UAV Based Distributed ATR under Realistic Simulated Environmental Effects

Xiaohan Chen, Shanshan Gong, Natalia A. Schmid and Matthew C. Valenti

Lane Department of Computer Science and Electrical Engineering
West Virginia University, Morgantown, WV 26506

ABSTRACT

Over the past several years, the military has grown increasingly reliant upon the use of unattended aerial vehicles (UAVs) for surveillance missions. There is an increasing trend towards fielding swarms of UAVs operating as large-scale sensor networks in the air.[1] Such systems tend to be used primarily for the purpose of acquiring sensory data with the goal of automatic detection, identification, and tracking objects of interest. These trends have been paralleled by advances in both distributed detection,[2] image/signal processing and data fusion techniques. Furthermore, swarmed UAV systems must operate under severe constraints on environmental conditions and sensor limitations. In this work, we investigate the effects of environmental conditions on target detection and recognition performance in a UAV network. We assume that each UAV is equipped with an optical camera, and use a realistic computer simulation to generate synthetic images. The detection algorithm relies on Haar-based features while the automatic target recognition (ATR) algorithm relies on Bessel K features. The performance of both algorithms is evaluated using simulated images that closely mimic data acquired in a UAV network under realistic environmental conditions. We design several fusion techniques and analyze both the case of a single observation and the case of multiple observations of the same target.

Keywords: Unattended Arial Vehicles, Automatic Target Recognition, Optical Camera, Data Fusion, Detection

1. INTRODUCTION

Recent advances in wireless communication, signal processing, and distributed control made large-scale sensor networks of UAVs a potentially most reliable machinery for surveillance missions. There is an increasing trend towards fielding swarms of UAVs operating as large-scale sensor networks in the air.[1] These systems are intended to be used primarily for the purpose of surveillance or search of disaster areas and thus for acquiring sensory data. The data are further processed using state-of-the art detection, recognition, and tracking algorithms. The challenge that swarmed UAV systems face is that they must operate under severe communication constraints, varying environmental conditions and sensor limitations. In this paper, we place the emphasis on the environmental and sensor limitations and explore a possibility to improve detection and recognition performance by means of data fusion.

We focus on the case where the acquired sensory data are in the form of optical images. Traditionally, optical cameras are low in cost and small in size, which makes them a high preference imagery sensors for a variety of military and civilian applications. The major limitation of optical cameras is their inability to deal with environmental conditions and imperfect camera set ups which lowers fidelity of the results in detection and recognition tasks. In this work, we investigate the influence of environmental and camera effects on the performance of selected detection and recognition algorithms. We use a realistic computer simulation to generate synthetic images. The detection algorithm relies on Haar-based features while the automatic target recognition (ATR) algorithm relies on Bessel K features. The performance of both algorithms is evaluated using simulated images that closely mimic data acquired in a UAV network under realistic environmental conditions. We further implement data fusion techniques and demonstrate detection and recognition performance improvements due to data fusion.

Further author information: (Send correspondence to Natalia A. Schmid)
Xiaohan Chen and Shanshan Gong: E-mail: {xchen10,sgong1}@mix.wvu.edu
Natalia A. Schmid and Matthew C. Valenti: E-mail: {Natalia.Schmid, Matthew.Valenti}@mail.wvu.edu

1.1. Literature Review

The literature contains a large number of detection, recognition, and data fusion algorithms applied to optical data. While it is impossible to list here all published works, we choose to characterize a few.

Detection of a possible object of interest is one of the most critical steps in object recognition problems, since the results of postprocessing depend on this step. Target detection approaches can be classified into three categories: feature invariant approaches, template matching methods and appearance-based methods. In feature invariant approaches, the algorithms aim to find structural features such as edges,[3] textures, etc. that exist even when the pose, viewpoint, or lighting conditions vary, and then use the features to locate targets. In template matching methods, several standard patterns of a target are stored to describe the target. The correlations between an input image and the stored patterns are computed for detection. In contrast to template matching, the models (or templates) in appearance-based methods[4] are learned from a set of training images which should capture the representative variability of target appearance. These learned models are then used for detection. In this work, we will use a local Haar filter-based detection method which is very popular in the field of face recognition.

Based on encoded information, ATR algorithms are broadly classified into three categories: shape-based, appearance-based, and Computer Aided Design (CAD)-based methods. In shape-based recognition, the contour of the object is extracted, and then the shape templates are used to match the extracted contours. In appearance-based (or view-based) approach, the 2D intensity templates of 3D target acquired from different viewpoints are stored as a model. Some view-based methods use statistical techniques to analyze the distribution of the target image vectors in the vector space, and derive an effective representation (feature space) according to different applications. Other methods design distortion-invariant filters to perform a correlation matching between the model view and the input image. In CAD-based ATR, an explicit 3D model of a target is generated and subsequently used in target recognition employing imagery acquired by a variety of sensors.

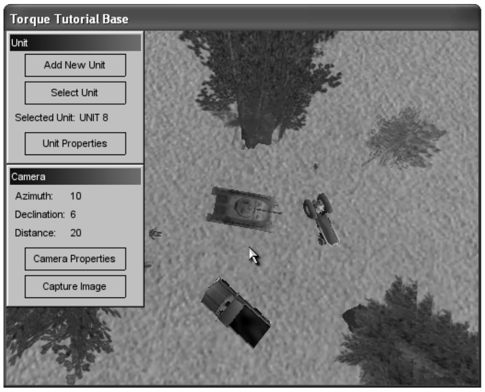
Multi-sensor data fusion system can be characterized by levels:[5] signal, pixel, feature and decision-level. The first level (called signal level) concerns with the aggregation of raw data provided directly from sensors, without any transformation. Pixel or image level fusion creates new images that are more suitable for the purposes of object detection and recognition. The next fusion method is feature level fusion. The raw data are first encoded (features are extracted) before being aggregated. Finally, the highest abstraction level corresponds to the decision fusion. It is reduced to combining decisions proposed by classifiers/detectors.

Most of the algorithms summarized above assume good quality data for training and performance evaluation. However, in practice these algorithms would be subject to highly distorted and noisy data. This work strives to investigate the limitations of two state-of-the art (detection and recognition) algorithms.

2. DATA DESCRIPTION

2.1. Simulated Optical Data

An ATR Training Tool provided by Augusta Systems Inc. was used to build a simulated database. The tool is capable of generating prospective projections of 18 distinct objects projected at different orientation and elevation angles and sampled at distinct resolutions. The objects can be manually superimposed onto a background to simulate various ground conditions. The camera parameters such as position, azimuth, declination and distance can be varied to simulate an UAV flight. The resolution of captured images can be adjusted from 512×384 to 1152×864. A snapshot of the Graphical User Interface (GUI) of the tool is shown in Fig. 1. Every image generated by the 3D optical tool is first processed by a target detector and then fed into a recognition system. Prior to recognition, a potential target is located and placed in a canonical (or object-centered) reference frame suitable for recognition. In our experiments, we use three target types: tank, truck, and tractor. Sample images used for recognition are shown in Fig. 2. Each 3D target is projected using discrete orientation angles spaced 5 degree apart and elevation angles from 0 to 45 spaced 15 degree apart.



Selected Unit: UNIT 8

Azimuth: 10

Declination: 6

Distance: 20

Figure 1. The GUI of the ATR training tool.

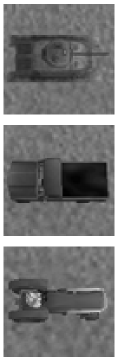


Figure 2. Sample targets for recognition from simulated ATR database.

2.2. Simulated Environmental and Camera Effects

Apart from generated images of objects as described in the previous section, we expand the dataset by adding six distorted versions of each original image. These simulate various camera and environmental effects that can occur in real world images. The distorted images include one of the following factors: Gaussian noise, illumination effect, varying contrast, motion blur and defocus blur. By controlling the value of the parameters, different levels of noise in images can be generated. The details of generation procedures are summarized below.

1. Images contaminated by Gaussian noise contain additive white noise with zero mean and variance σ^2. The variance σ^2 takes values in the range from 0.005 to 0.02 spaced 0.005 apart for Level 1 to Level 4, respectively.

2. The images are brightened or darkened by increasing or decreasing the intensities. This procedure simulates illumination effect. Denote by β the parameter that controls the level of illumination. We first normalize image intensities to $(0, 1)$, then brighten images by raising to the power of a number less than one, that is, $(1 - \beta,\ \beta \in (0, 1))$ or darken images by raising to the power of a number larger than one, that is, $(\frac{1}{\beta+1},\ \beta \in (-1, 0))$. The parameter β is set to be -0.8 and -0.4 at Levels 1 and Level 2 for dark images and is set to be 0.4 and 0.8 at Levels 3 and 4 for brightening images.

3. We model contrast change by linear mapping the normalized histogram to a new one. If the histogram is "squeezed," then the new image will have low contrast. The more compression, the lower the contrast is. The range is determined by parameter $1 - 2TOL$ with TOL taking values in the range from 0.15 to 0.35 spaced 0.05 for Level 1 to Level 4, respecftively.

4. A linear relative motion of an optical camera or an object is simulated by convolving images with a two parameters point spread function (PSF).[6] Length L in pixels and angle θ in degrees correspond to motion in specific direction with predefined camera velocity. L takes values in the range from 2 to 8, 2 units apart for Levels 1 through 4, respectively. The parameter θ follows uniform distribution on $[0, 360°]$ for all levels.

5. The images are filtered by a two-dimensional circular averaging filter to generate defocus blur.[6] Defocus level corresponds to the radius r of the averaging filter. r takes values from 2 to 8 with the step 2 units for Levels 1 through 4, respectively.

The samples of distorted images from the tractor are displayed in Fig. 3.

3. SINGLE- AND MULTI-FRAME TARGET DETECTION AND RECOGNITION

In this work we adopt Haar feature-based detection algorithm. The algorithm was originally developed as a face detector.[4] Its robust performance and computational efficiency when applied to optical images motivated us to

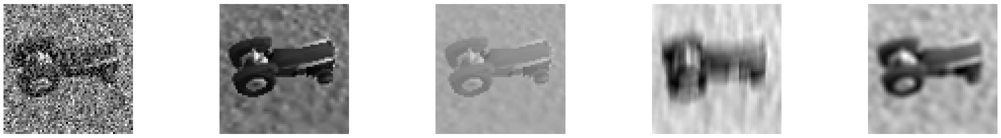


Figure 3. Distorted images of the tractor. From left to right: the image with additive Gaussian noise, the image characterized by a low illumination, a low contrast image, motion-blurred image and defocused image.

use the algorithm in this work. Detected regions within images are further subjected to recognition method based on Bessel K forms. Previously Bessel K forms were successfully used to perform analysis of natural images.[7] Since UAV network may contain multiple image copies of the same target, we develop data fusion techniques for improved detection and recognition.

3.1. Detection based on a single frame

The employed target detector framework is a modified version of the Viola-Jones face detector.[4] A software copy of the algorithm is available through the Open Computer Vision Library.

The rapid target detection scheme is based on the idea of a boosted classifier cascade[4] but extends the original feature set and offers different boosting variants for learning. The classifier cascade is trained on a set of positive images (targets) and a set of negative images (non-targets). For each training image, an over-complete set of Haar-like feature pool is calculated and AdaBoost algorithm of Schapire and Singer[8] is used to build a stage classifier. After the classifier cascade is trained, the detection algorithm is applied to a query image. A search window is sled over the query image. At each window location and scale the content of the window is classified as target or non-target.

In each round of boosting, a weak learning algorithm is applied to select a single rectangle feature which best separates the positive and negative samples. For each feature, the weak learner determines the optimal threshold classification function, such that the minimum number of examples are misclassified. Thus, a weak classifier $h_j(\cdot)$ is a binary valued function obtained by comparing the j-th feature value $f_j(\cdot)$ with a threshold θ_j:

$$h_j(x) = \begin{cases} \alpha_j & if \;\; f_j(x) > \theta_j \\ \beta_j & otherwise \end{cases} \tag{1}$$

Here x is a sub-window of an image. The value of the feature is equal to weighted differences of integrals over rectangular subregions. α_j and β_j are positive or negative votes of each feature set by AdaBoost during the learning process. θ_j is the optimal threshold obtained by the weak learner.

The form of the final stage classifier returned by AdaBoost is a thresholded linear combination of weak classifiers (see Fig. 4). The stage classifier is given by:

$$C(x) = \begin{cases} 1, & if \;\; \sum_j h_j(x) > T, \\ 0, & otherwise, \end{cases} \tag{2}$$

where T is the stage threshold set by AdaBoost during the learning process.

In order to improve computational efficiency and also reduce the false positive rate, a sequence of increasingly more complex classifiers called cascade is used. A cascade of classifiers is a degenerated decision tree where at each stage almost all objects of interest are detected while only a certain fraction of the non-object patterns are rejected. The more an input window looks like an object, the larger the number of classifiers are evaluated on it and the longer it takes to classify the window. Since most windows of an image do not look like objects, they are quickly discarded as non-objects. Fig. 5 illustrates a cascade.

To evaluate detection performance, we involve Receiver Operating Characteristic (ROC) curves. The detection threshold is selected as the threshold of the final classifier stage. Adjusting the threshold to $+\infty$ will yield a detection rate of 0.0 and a false positive rate of 0.0. Adjusting the threshold to $-\infty$, however, increases both the detection rate and false positive rate, but only to a certain point. In fact, a threshold of $-\infty$ in the

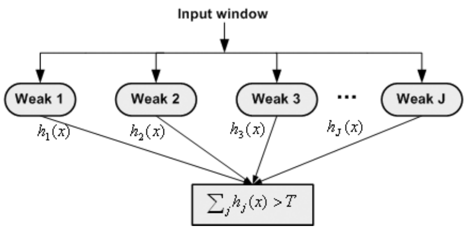


Figure 4. Stage classifier.

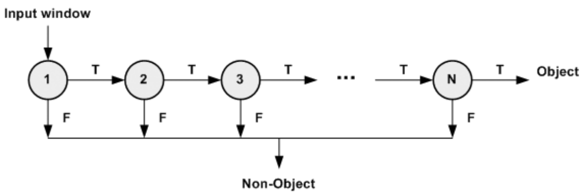


Figure 5. Cascade of classifiers.

final layer is equivalent to removing that layer. Further increasing the detection and false positive rates requires decreasing value of the threshold of the next classifier in cascade. Thus, in order to construct a complete ROC curve, classifier layers are removed one by one. We use the number of false positives as opposed to the rate of false positive to label the x-axis. The false positive rate can be calculated by simply dividing the number of false positives by the total number of scanned sub-windows.

3.2. Recognition based on a single frame

Comprehensive studies[7] of natural scenes have shown that the distributions of pixel intensities in linearly filtered images are described by a family of Bessel K distribution functions. This constitutes a basis for the implemented recognition algorithm.

Bessel K forms is a stochastic model that can be used to measure image variability. This parametric family is applied to model lower order probability densities of pixel values resulting from bandpass filtering of images. The main idea of the recognition algorithm based on Bessel K forms is to select the critical features of each object class by passing an image through a bank of linear filters and then analyzing statistics of the filtered images. As shown by Grenander and Srivastava,[7] Bessel K forms parameterized by only two parameters: (1) the shape parameter p, $p > 0$, and (2) the scale parameter c, $c > 0$, may provide a good statistical fit to empirical histogram distributions of filtered images.

Denote by I an image and by $\mathcal{F}$ a filter, then the filtered image $\mathcal{I} = I * \mathcal{F}$, where $*$ denotes 2-dimensional convolution operation. Under the conditions stated in,[7] the probability density function of the random variable $\mathcal{I}(\cdot)$ can be approximated by

$$f_K(x; p, c) = \frac{2}{Z(p,c)} |x|^{p-0.5} K_{(p-0.5)}(\sqrt{\frac{2}{c}}|x|), \tag{3}$$

where $K_\nu(x)$ is the modified Bessel function of the second kind, and $Z(p,c)$ is the normalization given by

$$Z(p,c) = \sqrt{\pi}\Gamma(p)(2c)^{0.5p+0.25}.$$

Given J filters, the image I can be represented using $2J$ Bessel parameters.

To approximate the empirical density of the filtered image by a Bessel K form, the parameters p and c are estimated from the observed data using

$$\hat{p} = \frac{3}{SK(\mathcal{I}) - 3} \text{ and } \hat{c} = \frac{SV(\mathcal{I})}{\hat{p}}, \tag{4}$$

where SK is the sample kurtosis and SV is the sample variance of the pixel values in $\mathcal{I}$. Since the moment-based estimate of p in (4) is sensitive with respect to outliers, in our computations we replace it with an estimate based on empirical quartiles given by

$$\hat{p} = \frac{3}{\hat{SK}(\mathcal{I}) - 3}, \quad \text{with} \quad \hat{SK}(\mathcal{I}) = \frac{q_{0.995}(\mathcal{I}) - q_{0.005}(\mathcal{I})}{q_{0.75}(\mathcal{I}) - q_{0.25}(\mathcal{I})},$$

where $q(\cdot)$ is the quartile function that returns the x quartile of a set of samples. This method provides reasonable fit. As shown in Fig. 6, the histogram (dashed line) of images filtered by Gabor filters[9] closely follows the

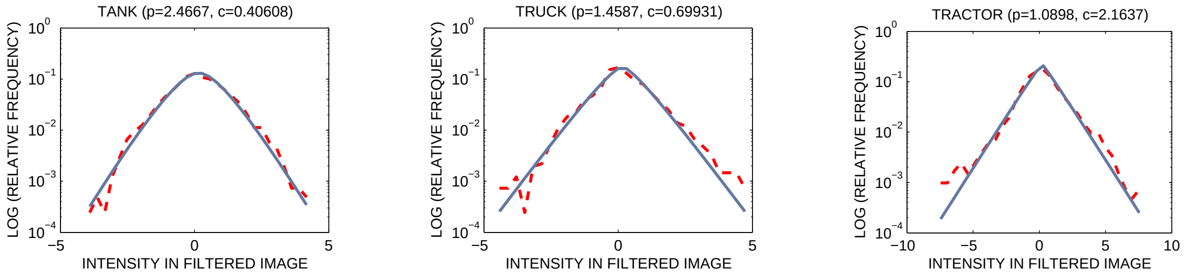


Figure 6. The marginal densities. The empirical histogram distributions are marked in dashed line. The Bessel K form approximations are shown in solid lines.

estimated Bessel K forms (solid line). To quantify the difference between two filtered images based on their distributions, two distance measures: (1) a pseudo-metric introduced by Srivastava[7] and (2) the K-measure[10] between two Bessel K forms, are used. The pseudo-metric is defined as

$$d_I(\mathcal{I}_1, \mathcal{I}_2) = \sqrt{\int_{-\infty}^{+\infty} \left(f_K(x; p_1, c_1) - f_K(x; p_2, c_2)\right)^2}. \tag{5}$$

The closed form of $d_I(\mathcal{I}_1, \mathcal{I}_2)$ for the case of $p_1, p_2 > 0.25, c_1, c_2 > 0$ is given by:

$$d_I(\mathcal{I}_1, \mathcal{I}_2) = \left[\frac{\Gamma(0.5)}{2\sqrt{2\pi}} \left(\frac{\mathcal{G}(2p_1)}{\sqrt{c_1}} + \frac{\mathcal{G}(2p_2)}{\sqrt{c_2}} - \frac{2\mathcal{G}(p_1+p_2)}{\sqrt{c_1}} \left(\frac{c_1}{c_2}\right) p_2 \mathcal{H}\right)\right]^{\frac{1}{2}},$$

where $\mathcal{G}(p) = \frac{\Gamma(p-0.5)}{\Gamma(p)}$ and $\mathcal{H} = H\left(p_1 + p_2 - 0.5, p_2; p_1 + p_2; 1 - \frac{c_1}{c_2}\right)$. The function H is the hypergeometric function. In cases where $\hat{p} < 0.25$ for an image-filter combination, we compute the pseudo-metric numerically using the quadrature integration.

The K-measure is defined as

$$d_{KL}(\mathcal{I}_1, \mathcal{I}_2) = D\left(f_K(x; p_1, c_1) \| f_K(x; p_2, c_2)\right) + D\left(f_K(x; p_2, c_2) \| f_K(x; p_1, c_1)\right), \tag{6}$$

where $D\left(f_K(x; p_1, c_1) \| f_K(x; p_2, c_2)\right)$ is the relative entropy between two distribution functions $f_K(x; p_1, c_1)$ and $f_K(x; p_2, c_2)$ given by

$$D\left(f_K(x; p_1, c_1) \| f_K(x; p_2, c_2)\right) = \int_{-\infty}^{+\infty} \log\left(\frac{f_K(x; p_1, c_1)}{f_K(x; p_2, c_2)}\right) f_K(x; p_1, c_1) dx.$$

In the above expressions $f_K(\cdot)$ is the Bessel K probability density function introduced in (3).

Given two images $\{I_1, I_2\}$ and a bank of filters $\{\mathcal{F}_j, j = 1, 2, \cdots, J\}$, we evaluate a set of filtered images $\{\mathcal{I}_{(n,j)} = I_n * \mathcal{F}_j,\ n = 1, 2;\ j = 1, \cdots, J\}$. After estimating the parameter $p_{(n,j)}$ and $c_{(n,j)}$, each image is mapped to J points in the density space. The distance between two images are calculated by

$$d_I(I_1, I_2) = \sum_{j=1}^{J} d_I(\mathcal{I}_{(1,j)}, \mathcal{I}_{(2,j)}), \tag{7}$$

and

$$d_{KL}(I_1, I_2) = \sum_{j=1}^{J} d_{KL}(\mathcal{I}_{(1,j)}, \mathcal{I}_{(2,j)}), \tag{8}$$

where $d_I(\mathcal{I}_{(1,j)}, \mathcal{I}_{(2,j)})$ and $d_{KL}(\mathcal{I}_{(1,j)}, \mathcal{I}_{(2,j)})$ are defined in (5) and (6).

The purpose of using the two distance measures is to balance accuracy and computational efficiency. K-measure is an accurate measure of similarity of two probability density functions. However, it cannot be obtained in closed form for Bessel K forms. Numerical evaluation of K-measure is computationally expensive. The pseudo-metric (5) has closed form for Bessel K forms, which means that the computation cost is relatively low. The major drawback of the pseudo-metric is its low precision. To measure the difference between two histograms fast and with relatively high precision, we combine these two distance measures. First, we use the fast method, the pseudo-metric, to evaluate the distance between the input image and all templates in the database. If the pseudo-metric has multiple minima close in their values, there will be a potential misclassification. The precise metric, the K-measure, is then used to re-calculate the distances and make the final decision. By setting threshold properly, we obtain relatively fast and reliable result.

3.3. Data fusion for improved detection

In this section, we motivate and describe two data fusion methods for improved detection performance.

Consider a scenario where a set of UAVs perform an area search. UAVs monitor the ground continuously at a slow rate (for instance, 2-5 frames per second). We assume that an UVA while passing a target is capable of acquiring only a few (1-4 frames) containing this target. Now, if a UAV detects a potential target within a frame, it may appeal to its neighbors to perform additional monitoring of the area. Thus, this scenario may result in collecting a relatively large number of optical frames containing information about a target.

If a target image is acquired at a low resolution (due to high altitude flight or absence of zoom), a single frame-based detection provides poor results. However, if a set of frames containing information about the same target are available, the detection performance may be improved considerably due to use of a super-resolution (SR) technique.

We also explore data fusion techniques for images with a sufficient resolution. For this we use a score-level data fusion technique.

3.3.1. Image-level data fusion for improved detection

In our swarmed UAV system, when a moderate amount of scene motion exists between frames, low-resolution images can be fused to yield an image of a higher resolution compared to any original low resolution frames. The literature describes a variety of approaches that exploit SR techniques.[11] We implement a modified version of the algorithm proposed by Hardie et al.[12] This technique is fast and well suited for our ATR purpose.

To generate a high-resolution (HR) image, low-resolution (LR) images are first registered relative to a specific reference frame. We involve a two step procedure for automatic registration of frames. In the first step, we use optical flow to extract similar features in different frames and then apply purely geometric matching procedure.[13] In the second step, sub-pixel image registration is achieved by a gradient-based registration technique[12] and a non-uniform liner interpolation method[14] is used to generate high-resolution grid. The results demonstrating the effect of SR on detection performance are provided in Sec. 4.3.

3.3.2. Score-level data fusion for improved detection

Consider now the case when frames contain targets represented by a large number of pixels sufficient for successful detection. However, the images may be of poor quality. We apply a two-step data fusion procedure at the score-level and demonstrate improved detection performance.

First, the fames containing information about the same target are registered with respect to a reference frame using the control-point based image registration method.[13] Then the single-frame object detector described in Sec. 3.1 is applied to the combined registered overlapping image areas. The scores produced by the stage classifiers (2) applied to different image frames are combined to generate the final detection results. Kittler et al.[15] summarize on classes of combination strategies at the score level. We choose Sum Rule and Majority Vote Rule because of their simplicity and computational efficiency.

The Sum Rule is described by the following equation:

$$C(x_1)_{fused} = \begin{cases} 1 & if\ \frac{1}{N}\sum_{n=1}^{N}\sum_j h_j(x_n) > T \\ 0 & \quad otherwise, \end{cases} \tag{9}$$

where x_1 is the specified reference image frame and x_n is a sub-window of image n among the N $(N \geq 2)$ frames of registered images.

The Majority Vote Rule is given by:

$$C(x_n) = \begin{cases} 1 & if \sum_j h_j(x_n) > T \\ 0 & otherwise \end{cases} \tag{10}$$

$$C(x_1)_{fused} = \begin{cases} 1 & if \ \sum_n C(x_n) > \frac{N}{2} \\ 0 & otherwise, \end{cases} \tag{11}$$

where x_1 is the specified reference image frame and x_n is a sub-window of image n among the N $(N \geq 3)$ frames of registered images.

3.4. Data fusion for improved recognition

In this section we describe a multivariate Bessel K form for improved recognition.

The multivariate Bessel K forms can be formed as a mixture of Gaussian variables, where the mixing variable is a scaled Gamma distributed random variable with parameters p and c. Multivariate Bessel K forms are a special case of a larger family, namely, the generalized hyperbolic distributed family (see Barndorff-Nielson et al[16] for details). Denote by $\mathbf{v}$ a d-dimensional random vector following Guassian distribution with zero mean and identity covariant matrix. Let z be a random variable following Gamma distribution with parameters p and c. Form

$$\mathbf{x} = \sqrt{z}\Gamma^{\frac{1}{2}}\mathbf{v}.$$

Then $\mathbf{x}$ is a d-dimensional random vector following Bessel K distribution with parameters p, c and Γ. The probability density function of $\mathbf{x}$ is given by

$$f_K(\mathbf{x}; p, c, \Gamma) = \frac{2}{Z_M(p,c)} \left(\sqrt{\mathbf{q(x)}}\right)^{p-\frac{d}{2}} K_{(p-0.5)}\left(\sqrt{\frac{2}{c}\mathbf{q(x)}}\right), \tag{12}$$

where $\mathbf{q(x)} = \mathbf{x}^T\Gamma^{-1}\mathbf{x}$ and $Z_M(p,c)$ is the normalization given by

$$Z_M(p,c) = \pi^{\frac{d}{2}}\Gamma(p)(2c)^{0.5p+0.25d}.$$

When $d = 1$, (12) reduces to (3).

As illustrated in Fig. 7, the pair of images $I(\alpha_1)$ and $I(\alpha_2)$ are taken from the same object but at different poses. They can be jointly represented by $3J$ sets of parameters

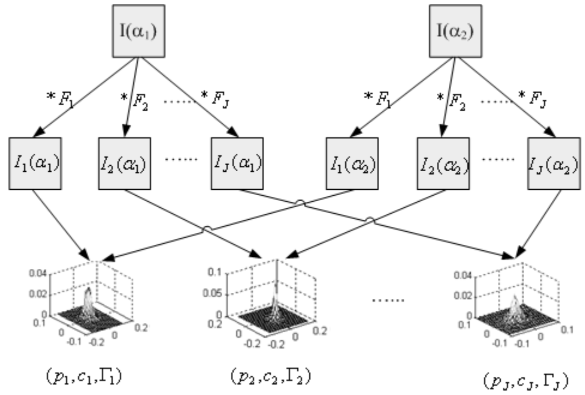


Figure 7. Representation of a pair of images $I(\alpha_1)$ and $I(\alpha_2)$ by $3J$ Bessel parameters.

To estimate the parameters p, c and Γ, we first find the mean and covariance matrix as

$$\hat{\mu} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i \quad \text{and} \quad \hat{\Gamma} = \frac{\hat{C}}{\left(\det \hat{C}\right)^{\frac{1}{d}}},$$

where N is the sample size and $\hat{C} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \hat{\mu})(\mathbf{x}_i - \hat{\mu})^T$. Then we generate a new random vector $\mathbf{y}_i = \hat{\Gamma}^{-\frac{1}{2}}(\mathbf{x}_i - \hat{\mu})$, which follows d-dimensional Bessel K distribution with zero mean, identity covariance matrix, the shape parameter p and the scale parameter c. The marginal distribution of $y_{ki}, \ k = 1, \cdots, d$ follows univariate Bessel K form. So we can estimate p and c as $\hat{p} = \frac{1}{d}\sum_{k=1}^{d}\hat{p}_k$ and $\hat{c} = \frac{1}{d}\sum_{k=1}^{d}\hat{c}_k$, where $\hat{p}_k$ and $\hat{c}_k$ are the estimates from the k^{th} projection.

We use the K measure to qualify the distance between two pairs of images.

4. NUMERICAL RESULTS

In this section, we use the simulated dataset described in Sec. 2 to evaluate the influence of environmental and camera effects on detection and recognition performance. We further analyze detection and recognition results from a single and multiple frames.

Both detection and recognition algorithms operate in two modes: training and testing. To train our detection algorithm, we compile sets of positive and negative training images. The set of positive training images contains cropped projections of single targets at different orientations and elevations. The set of negative images is composed of cropped images of non-targets. To test the detection algorithm, that is, to evaluate the detection performance, we used the 3-D ATR Training Tool. A number of 3-D scenes were generated by placing targets and non-targets on a background (grass, sand, etc.). The projections of scenes acquired at different orientations and elevations were treated as testing images.

To train and test recognition algorithm, we used cropped images of targets. The undistorted images of all targets at orientations $0, 15, 30, \cdots, 345$ and elevation 15 degree form the training set. The remaining undistorted images are used to evaluate the recognition performance for single frame and multi-frame cases. Distorted images are used to test the influence of environmental and camera effects on recognition performance. To process data, we used a bank of 38 filters including Gaussian filters, Laplacian of Gaussian filters and Gabor filters.

4.1. Influence of Environmental and Camera Effects on Detection Performance

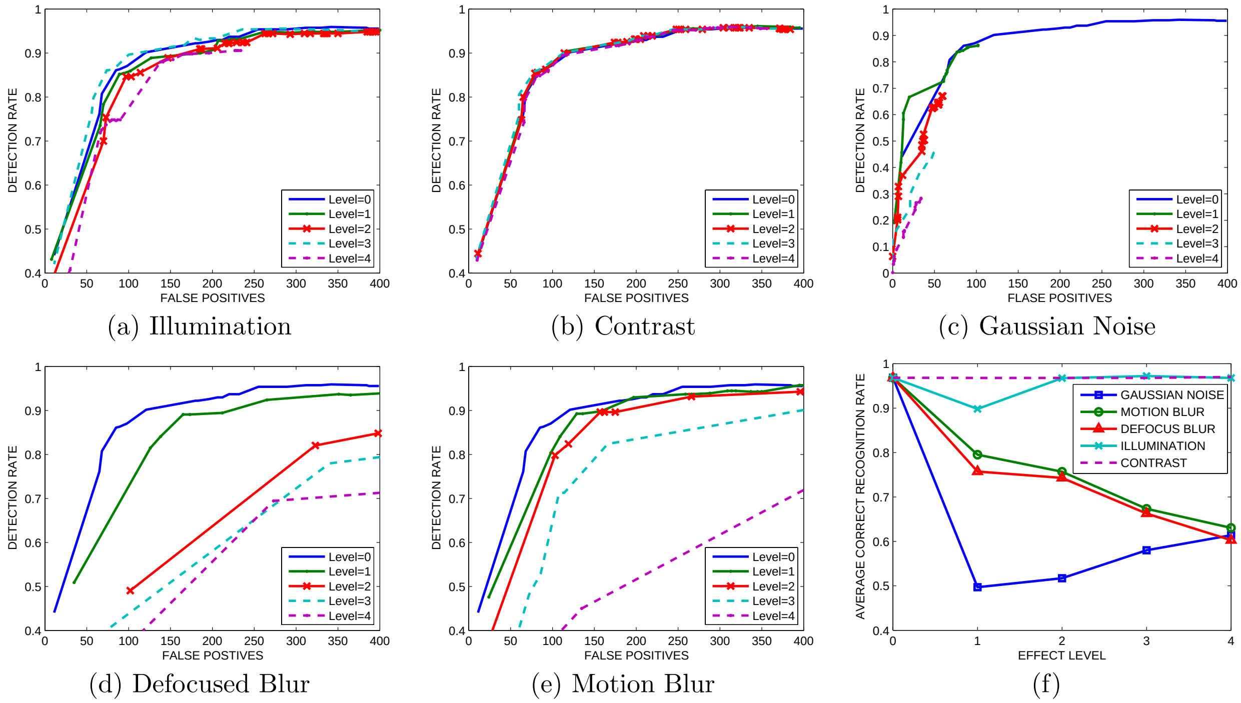
In our evaluations of detection capabilities of Haar feature-based algorithm we use a dataset consisting of 277 grayscale images generated using ATR training tool. These images contain 440 targets parameterized by varying type, location, orientation, and camera elevation angle. We further generate 5 distorted images per each original "clean" image. The distorted images include illumination variations, contrast variations, Gaussian noise, defocus blur and motion blur. The effects are generated individually and tested separately. For each effect, the distortion is increased from low level to high level in which level 0 corresponds to the case when no distortion is imposed. For the illumination changes, levels below 0 indicate darker images and levels above 0 indicate brighter images compared to the original images. The results of detection performance evaluation are shown in Fig. 8(a)-(e).

From the ROC curves, we can conclude that illumination and contrast variations do not affect the detection performance significantly. This is because the Haar feature-based detector implements a light correction procedure prior stage classification. To be more specific, prior to stage classification all test windows are normalized to minimize the effect of different lighting conditions. The procedure of normalization is as follows:

$$I^{-}(x, y) = \frac{I(x, y) - \mu}{c\sigma}, \ c \,\epsilon R^{+}, \tag{13}$$

where $I(x, y)$ is the pixel value within the sub-window during detection scanning. μ and σ are the mean and the standard deviation of $I(x, y)$.

On other effects, both blur and noise degrade detection performance: the number of false alarms in detection increases with increased level of effects.



(a) Illumination (b) Contrast (c) Gaussian Noise

(d) Defocused Blur (e) Motion Blur (f)

Figure 8. (a)-(e) Detection and (f) Recognition performance as functions of various environmental and camera effects.

4.2. Influence of Environmental and Camera Effects on Recognition Performance

Each effect is tested individually. To evaluate the influence of an environmental or camera effect on recognition performance, we generate a number of distorted images parameterized by varying distortion level: from lowest level to highest level. The average correct recognition rate as a function of distortion level parameterized by various effects are shown in Fig. 8(f). Level 0 corresponds to the case when no distortion is added.

From Fig. 8(f) we conclude that the average recognition performance decreases when distortion level increases, except the illumination and contrast effects. These two effects are compensated prior to evaluation of the recognition performance (see the image normalization step in (13)).

4.3. Detection Performance: Single and Multi-frames cases

To evaluate improvement in detection performance due to involvement of multiple frames, we consider two fusion scenarios: fusion at image level and fusion at the score-level. The first scenario will be beneficial in the case of low resolution images or partially occluded images. The other scenario results in performance improvement under a broad range of conditions.

4.3.1. Detection Performance: Image-level Data Fusion

We use superresolution[12] as a method of fusing data at the image level. High-resolution (HR) images are constructed from artificially-generated, low-resolution (LR) images. To generate the HR image, the LR images are registered relative to a specific frame of reference. Following this registration, available LR pixels are used to sparsely populate a HR image grid, and non-uniform interpolation techniques are applied to the remaining gridpoints to generate an estimate of the HR image.

We train two detectors on the same set of positive images but at different image resolutions. The high-resolution detector is trained on positive samples of 30 × 30 pixels. The low-resolution detector is trained on positive samples of 15 × 15 pixels. The size of positive samples is determined by the minimum size of targets in images submitted for detection. The summary of parameters is provided in Table 1.

Table 1. Summary on High Resolution and Low Resolution Detectors used in our experiments

	LR Detector	HR Detector
Number of Postive samples	180	180
Number of Negative samples	500	500
Stages	11	13
$Width \times Height$/pixels	15×15	30×30

The two detectors were tested using 3 datasets: low-resolution(LR), high-resolution(HR) and super-resolved (SR) test datasets. LR images are generated by randomly translating and rotating images in HR dataset and then downsampling by a factor of 2 in each dimension. SR images are generated using K LR frames. The performances of the detector trained on HR images is tested on a number of SR datasets constructed from $K = 2,\ 4$ and 8 frames of LR images. In the ideal case of perfect half-pixel displacement, one needs only 4 LR images to obtain a HR estimate. However, since displacements are random (both translations and rotations), a larger number of LR images is needed for accurate reconstruction.

The testing results are shown in Fig. 9. The performance of LR detector on LR test database is poor indicating that LR images lack important details for successful recognition. In this case, HR detector considerably outperforms SR detector.

4.3.2. Detection Performance: Score-level Data Fusion

For score-level data fusion, we employ Sum Rule (9) to performance data fusion. Two image sequences of the same scene containing several targets are captured from different view angles and distances by the ATR Training Tool to simulate surveillance tasks performed by two independent UAVs. Each sequence consists of 40 frames of images and 160 targets. The testing results are shown in Fig. 10. The experiments show the score-level data fusion technique can reduce false alarms and keep high detection rate.

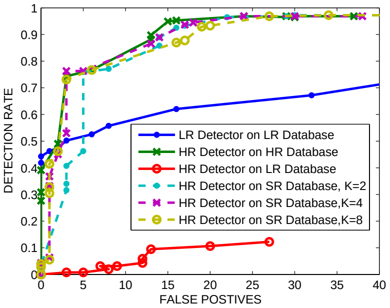


Figure 9. Image-level Data Fusion Results.

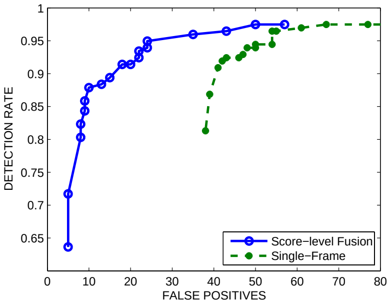


Figure 10. Score-level Data Fusion Results.

4.4. Recognition Performance: Single and Multi-frames cases

In this experiment, we involve only the original images generated using the ATR Training Tool. We assume that the relative rotation angle between the test image pair is known. Two different relative angles are tested, 5- and 10-degree. For each angle, there are 24 sets of multivariate Bessel K parameters describing each object.

Since images are generated at 4 different elevation angles, the total number of testing pairs is $4 \times 4 \times 72$ per target. Table 2 summarizes the results of testing of the multivariate Bessel K recognition algorithm. The correct recognition and error rates are presented in the form of a confusion matrix for single and two-frame cases (with 5 and 10 degree relative orientation). Note that multivariate Bessel K forms result in considerably improved performance when the relative orientation between two images is 10 degrees, that is, when data are less correlated compared to the case with the relative orientation of 5 degree.

Table 2. Recognition Performance using Single and Two images.

	Single	Two (5 degree)	Two (10 degree)
Confusion Matrix	$\begin{bmatrix} 0.9545 & 0 & 0.0114 \\ 0 & 0.9811 & 0 \\ 0.0455 & 0.0189 & 0.9886 \end{bmatrix}$	$\begin{bmatrix} 0.9931 & 0.0642 & 0 \\ 0 & 0.9358 & 0 \\ 0.0069 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.9991 & 0.0069 & 0 \\ 0.0009 & 0.9931 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

5. CONCLUSIONS

In this work, we staged a potential scenario for optical data acquisition by a UAV network. We analyzed the influence of environmental and camera effects on detection and recognition performance. The detection algorithm relies on Haar-like features. The recognition algorithm is based on estimation of Bessel K forms. We further implemented and tested a number of data fusion schemes for improved detection and recognition. The schemes include superresolution and score-level fusion for detection and multivariate Bessel K forms for recognition.

ACKNOWLEDGMENTS

This work was supported by Augusta Systems in support of phase-II Navy STTR contract number N68335-05-C-0303.

REFERENCES

1. D. M. Hart and P. A. Craid-Hart, "Reducing swarming theory to practice for UAV control," *Proc. IEEE Aerospace Conf.* , pp. 3050–3063, Mar. 2004.
2. R. S. Blum, S. A. Kassam, and H. V. Poor, "Distributed detection with multiple sensors I: Advanced topics," *Proc. IEEE* **85**, pp. 64–79, Jan. 1997.
3. M. B. T.K. Leung and P. Perona, "Finding faces in cluttered scenes using random labeled graph matching," *Proc. Fifth IEEE ICCV* , pp. 637–644, 1995.
4. P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *IEEE CVPR* , pp. 1–9, 2001.
5. J. Aggarwal, *Multisensor Fusion For Computer Vision*, Springer-Verlag, 1993.
6. A. E. Savakis and H. J. Trussell, "Blur identification by residual spectral matching," *IEEE Transactions on Image Processing* **2**, pp. 141–151, April 1993.
7. A. Srivastava, X. Liu, and U. Grenander, "Universal analytical forms for modeling image probabilities," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**, pp. 1200–1214, Sept. 2002.
8. R. Schapire and Y. Singer, "Improving boosting algorithms using confidence-rated predictions," 1999.
9. B. Jahne, H. Haubecker, and P. Geibler, *Handbook of Computer Vision and Applications*, Academic, San Diego, CA, 1999.
10. S. Kullback, *Information Theory and Statistics*, Dover Publications, New York, 1997.
11. M. P. S. Park and M. Kang, "Super-resolution image reconstruction: (a) technical overview," *IEEE Signal Processing Magazine* **20**, pp. 21–36, May 2003.
12. R. H. M. Alam, J. Bognar and B. Yasuda, "Infrared image registration and high-resolution reconstruction using multiple translationally shifted aliased video frames," *IEEE Trans. on Instrumentation and Measurement* **49**, pp. 915–923, Oct. 2000.
13. C. K. L. Fonseca, G.Hewer and B. Manjunath, "Registration and fusion of multispectral images using a new control point assessment method derived from optical flow ideas," *Proc. SPIE Conference* , April 1999.
14. C. B. Barber, D. P. Dobkin, and H. T. Huhdanpaa, "The quickhull algorithm of convex hualls," *ACM Transactions on Mathematical Software* **22**, pp. 469–483, Dec. 1996.
15. R. P. D. Josef Kittler, Mohamad Hatef and J. Matas, "On combining classifiers," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **20**, pp. 226–239, March 1998.
16. O. Barndorff-Nielsen, J. Kent, and M. Sorensen, "Normal variance-mean mixtures and z distributions," *Int'l Statistical Rev.* **50**, pp. 145–159, 1982.