u** is an *N*x1 vector of the desired peak values for each of the *N* training images. In Eq. (2), **C** and **D** are diagonal matrices containing the power spectral density of the noise and the average power spectrum of the training images, respectively. The *α* weighting parameter balances the two terms of Eq. (2), effecting a tradeoff between output noise variance (ONV) and average correlation energy (ACE). It should be noted that these equations reduce to the minimum variance synthetic discriminant function (MVSDF) filter when *α*=0 and to the MACE filter when *α*=1.

2.3 MACE-MRH filters and FPMRH filtering

FPMRH correlation filtering is an extension of a recently introduced correlation filter design called MACE-MRH filters[8], which are formulated to tolerate target scale distortion. MACE-MRH filters are based on the Mellin radial harmonic (MRH) transform, an orthogonal basis set with the property that scaling a signal only affects its MRH coefficients by a phase factor. This scaling property is exploited in the MACE-MRH filter design theory to yield a filter with a user-controlled scale response; that is, the user can specify the desired correlation peak response curve of the filter over all input scale factors. A typical family of scale response curves is the set of rectangular functions, which cause a given filter to recognize the input pattern within some limited range of scale factors. A rectangular scale response curve is useful in designing banks of scale-tolerant filters, where each filter in the bank is responsible for recognizing targets in some small partition of the total range of scale distortion. MACE-MRH filters have higher discrimination capability than earlier scale-invariant filter designs[9] because they retain much more of the available pattern information.

The performance of MACE-MRH correlation filters can be improved by modifying the training and testing images that are used with the filters. FPMRH filtering refers to using MACE-MRH filters with fractional power-modified images, which is carried out as follows. First, we compute the DFT of the training or testing image to yield the frequency-domain array. We then raise the magnitude of each frequency domain coefficient to some power between 0 and 1 (using the same power for all the coefficients), while leaving the phase information unaltered. Finally, we compute the inverse DFT of this modified frequency array to yield the modified space-domain training or testing image. This type of modification has the effect of balancing the frequency content of the image and consequently the contribution of each spatial frequency to the overall correlation. FPMRH filtering produces much sharper and hence more detectable correlation peaks than regular MACE-MRH filters. We refer the reader to Ref. 10 for more details on MACE-MRH and FPMRH correlation filtering.

3. DATABASE AND EXPERIMENTAL SETUP

The first part of this section describes the database that was used to perform scale-tolerant pattern recognition with OTSDF and FPMRH correlation filters. The second part describes our approach to target recognition, including how the filters were trained and applied to the data and the metrics that were used to evaluate their performance.

3.1 Database

We used a database of infrared (IR) video imagery for performing target recognition using correlation filters. The database consists of approximately 45 long-wave infrared (LWIR) and 6 mid-wave infrared (MWIR) video sequences with a resolution of 128x128 pixels per frame. The video sequences are of length 250-300 frames on average at 30 frames per second (fps).

Each video sequence depicts one or more ground targets viewed from a missile seeker attached to a helicopter on an approach trajectory toward the targets, which introduces large amounts of scale distortion throughout the sequence. Targets in some sequences are in motion while others are stationary, and a few of the targets change their direction of movement during the sequence. Examples of the types of targets appearing in the sequences include Bradley tanks, M60 tanks, armored personnel carriers (APCs), and trucks. Target sizes typically range from 3-5 pixels (beginning of sequence) to 30-40 pixels (end of sequence) in diameter. Ground truth data accompanying the database provides detailed information about the targets in each frame, including the target name, *x*- and *y*-coordinates of the target, and the dimensions of a bounding box that fits tightly around the target pixels.

A key feature of the database is that seven different versions of each sequence are stored at increasing levels of video compression, including the original uncompressed version. The H.264 standard was used to compress and decompress the sequences with compression ratios as high as 65:1. Table 1 lists the labels used for each compression level in this paper and their corresponding compression ratios and data rates for a particular sequence from the database. Fig. 1 shows a sample frame from sequence "L1816" at three different levels of compression. The more highly compressed versions of the sequences provide a greater challenge for ATR algorithms because they contain significant amounts of noise and artifacts due to information loss.

Label	*Compression Ratio*	*Video Data Rate (kbps)*
Original	1:1	3932
q10	4:1	1075
q15	6:1	668
q20	13:1	310
q24	36:1	158
q26	50:1	115
q30	65:1	60

Table 1: List of several compression level labels and corresponding compression ratios and data rates for sequence "L1816." The letter "q" in the labels refers to the quantization step size used in compression; thus, higher step sizes result in greater compression and thus more information loss.

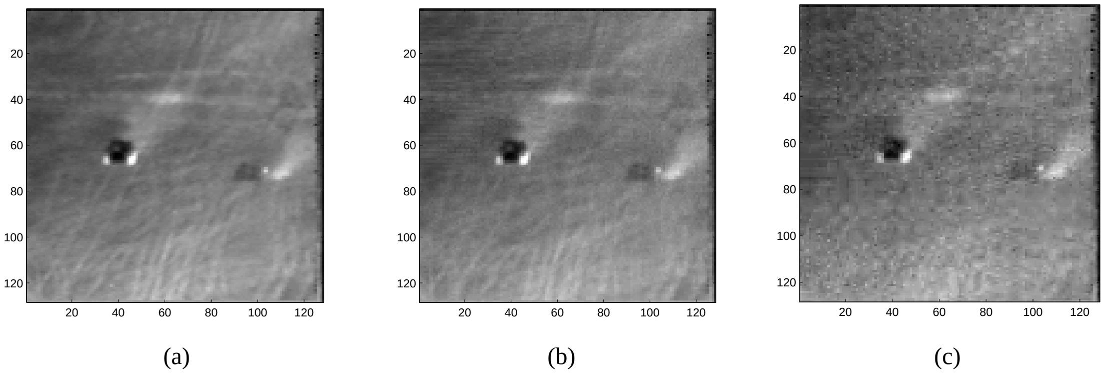


Figure 1: Frame 250 from sequence "L1816" at various compression levels: (a) no compression, (b) q20, and (c) q30.

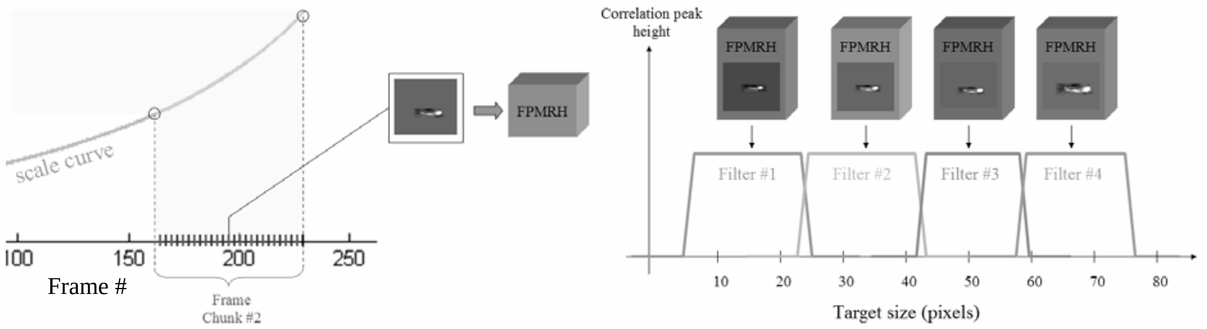
3.2 Recognition experiments

The primary goal in our design of each type of ATR classifier was to be able to detect all targets of a particular class across all frames of a given video sequence while rejecting clutter and false-class targets in that sequence. Thus, we trained each correlation filter classifier on images from a single sequence in order to recognize the targets in that sequence. Because of the high amount of target variability between sequences (out-of-plane rotation, thermal state, etc.), we only used a given classifier on the sequence from which it was trained. The main focus of our work was on specific target recognition rather than generic target detection; for this reason, each classifier was designed to recognize one particular target and reject all others.

Our recognition experiments employ two types of correlation filters for target recognition: optimal tradeoff synthetic discriminant function (OTSDF) filters and fractional power MACE-MRH (FPMRH) filters. Our approach to handling large amounts of scale distortion was to create correlation filter banks in which each filter is designed to handle a small portion of the overall range of distortion. This approach is similar to that used in a separate work to compare MACE-MRH filters and FPMRH filters[10]. In our implementation we used the odd-numbered frames of the uncompressed sequence for training data and the even-numbered frames for testing data. While this may be viewed as using an excessive number of training images, we want to make sure that any performance degradation we observe is due to the compression schemes rather than from using an insufficient number of training images. Each sequence was further partitioned into N disjoint "frame chunks" that represent sub-ranges of equal amounts of target scale distortion. The number N of frame chunks is a classifier design parameter that determines how many filters will be included in the filter bank. The beginning and end frames of each frame chunk for a given sequence were determined by using the target size information from the ground truth data to estimate the relative scale of the targets in each frame.

The training images were created by preprocessing the necessary frames of the sequence as follows. First, the target region was cropped from the frame according to the ground truth data. The target image chip was then placed against a 128x128 zero-background image and multiplied by a 2-D separable Tukey window to create a smooth transition from pattern to background. The image chip was processed prior to windowing so that the entire training image (containing the windowed image chip and zero background) was zero-mean.

The FPMRH filter design requires only one training image per filter; therefore, in order to generate FPMRH filter banks, we trained each of the N filters in the bank using the central frame (after preprocessing) of the corresponding frame chunk as illustrated in Fig. 2(a). We designed the scale response of each filter so that the union of the scale responses of all N filters spans the entire range of scale distortion in the sequence. This filter bank design scheme is illustrated in Fig. 2(b).



(a) (b)

Figure 2: Illustration of the training scheme for FPMRH filters in the ATR simulations. The illustration in (a) shows that each filter uses only one image from its corresponding frame chunk. The scale response curve of each filter is designed such that the set of FPMRH filters spans the entire range of scale distortion in the sequence.

To generate correlation filter banks using the OTSDF filter design, we train each filter on *all* the preprocessed frames from the corresponding frame chunk. The reason for this is that OTSDF filters have no inherent tolerance to geometric distortions; thus we must train these filters on a set of images that faithfully represents the expected distortion. This scheme is illustrated in Fig. 3. The OTSDF tradeoff parameter α can be adjusted to increase either output peak sharpness or noise tolerance; for these experiments we used $\alpha = 10^{-5}$, which results in filters close to the MACE filter but with increased noise tolerance and decreased peak sharpness. This value was determined empirically by comparing error rates generated by OTSDF filters with various values of α. A white noise model was used in the OTSDF filter design; we do not investigate the use of correlated noise models in this paper.

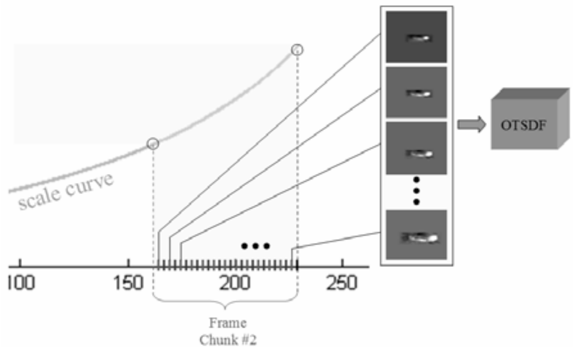


Figure 3: Illustration of the training scheme for OTSDF filters in the ATR simulations. Each OTSDF filter is trained on all of the images from the corresponding frame chunk in order to achieve some scale tolerance.

During the testing stage, a correlation filter bank was applied to the testing portion of the video sequence frame by frame; that is, no tracking algorithms were used to assist in locating the target. For each testing frame, all *N* filters in the bank were applied to the frame, resulting in *N* separate correlation planes per frame. Detection of peaks was carried out in the following manner. First, each correlation plane was searched for peaks by computing the peak-to-sidelobe ratio (PSR) of each peak within a 30x30 window and outside of a 6x6 exclusion window. PSR is a measure of the sharpness of the correlation peak; it is defined as the ratio of the correlation peak height with mean subtracted to the standard deviation of the correlation plane within some window around the peak. A smaller window around the peak is typically excluded from the standard deviation to reduce the effects of large neighboring values close to the peak. The PSR scores from all peaks in the *N* correlation planes were thresholded and collected into a single set of target detections for the frame.

The overall performance of a filter bank on a video sequence was evaluated by summing the results from all individual frames. The following conditions were used to record the recognition results of each frame:

- *TP (true positive)* – The target of interest (TOI) appeared in the frame, and a peak was detected within its ground truth bounding box.
- *FP (false positive)* – One or more peaks were detected at locations other than the location of the TOI.
- *TN (true negative)* – The TOI did not appear in the frame, and no peaks were detected anywhere in the frame.
- *FN (false negative)* – The TOI appeared in the frame, but no peaks were detected.

The numbers of occurrences of each condition across all testing frames were summed to produce the cumulative result. Note that it is possible for both TP and FP to occur in any given frame; for example, if both a true target and a false target were detected in a frame, we would assign both the TP and the FP conditions to that frame.

4. PERFORMANCE ESTIMATION USING FISHER RATIOS

Evaluating the performance of a classifier on a particular dataset can be done by simply using the classifier to perform recognition and observing the resulting error rate. However, this process can be computationally intensive, and furthermore we might not know whether the selected classifier is the best one for the dataset. It is therefore of interest to be able to estimate the performance of some unknown, “well-chosen” classifier on a given dataset without first designing such a classifier and testing it on the dataset. In other words, we would like to determine the difficulty of the dataset independent of the specific classifier used. In the context of video compression, it is expected that higher compression ratios will cause more information loss and degrade the image quality of the ATR sequences, resulting in a more difficult target recognition problem. We therefore attempt to evaluate the effects of compression on the video sequences with respect to the expected performance of a correlation filter classifier by examining only the data itself.

When a correlation filter is applied to an input image, the resulting output is inspected for peaks. The PSR metric is commonly used to measure the likelihood of the target at a detected peak; however, if the energy in the correlation plane remains approximately constant from one image to another, then peak height alone would be a comparable measure of target likelihood. The peak height would be thresholded to determine whether a target is present. We can treat the peak height as the projection of a shifted version of the input image onto the correlation filter template, illustrated by the following correlation equation:

$$c[m,n] = \sum_k \sum_l x[k-m, l-n]h[k,l], \tag{3}$$

where $c[m,n]$ is the correlation peak height at location (m,n), $x[m,n]$ is the input image, and $h[m,n]$ is the filter template. If we rewrite this expression in vector notation as follows:

$$c_{mn} = \mathbf{x}_{mn}^T \mathbf{h} \tag{4}$$

then it is easy to see that the template vector **h** essentially acts as a linear discriminant that separates target vectors (or equivalently the input image vector **x** shifted so that the target appears at the origin) from non-targets (the input image shifted so that a target is not present at the origin) in terms of peak height.

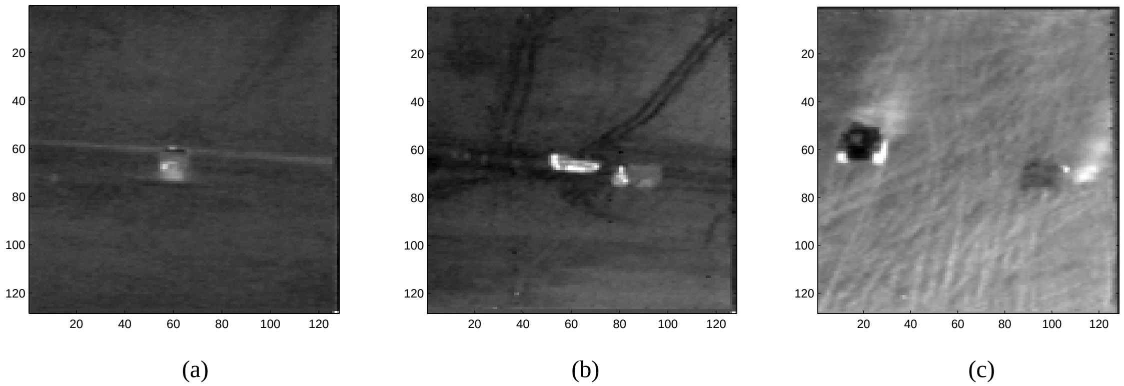
We invoke the above argument to propose that Fisher linear discriminants can be used to predict the effects of video compression on a given dataset. In other words, by forming two classes of images from the video sequence, one containing centered target images and the other containing non-target images and non-centered target images, and using a Fisher linear discriminant to separate the two classes, we can estimate the expected performance degradation of a correlation filter classifier as a function of the compression rate. The corresponding Fisher ratio, a measure of the separability of the two classes, can then be considered an estimate of the relative difficulty of the dataset for a given compression rate with respect to a correlation filter. The Fisher ratio F is defined as follows:

$$F = \frac{|m_1 - m_2|^2}{s_1^2 + s_2^2} \tag{5}$$

where m_1 and m_2 are the projected means of each class of vectors, and s_1 and s_2 are the variances of the projected values. The Fisher linear discriminant maximizes this ratio and thus achieves maximal separation of the two classes in the Fisher sense, resulting in high discriminant values for Class 1 and low values for Class 2 on average.

Consider the set of all shifted versions of each frame in some time portion of the video sequence of interest. If we were to take all such frames in which the true target is centered at the origin and group them into Class 1, and group all the remaining frames into Class 2, the resultant Fisher linear discriminant vector would likely be a good correlation template for target recognition on this sequence. This is because for each possible 2-D shift, the projection of the shifted test image onto the discriminant vector would yield high values when the target is centered at the origin and low values everywhere else. Moreover, these high and low values would be maximally separated according to the Fisher criterion, which would be beneficial for peak thresholding. Of course, the discriminant vector would be highly specific (and possibly overfitted) to the particular dataset on which it was trained; however, we are interested in determining the best possible recognition performance on the given data, and thus we seek the best possible classifier within the limits of conventional correlation filtering.

In our performance estimation experiments, we partitioned sequences into frame chunks and evaluated Fisher ratios on each chunk separately. The reason for this was to imitate the recognition scheme of correlation filter banks, where each filter in the bank is trained on a particular chunk. We elected not to use the full-size 128 x 128 images in the computation but instead to use smaller *m* x *n* images approximately the size of the largest target in the chunk. This is because the correlation filter template values are typically only non-zero in a small target-sized area, which causes only a small portion of the template image (and thus the testing image) to actually contribute to the resulting correlation value. For a given frame chunk of size N frames in which every frame contains a target, we first extracted N centered $m \times n$ target images from the frames, registered these images by cross-correlating with the largest image in the set, and grouped them into Class 1. Next, we sampled 25 $m \times n$ clutter images from evenly-spaced regions around each frame, resulting in a total of 25N images, and grouped these into Class 2. This density of clutter sampling is much smaller than the entire set of possible clutter samples (that is, $m \times n$ images extracted from each frame at *every* possible shift); however, we believe that 25 samples per frame is an adequate sampling since most of the background regions are of reasonably uniform texture and appearance. In cases where one or more false targets appeared in the scene, we also included $m \times n$ images of these targets in Class 2. Each image in the classes was represented as a vector of length mn by lexicographic ordering of the image pixels. The Fisher ratio was computed using Eq. (5).



(a) (b) (c)

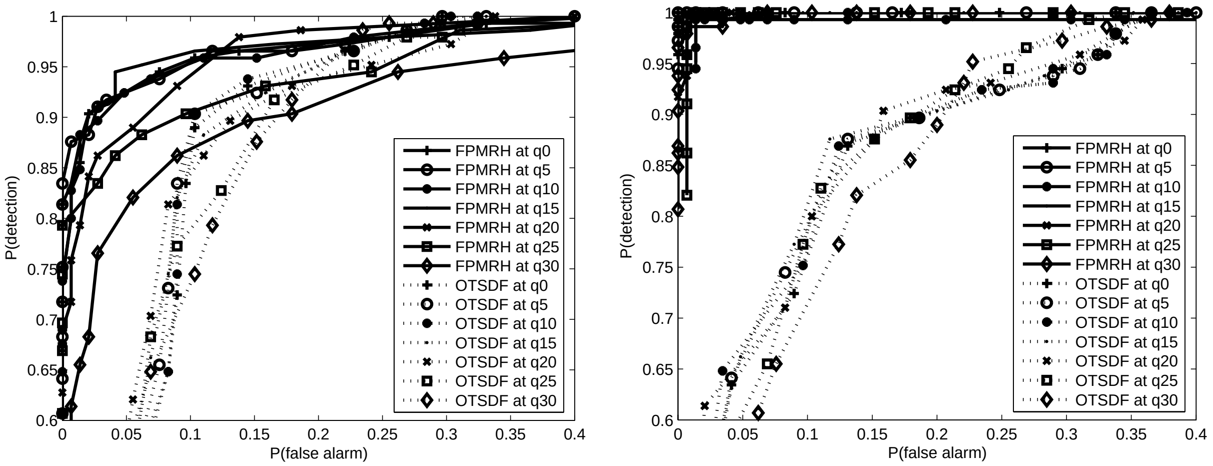
Figure 4: Sample frames from the video sequences: (a) “L1415” frame 200, (b) “L1608” frame 230, and (c) “L1816” frame 280.

5. EXPERIMENTAL RESULTS

In this section we show recognition results from several sequences using FPMRH and OTSDF correlation filters. Recognition performance is presented using ROC curves, PSR plots, and “equal-error rate” (EER) curves. The PSR plots show the maximum PSR value obtained in each frame for both the true-class target (if it appears) and all false-class objects, including clutter and false targets. These plots convey the actual separability (in terms of PSR) between true class and false class. For the EER curves, we define EER as the false alarm rate at the point on the ROC curve at which the detection rate plus the false alarm rate equals one, or equivalently, the intersection of the ROC curve with the line from (0,1) to (1,0). The EER metric is a single-value description of the overall performance of the classifier; thus, in order to concisely illustrate performance as a function of compression rate, we plot the detection rate of the classifier for each compression level at the EER point.

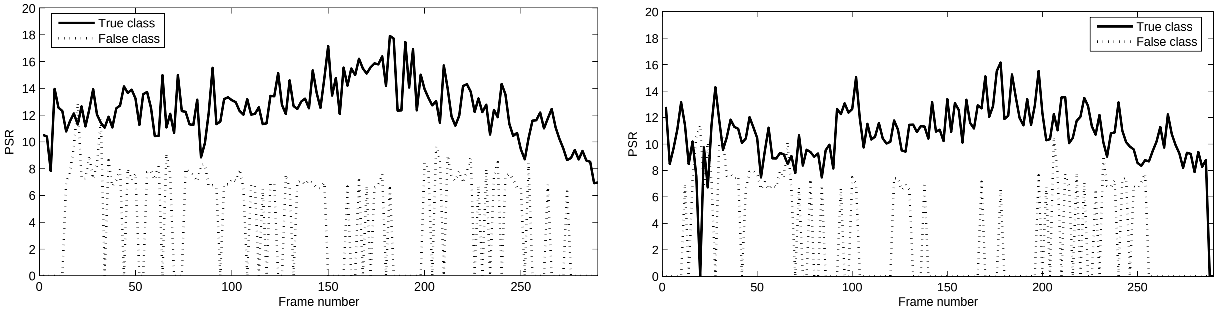
We also show Fisher separability metrics alongside the EER curves for comparison. Note that we do not compare the actual values of the Fisher ratios with the corresponding EER values, as a precise relationship between these two quantities has not been established. Instead, we compare the overall trend of each Fisher curve to that of the EER curve. We observe that although a few frame chunks yield rather misleading results, in general the Fisher ratio curves tend to rise and fall in close agreement with the EER curves. Those frame chunks derived from the end of the sequence (e.g. chunks 3 and 4 in a 4-chunk partition) generally contain very few images because of increasingly large scale variability, and thus their Fisher ratio curves will be neither as informative nor as reliable as those from the beginning frame chunks.

The first set of plots shows recognition results from sequence L1608. This sequence contains two targets, an M60 tank (true-class target) and a truck (false-class target). The truck partially obscures the tank for approximately 10-15 frames near the end of the sequence. A sample frame from this sequence is shown in Fig. 4. Fig. 5 shows the ROC curves generated by using FPMRH and OTSDF filter banks of both 4 filters and 7 filters on this sequence. The FPMRH filter banks outperform the OTSDF banks in almost all cases and achieve near 100% recognition at all compression levels with 7 filters. Fig. 6 shows PSR plots for the 4-filter FPMRH bank at the compression level extremes. Fig. 7(a) shows the detection rate performance of several filter banks at the EER points of their respective ROC curves across all seven compression levels. We included two FPMRH banks and an OTSDF bank. We show in Fig. 7(b) the Fisher ratio curves from each of four frame chunks of sequence L1608. By observing the Fisher curves (in particular those from the beginning frame chunks), we expect that correlation filter performance should remain relatively stable for compression levels up to q15 or q20 and should degrade under higher compression. Comparing this observation to the actual performance curves suggests that the Fisher estimates are of some value in predicting performance degradation.



(a) (b)

Figure 5: ROC curves for sequence L1608 using (a) 4 filters per bank and (b) 7 filters per bank. The curves were generated at different compression levels (described by the quantization step size *q*) using FPMRH filters and OTSDF filters.



(a) (b)

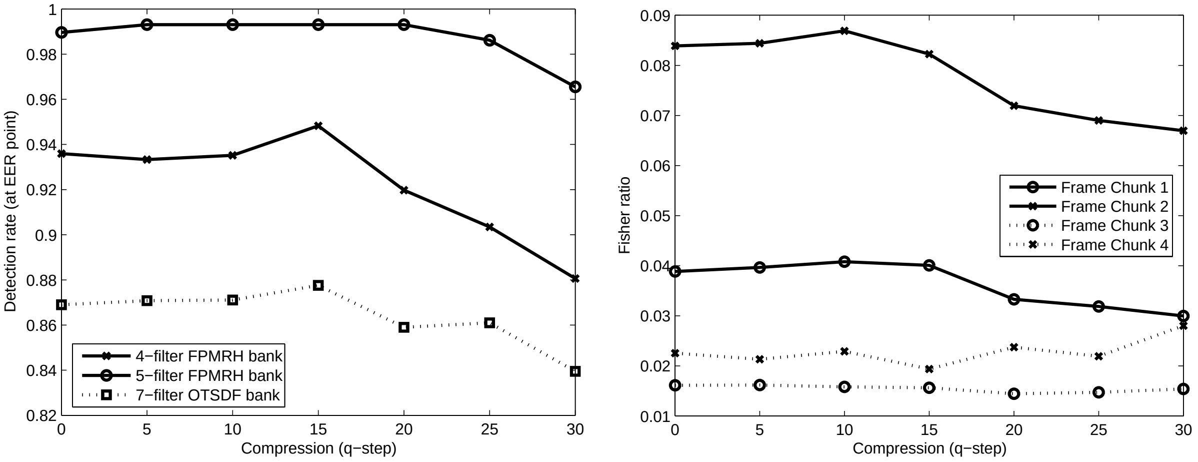
Figure 6: Plots of PSR scores for true-class targets and false-class objects in sequence L1608 at (a) no compression and (b) q30 compression. These plots illustrate the level of separability between targets and clutter in terms of PSR scores.

We show additional results from sequences L1415 and L1816 in Figs. 8-10. Sample frames from these sequences are shown in Fig. 4. In these sequences as well as in the previous sequence we observe that compression levels up to q15 do not have a significant negative impact on performance; in fact, many cases show that increased compression up to q15 actually improves performance by a small amount. Accordingly, the Fisher ratio curves for these sequences predict a similar trend in general. These results also show that increasing the number of filters per bank provides improved performance; unfortunately, such improvement comes at the cost of increased computation. The PSR plots in Fig. 8 show the decreased margin of separation going from no compression to high (q30) compression.

6. CONCLUSIONS

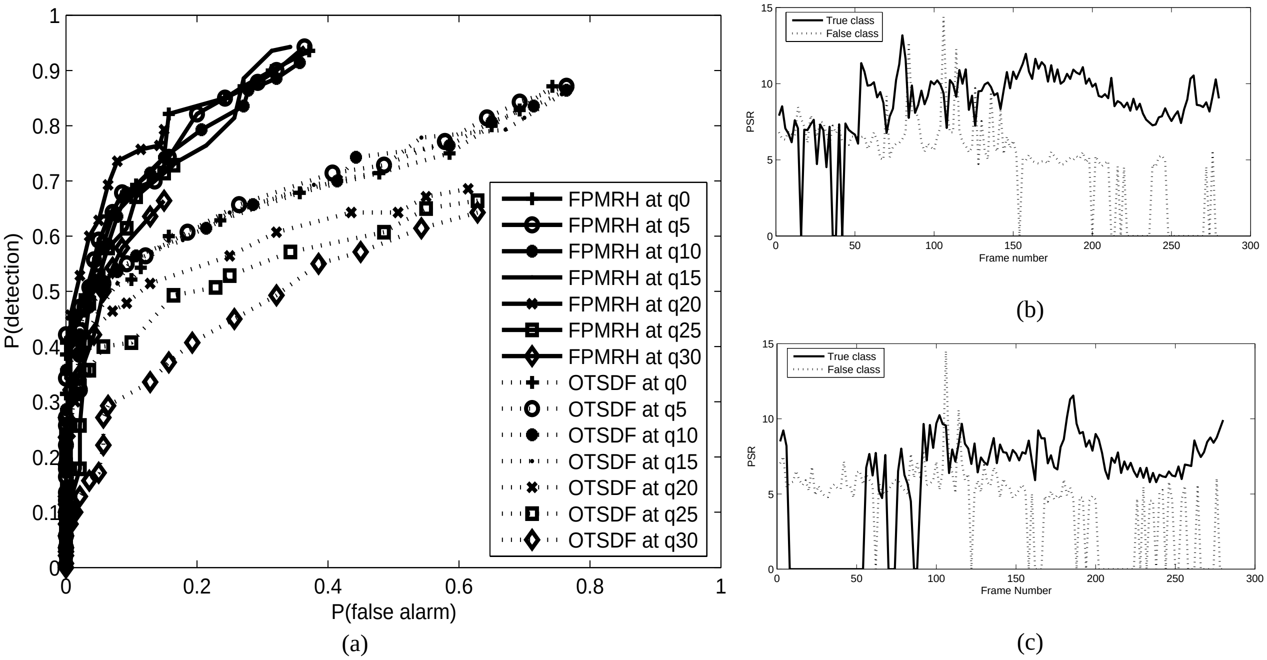
We have demonstrated the performance of FPMRH and OTSDF correlation filter classifiers on several video sequences from a database of compressed and decompressed IR imagery. The FPMRH filters outperformed the OTSDF in most cases, despite the fact that the OTSDF filters had the advantage of much more training data. Recognition performance was generally unharmed by H.264 video compression up to the q15 quality level (corresponding to approximately a 6:1 compression ratio); higher compression levels, especially q25 and q30, resulted in a much more significant loss of performance. Further investigation of this subject should include more types of correlation filters, such as MACH filters and quadratic correlation filters, as well as the use of clutter samples for training the filters.

We also demonstrated the usefulness of the proposed Fisher separability metric for predicting performance degradation. The Fisher curves computed from the beginning frame chunks offered a reasonably accurate estimate of the highest compression level that could be used without a significant impact on performance, as most of the curves exhibited a distinct downward slope after level q20. Denser clutter sampling for the Fisher classes might lead to improved accuracy and reliability. Nevertheless, a precise mathematical relationship between correlation filter performance and Fisher ratios has not been established, and thus a much larger study is needed in order to verify the reliability of such estimates.



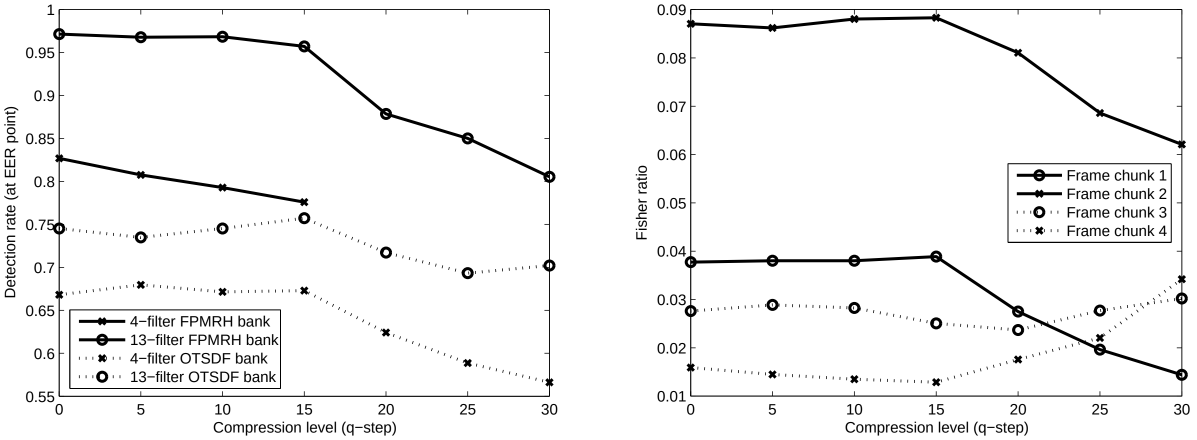
(a) (b)

Figure 7: (a) Performance of three different classifiers on sequence L1608 as a function of compression level. (b) Fisher ratio between true-class and false-class images in this sequence as a function of compression level. Fisher ratio curves are shown for each of the four frame chunks used to design the 4-filter bank. Results from frame chunks 3 and 4 were scaled by factors of 0.2 and 0.015 for visual comparison.



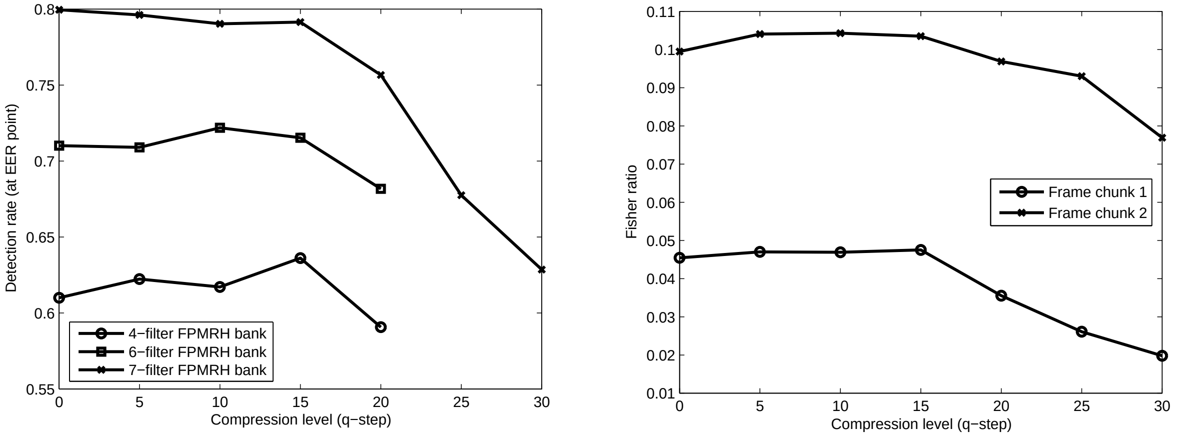
(a) (b) (c)

Figure 8: Performance plots for sequence L1415. ROC curves for 4-filter FPMRH and OTSDF banks are shown in (a). PSR scores for the 4-filter FPMRH bank are shown at (b) no compression and (c) q30 compression.



(a) (b)

Figure 9: (a) Performance of four classifiers on sequence L1415 as a function of compression level. (b) Fisher ratio between true-class and false-class images in this sequence as a function of compression level. Fisher ratio curves are shown for each of the four frame chunks used to design the 4-filter banks.



(a) (b)

Figure 10: (a) Performance of three FPMRH classifiers on sequence L1816 versus compression level. (b) Fisher ratio between true-class and false-class images in this sequence as a function of compression level. Fisher ratio curves are shown for the first two of the four frame chunks used to design the 4-filter banks. The target was not present in the last two frame chunks of the sequence.

REFERENCES

1. A. Luthra, P. Topiwala, “Overview of the H.264 Video Compression Standard,” *Proc. SPIE Int’l Symposium, Appl. Dig. Imaging*, 2003.
2. J. Starch, R. Sharma, and S. Shaw, “A unified approach to feature extraction for model based ATR,” *Proc. SPIE* vol. 2757, 1996.
3. L. A. Chan, S. Z. Der, and N. M. Nasrabadi, “Neural based target detectors for multi-band infrared imagery,” *Image Recognition and Classification, Algorithms, Systems, and Applications*, Marcel Dekker, Inc., New York, 2002.
4. B. Bhanu and J. Ahn, “A system for model-based recognition of articulated objects,” *Proc. Intl. Conf. Patt. Recog.*, 1998.

5. R. Singh and B. V. K. Vijaya Kumar, “Performance of the extended maximum average correlation height (EMACH) filter and the polynomial distance classifier correlation filter (PDCCF) for multi-class SAR detection and classification,” *Proc. SPIE,* vol. 4727, 2002.

6. S. R. F. Sims and A. Mahalanobis, “Performance evaluation of quadratic correlation filters for target detection and discrimination in infrared imagery,” *Opt. Eng.,* vol. 43, 2004.

7. P. Refregier, “Filter design for optical pattern recognition: multicriteria optimization approach,” *Opt. Lett.,* vol. 15, 1991.

8. R. A. Kerekes and B. V. K. Vijaya Kumar, “Correlation filters with controlled scale response,” submitted to *IEEE Trans. Image Proc.*, May 2004.

9. A. Moya, J. Esteve-Taboada, J. Garcia, and C. Ferreira, “Shift- and scale-invariant recognition of contour objects with logarithmic radial harmonic filters,” *Appl. Opt.,* vol 39, 2000.

10. R. Kerekes, M. Savvides, B. V. K. Vijaya Kumar, and S. R. F. Sims, “Fractional power scale-tolerant correlation filtering for enhanced automatic target recognition performance,” *Proc. SPIE Defense and Security Symposium*, 2005.