10.11

Statistical Models of Targets and Clutter for Use in Bayesian Object Recognition

Anuj Srivastava
Florida State University

Michael I. Miller
Johns Hopkins University

Ulf Grenander
Brown University

1 Introduction

When human beings look at camera images of known objects, such as a table, a chair, a human face, or a car, we recognize them immediately. For example, Fig. 1 shows several images of tanks in different backgrounds. Even if these images are corrupted or noisy, low-contrast, or have partial obscuration, we can still recognize tanks in these images. This observation points to an important fact: *The human visual recognition system is an awesome system with extraordinary processing power.* Can we design an automated system: equipped with cameras, computers, databases and algorithms, to achieve a similar performance in object recognition? So far there has been only a limited success in this area. In this chapter, we analyze this issue in the context of a very specific problem in automated image analysis, called automated target recognition (ATR). By restricting to ATR we can use the additional contextual information available, in designing ATR algorithms. In a general ATR situation, a number of remote sensors [cameras, radars, ladars, three-dimensional (3D) scanners, etc.] observe a scene containing a number of dynamic or stationary targets (a more detailed introduction can be found in [6]). These sensors produce observations, in the form of images or signals, which are then analyzed by computer algorithms to *detect, track,* and *recognize* the targets of interest in that scene. Our goal is to derive ATR algorithms and analyze them for their performance. Our approach relies on two main building blocks: (a) efficient mathematic representations of scenes containing targets, and (b) efficient algorithms for inferences on these representation spaces. This chapter describes these two steps to ATR.

One fundamental issue in ATR is the following. Consider a normal hand-held camera taking pictures of a car. Depending on the relative orientation between the camera and the car, and the distance between them, the car appears vastly different in different pictures. The possible variability in relative orientation, also called the *pose,* causes a tremendous variability in the profiles of the targets as seen by a camera, or a sensor in general. This fact underlines one difficulty in the design of a completely automated algorithm of target recognition. How to mathematically model the variability in the sensor outputs due to the variability in target pose? The

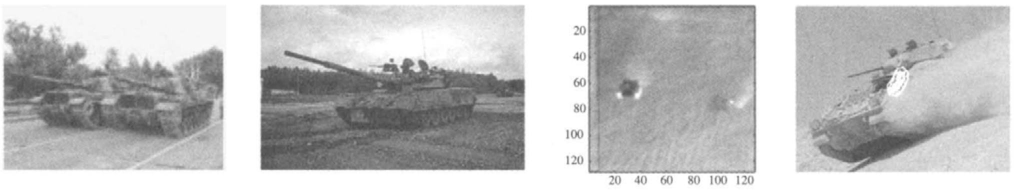

FIGURE 1 Examples of tank images showing partially obscured targets or targets in low-contrast images.

task is further complicated by relative motion between the sensors and the targets, imperfections in sensor operations, atmospheric effects such as illumination or temperature, and the presence of structured clutter that can potentially obscure the targets.

We will use elements of *deformable template theory* to mathematically model the variations in target pose. For each possible object, we define a template [using 3D computer-assisted design (CAD) models and other descriptors] of standard size, pose and location. All occurrences of a target in a scene can then be represented by scaling, rotating and translating its template appropriately. All possible scales, rotations, and translations form sets that have interesting geometrical properties. As described later, they have a *group* structure. In short, these transformations are used to transform the templates to match the occurrence of targets in a scene. The objects and the scenes containing them are 3D even though our observations of them are one-dimensional (1D) or two-dimensional (2D). Using the physics of the sensor operation, we will derive projections which transform 3D scenes into sensor outputs, thus, mathematically modeling the sensor operation. These operators can be deterministic or random with known probability distributions.

In view of several competing ATR approaches presented in recent years, it becomes important to develop a coherent framework for performance analysis. This analysis should include both prognostics (e.g., the best performance that can be achieved regardless of the algorithm) and diagnostics (e.g., the performance analysis of a given algorithm). Several authors have presented metrics for ATR performance analyses, although in limited frameworks [1, 7, 15, 18, 27]. A detailed review of current ATR approaches is also presented in the report [5], in the context of synthetic aperture radar (SAR) ATR. One advantage of the Bayesian framework is that it provides metrics and bounds for comparing algorithmic performance, both between each other and with the best that can be achieved.

Section 2 introduces the deformable template approach to represent the target variabilities while Section 3 defines statistical models for some commonly used sensors. Section 4 sets up a Bayesian framework to solve pose and location estimation and target recognition problems. Section 5 defines and computes MMSE estimates for the target pose and location, and Section 6 summarizes procedure for target recognition.

2 Statistical Models

2.1 Target Representations

Representation is an essential element of image understanding and target recognition. The generation of efficient models for representations of target shapes supporting recognition invariant to orientations and locations is crucial. Targets are observed at arbitrary positions and orientations, in highly variable environments. The variability in target pose, with respect to the sensor, is important because at different orientations the targets appear very different. Even the same target can appear completely different at two different orientations. Due to the nonlinear relationship between target orientation and image pixel values, the orientation parameter has to be modeled explicitly and estimated for target recognition. This task is complicated by relative motion between the sensors and the targets, imperfections in sensor operations, and the presence of clutter elements in the scene. Furthermore, different sensors capture widely different aspects of the target. A video captures the visible light reflection, radar captures the electromagnetic scattering, forward-looking infrared (FLIR) camera captures the thermodynamic profile, and so on. For these widely varying sensor outputs what should be chosen to represent the targets?

In recent years, a successful approach to target representations has been the **deformable template** theory. In this approach, the starting point is to select a standard template for each of the targets and then define a family of transformations to account for the variability associated with target occurrences.

1. **Templates:** Start by defining a set of target labels:

   $\mathcal{A} = \{\text{airplane, chair, car, lamp, table, jeep, truck, tank,} \ldots\}$.

   Each $\alpha \in \mathcal{A}$ denotes a particular target. For each $\alpha \in \mathcal{A}$, we define I^{α} to be a template associated with that target. It includes all the physical attributes of the target that are reflected in the sensor output including shape, size, material, surface reflectivity (or BRDF), and thermal profile. Clearly, the constituents of I^{α} depend upon the sensor(s) being used. For a visible spectrum video camera, I^{α} may consist of a finite element description of its surface, surface texture, and the colors. Shown in Fig. 2 are 3D renderings of sample

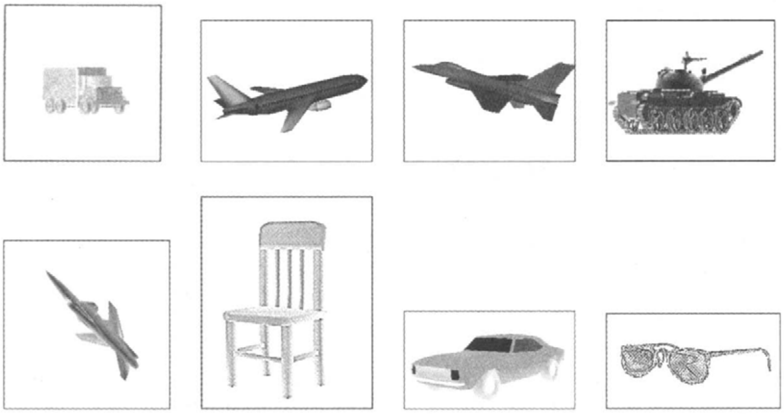

FIGURE 2 Templates for various targets.

target templates. In this case, each template consists of a set of polygonal patches covering the surface, the material description (texture and reflectivity), and surface colors.

2. **Transformations:** The targets when they appear in a scene do so at arbitrary positions, orientations, light conditions, and thermal profiles. The next issue is to account for this variability by defining a family of transformations, on the templates, to generate all possible occurrences of the targets. To understand the basic idea, consider this simple example from high-school geometry. We define two triangles to be **similar** if they have equal corresponding angles, for example the two triangles shown in Fig. 3. If we rotate, translate, and (uniformly) scale the left triangle appropriately, we will obtain the right triangle and vice versa. The transformation that takes one triangle to another is called the **similarity transformation.** The set of all possible similarity transformations, call it S, forms a *group*. A group is a set endowed with a group operation (denoted here by $\circ$, often called the product) such that for any two elements in the group their product also lies in the group. Additionally, there exists an identity element, e, such that its product with any element of the group does not change that element (please refer to [25] for more details). As an example, $\mathbb{R}^n$ is a group with vector-addition as the group operation and zero-vector as the identity element. Similarly, the set of $n \times n$ non-singular matrices is a group with matrix-multiplication as the group operation and the identity matrix as the identity element. The group structure is instrumental in defining compositions of the transformations: one transformation (s_1) applied after another transformation (s_2) have an equivalent effect of a third transformation (s_3) applied alone. The third transformation is a product of the first two, $s_3 = s_2 \circ s_1$.

Now we extend the same idea to more complicated objects and seek groups that model their variations.

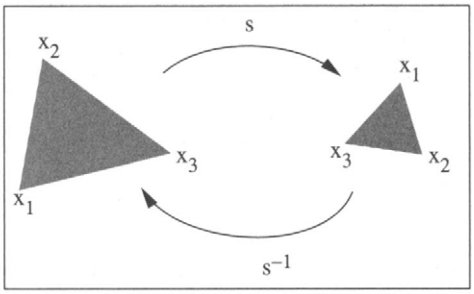

FIGURE 3 Two similar triangles in Euclidean geometry.

We need groups to rigidly rotate and translate 3D objects. Let O be a 3×3 matrix such that $OO^\dagger =$ identity ($\dagger$ denotes matrix transpose) and the determinant of O is 1. Then, for any point $x \in \mathbb{R}^3$ on an object, Ox is just a rotated version of x. O is called a rotation matrix and the set of all such rotation matrices is denoted by $SO(3)$, the *special orthogonal group* in three-dimensions. $SO(3)$ is a group with matrix multiplication as the group operation and 3×3 identity matrix as the identity element. If we fix an axis of rotation, as is the case for ground-based objects, then there is only one rotational freedom left. This rotation is modeled by 2×2 rotation matrices and their set is denoted by $SO(2)$. For translations, if we translate an object by a vector $p \in \mathbb{R}^3$, each point x on the object becomes $x + p$. The set of all possible translations in three-dimensions is the whole of $\mathbb{R}^3$. Similarly, if the translations are restricted to ground, then $\mathbb{R}^2$ is the *translation group*. More generally, in n-dimensional spaces, $SO(n)$ is the rotation group and $\mathbb{R}^n$ is the translation group. To accomplish both rotation and translation, we utilize a combination of $SO(n)$ and $\mathbb{R}^n$. Let U be a $(n+1) \times (n+1)$ matrix such that

$$\begin{bmatrix} O & p \\ 0 & 1 \end{bmatrix}, \quad \text{where } O \in SO(n) \text{ and } p \in \mathbb{R}^n.$$

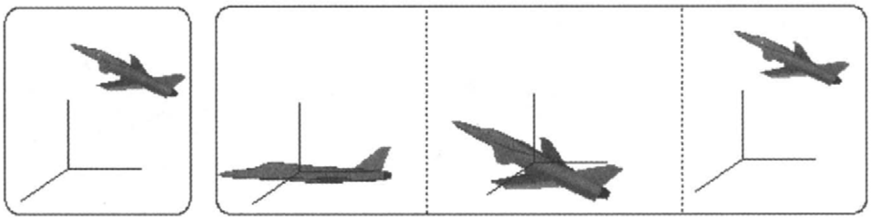

FIGURE 4 **Left panel:** an airplane at an arbitrary orientation and position. **Right panel:** The airplane template rotated and translated from an initial pose and location to match the pose and location in the left panel.

For a vector $x \in \mathbb{R}^n$ define an augmented $(n+1)$-vector $x_1 = \left[\begin{smallmatrix} x \\ 1 \end{smallmatrix}\right]$. Then, the first n entries of the vector Ux_1 represent a rotated and translated version of x. The set of all such matrices U is denoted by $SE(n)$, the *special Euclidean group*. $SE(n)$ is a group with matrix multiplication as group operation and the $(n+1) \times (n+1)$ identity matrix as its identity element.

Depending on the specific problem, the group of transformations S can be $\mathbb{R}^n$, $SO(n)$, $SE(n)$ or Cartesian products of them. For an element $s \in S$, let sI^α denote the target template I^α transformed by the element s. For example if $S = SE(3)$, then sI^{air} is the airplane template rotated and translated according to s, as shown in Fig. 4. The set of all possible transformed of a target α is given by

$$\mathcal{O}_\alpha = \{sI^\alpha, s \in S\}.$$

$\mathcal{O}_\alpha$ is called an **orbit** associated with the target α. Then, S is said to **act on** $\mathcal{O}_\alpha$ (on the left) because it satisfies the following two conditions:

(a) if e is the identity element of S, then

$$eI^\alpha = I^\alpha, \quad \text{for all } \alpha \in \mathcal{A}.$$

(b) if $s_1, s_2 \in S$, then

$$s_2(s_1 I^\alpha) = (s_1 \circ s_2)I^\alpha, \quad \text{for all } \alpha \in \mathcal{A}.$$

The strength of a deformable template approach comes from the fact that all targets' occurrences can be modeled using appropriate transformations on appropriate templates. Therefore, given an observed image of a target the task reduces to finding a template and a transformation best matches the given image image.

It must be noted that the variability in targets is not caused only by arbitrary orientations and positions. There are other factors such as light conditions, targets' surface temperatures, texture variations and their operational status. These factors can also be incorporated through more general transformations which are much higher dimensional than rigid rotation and translation. As an example, the thermodynamic variability in target surfaces as observed by FLIR cameras is modeled and estimated as a high-dimensional scalar field in [4].

2.2 Sensor Modeling

So far we have considered 3D target templates and a set of transformations on them to describe their occurrences in arbitrary scenes. The observations are however, in general, restricted to 1D or 2D arrays of numbers as generated by the sensors. Therefore, for a better understanding of images we have to build detailed models for these sensors. In these models the physics of sensor operation plays an important role because different sensors may produce very different pictures of the same scene. Microwave radars generate very different "pictures" of the target than second-generation FLIR cameras or video cameras.

In most sensors, imaging is essentially a projective mechanism operating by accumulating responses from the scene elements which project to the same pixel in the image. Mathematically, we will model the mechanism that maps the scene to some observation space $\mathcal{I}^{\mathcal{D}}$. In most cases $\mathcal{I}^{\mathcal{D}} = \mathbb{R}^d$ or $\mathbb{C}^d$ for some fixed number d. This mechanism can either be deterministic or random, and constitutes a mapping T by which a transformed 3D target, sI^α, appears to the observer as an image $I^D \in \mathcal{I}^{\mathcal{D}}$. In addition to T, a sensor may also generate random noise image, w, which is assumed to be additive. Then, the observation is modeled by

$$I^D = TsI^\alpha + w \in \mathcal{I}^{\mathcal{D}}. \tag{1}$$

In the ATR context, we must abstract this T in some generality to accommodate various sensors. The particular transformation T and the noise properties are determined by the sensor. For example, in case of an infrared camera, TsI^α is the mean field of a Poisson process for which the additive noise is not appropriate, see for example the discussion in [20]. It must be noted that accurate analytic expressions for T may not be available in all situations, but very often high-quality simulation experiment (using special hardware) can be used to sample T at some predefined target orientations. For modeling radar returns, the XPATCH simulator has been widely used while for FLIR cameras, PRISM is often used. Similarly, visible spectrum images can be simulated on high-performance graphics workstations.

I^D may have multiple components corresponding to multiple sensors observing the scene simultaneously: $I^D \equiv (I_1^D, I_2^D, \ldots)$. Since the images are random, they are characterized via a statistical transition law, called the *likelihood function* $P(\cdot|\cdot)$: $\mathcal{I}^D \times (S, \mathcal{A}) \rightarrow \mathbb{R}_+$, summarizing completely the mapping from the target α at transformation s to the output I^D. Some of the sensors used frequently in ATR applications are:

1. *Video imager:* A video sensor provides two-dimensional high-resolution real-valued images of rigid targets sampled on a lattice of certain size, $I^D \equiv \{I^D(y), y \in Y \equiv \{1, 2, \ldots, \}^2, I^D(y) \in \mathbb{R}\}$. The images are assumed to result from an orthographic or a perspective projection of a 3D surface intensity on to the camera focal plane, as shown in Fig. 5. The left panels depict an orthographic projection scheme utilized in pose estimation, when the target position is assumed known. The right panels illustrate the perspective projection system utilized when both the target pose and location are unknown. The lower left and right panels show TsI^{tank} for orthographic and perspective systems, respectively. It is assumed that the reflected light intensity is high so that $I^D \equiv \{I^D(y), y \in Y\}$ is taken to be a Gaussian random field, with the mean field given by TsI^α. Shown in the left panel of Fig. 6 is an example of a simulated noisy video image of a truck.
2. *High-range resolution radar:* A high-range resolution (HRR) radar provides 1D range profiles of rigid targets (see [12] for a reference). The transmitted electromagnetic pulses directed at a target are received back at the receiver, at times proportional to the distance traveled, representing the superposition of the echoes from all the reflectors in a bin along the range direction. The received signal is processed via a matched filter to generate a 1D magnitude profile versus range, $I^D \equiv \{I^D(y), y = 1, 2, \ldots, I^D(y) \in \mathbb{R}\}$. The middle panel of Fig. 6 shows a range profile of a T62 tank at certain orientation, for a carrier frequency in the millimeter-wavè region.
3. *FLIR:* A second-generation FLIR camera captures the thermodynamic profile of a target body via CCD detectors (see Snyder et al. [14, 20]). The measured data $I^D \equiv \{I^D(y), y \in \{1, 2, \ldots, \}^2, I^D(y) \in \mathbb{R}\}$ are each assumed to be Poisson with means given by the corresponding pixels of the perspective projection of the target's 3D thermodynamic state. The right panel of Fig. 6 shows a tank's thermal profile, when projected and blurred by the point-spread function of the camera, provides an infrared image.

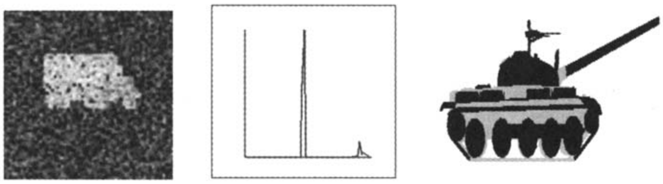

FIGURE 6 Simulated sample images obtained from different sensors: video imagery (SGI), high-range resolution range imagery (XPATCH), and forward-looking infrared imagery (PRISM).

2.3 Clutter Models

In addition to the targets of interest, a scene may contain several other objects that are exhibited in images of this scene. Such objects are called clutter objects and one needs to develop models for pixels falling on clutter objects, to reach a full statistical inference. Given the tremendous variability associated with objects, detailed (e.g., 3D deformable templates) models are not feasible for "all possible objects." Therefore, one seeks a balance by designing low-level, coarse representations that are tractable and yet capture significant image variation.

Mumford et al. [17] and others have discovered certain invariant patterns among large databases of images. They have established that image statistics, under a variety of representations, point to non-Gaussian behavior. For example, a popular mechanism of decomposing images locally—in space and frequency—using wavelet transforms leads to coefficients that are quite non-Gaussian. The histograms display heavy tails and sharp cusps at the median. It is imperative that the probability models used in image analysis incorporate such observed phenomena.

In recent papers Grenander et al. [10, 23] have proposed an analytic form, called *Bessel K form*, to model the observed 1D and 2D histograms of images. The forms are parametric, and

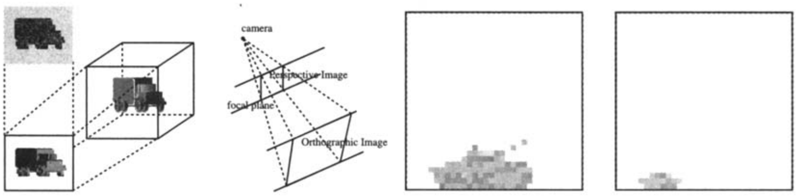

FIGURE 5 Left to right: orthographic projection model, perspective projection system, an orthographic image, and a perspective image.

hence efficient, and very closely match the observed behavior of non-Gaussianity.

Consider the following framework for modeling a clutter image I. Given a bank of filters $\{F^{(j)}, j = 1, 2, \ldots, J\}$, we compute, for each filter $F^{(j)}$, a filtered image $I^{(j)} = I * F^{(j)}$, where $*$ denotes the 2D convolution operation. As an example, a Gabor filter is a bandpass filter with a Gaussian kernel centered around a specific wavenumber (see [13] for details). For a rotation $\theta \in S^1$ on the unit circle, a Gabor filter is given by $R_\theta(z) \circ \left(\exp(-\frac{1}{2\sigma^2}(z(1)^2 + z(2)^2))\exp(-j\frac{2\pi z(1)}{\sigma})\right)$, where σ denotes the resolution (scale) associated with the filter and R_θ is the 2×2 rotation matrix. Another commonly used filter is the Laplacian Gaussian filter whose operation on I is given by $(G * \Delta)I$ where G is a Gaussian kernel and Δ is the Laplacian operator. In addition, one can use a wide variety of filters: neighborhood operators, steerable filters, interpolation filters, and so on. Each filter selects and isolates certain features present in the original image (Fig. 7).

We are interested in modeling the probability density of a random variable $I^{(j)}(z)$ where z is any pixel location in the image. As derived in [10], a promising model is: for $p > 0$, $c > 0$,

$$f(x; p, c) = \frac{1}{Z(p, c)} |x|^{p-0.5} K_{(p-0.5)}\left(\sqrt{\frac{2}{c}}|x|\right), \quad (2)$$

where K is the modified Bessel function and Z is the normalizing constant given by $Z(p, c) = \sqrt{\pi}\Gamma(p)(2c)^{0.5p+0.25}$. Let $\mathcal{D}$ be the space of all such densities: $\mathcal{D} = \{f(x;p,c) | p > 0, c > 0\}$. We refer to the elements of $\mathcal{D}$ as **Bessel K forms** and the parameters (p, c) as **Bessel parameters.** These parameters can be estimated directly from the observed data according to: $\hat{p} = 3/(\text{kurtosis-3})$ and $\hat{c} = \text{variance}/\hat{p}$. The elements of $\mathcal{D}$ have the following properties:

1. They are symmetric and unimodal for the mode at zero. For $p = 1$, $f(x;p, c)$ is the density of a double exponential. In general, it is the p-th convolution power (for any $p > 0$) of a double exponential density. Therefore, it is unimodal with the mode at $x = 0$. For the same reason, it is symmetric around zero.
2. The Bessel K forms are leptokurtic (the tails are heavier as compared to a normal curve with the same variance).
3. A Bessel K form is a specific kind of normal variance-mean mixture where the mixing variable is scaled Gamma with parameters p and c. It becomes a special case of a larger family of normal variance mixtures (as described by Barndorff-Nielson et al. [2]).
4. The family of Bessel K forms is infinitely divisible (i.e., any random variable in this family can be written as a sum of two independent random variables from this family). However, if I_1 and I_2 are independent with densities $f(x;p_1, c_1)$ and $f(x;p_2, c_2)$, respectively, with $c_1 \neq c_2$, the density of $a_1I_1 + a_2I_2$ ($a_1, a_2 \in \mathbb{R}$) may not be a Bessel K form.

Now that we have presented models for targets, sensors, and background clutter, we turn our attention to a Bayesian framework for target recognition from images.

3 Bayesian Framework

To analyze observed images and to set up estimation problems we will use the classic Bayesian framework.

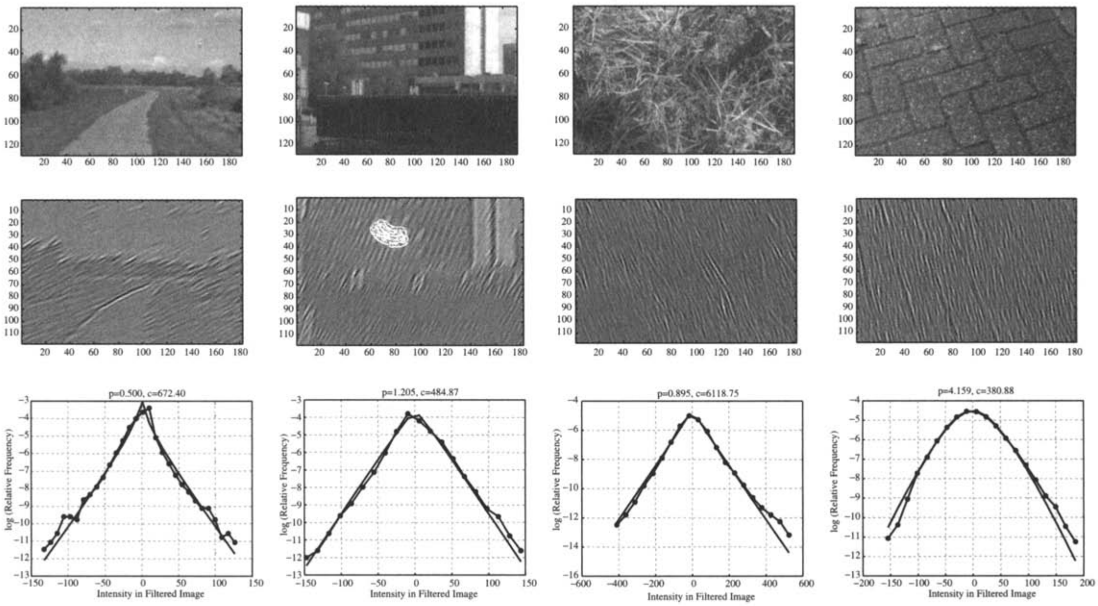

FIGURE 7 Images (**top panels**), their Gabor components (**middle panels**), and the marginal densities (**bottom panels**). The observed densities are drawn in marked lines and the estimated Bessel K forms are drawn in plain lines.

First, we need to explain the idea of probability densities on the nonlinear spaces, such as the transformation group S. A probability density function is defined as the derivative of a probability distribution function. For probabilities on $\mathbb{R}^n$, this derivative is with respect to the infinitesimal volume element in $\mathbb{R}^n$: $dx = dx_1 dx_2 \ldots dx_n$. On $SO(n)$, the volume element has a different form since $SO(n)$ is not a vector space. The derivatives of functions are evaluated with respect to an infinitesimal volume element, which we will denote by $\gamma(dO)$ (please refer to [3] for a description of this volume element, also called *Haar measure*). The product of the volume elements on $SO(n)$ and $\mathbb{R}^n$ provides a volume element on $SE(n)$. Note that just like $\int_{\mathbb{R}^n} f(x)dx$, the integration of a function on any set is defined with respect to the volume element of that set.

Now to model the uncertainty in associating an observed image to a particular template (indexed by α) and a particular transformation (denoted by s), we derive a posterior density on these unknowns. The posterior density is the product of the prior probability density on the unknowns and the likelihood of the data according to

$$P(s, \alpha | I^D) = \frac{1}{P(I^D)} P(s, \alpha) P(I^D | s, \alpha), \quad s \in S, \ \alpha \in \mathcal{A}.$$

The prior density $P(s, \alpha)$ incorporates our prior knowledge on finding a target α, at the pose and location dictated by the transformation s, in the scene. For example, in case of moving targets, the knowledge of target location may imply a higher probability of there being a future target presence in certain areas and low probability in others. The likelihood function $P(I^D | s, \alpha)$ quantifies the probability that a target α at the pose and location resulting from the transformation s will give rise to the observed image I^D. It is derived from the physical characteristics of the sensor map T, the statistics of the sensor noise, and the clutter model. As an example, for the video sensor described earlier the likelihood function takes the form,

$$P(I^D | s, \alpha) = P(I^D_{in} | s, \alpha) P(I^D_{out} | s, \alpha),$$

where

$$P(I^D_{in} | s, \alpha) = \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(\frac{-1}{2\sigma^2} \| I^D_{in} - TsI^\alpha \|^2 \right),$$

and

$$P(I^D_{out} | s, \alpha) = P(I^D_{out}) = \prod_z f(I^D_{out}(z); p, c).$$

I^D_{in} consists of image pixels that fall inside the ideal image TsI^α, while I^D_{out} consists of pixels falling on the background clutter. The resulting posterior includes all the information we have for target recognition.

Having obtained the posterior density we will generate the classic estimators such as maximum a posterior probability (MAP), minimum mean squared error (MMSE), minimum absolute error (MAE), and entropy-based estimators. In particular, we will seek MMSE estimators for the transformation, s, and MAP estimation for the target type, α. Along with the estimators, we will also compute quantities which represent errors in estimation and impose a lower bound on these errors.

4 Pose Location Estimation and Performance

In view of the nonlinear relationship between the template transformations and the observed images, we have to estimate the transformations explicitly. As a first step, we restrict our task to estimating the appropriate transformation from a given image I^D and for a fixed target hypothesis α. This α can be any index from our label set $\mathcal{A}$. Given the pose and location estimates for each α, we will seek the α that best matches the observed image in the next section.

A major difficulty being faced in an ATR setting is that, unlike the more classic estimation results, the parameter spaces of the scene of targets are nonlinear manifolds, with a group operation that is not necessarily addition. These groups can have a curved geometry, meaning that they may not be vector spaces. As an example, if we add two rotation matrices, then resultant is not a rotation matrix, so $SO(n)$ is said to have a curved geometry. In addition, the sensor outputs result from a sequence of nonlinear transformations on the scenes, including projective transformation, occlusion, ray-tracing, and so forth. Therefore, we cannot inherit the more direct results obtained for estimators in Euclidean spaces associated with additive Gaussian channels.

To illustrate, we isolate and focus on estimating the target orientation, represented by special orthogonal group $SO(n)$, since it is a group with curved geometry. We will define an optimal estimator, called the *Hilbert-Schmidt estimator,* which is the MMSE estimator under a chosen norm. It is shown that the error associated with this estimator provides a lower bound on the error associated with any estimator. To establish a notion of the error on $SO(n)$, we have to define a function that computes a distance between any two points on $SO(n)$. We do so by considering the elements of $SO(n)$ as $n \times n$ matrices. For $O \in SO(n)$ and $O = \{O_{ij}: i, j = 1, 2, \ldots, n\}$ define the Hilbert-Schmidt (HS) norm given by $||O|| = \sqrt{(\sum_{i,j=1}^n O_{ij}^2)}$. Due to orthogonality $||O||^2 = n$ and $||O_1 - O_2||^2 = 2(n - trace(O_1 O_2^T))$, where T denotes the matrix transpose. This is also called an extrinsic distance on $SO(n)$ as it is obtained by embedding $SO(n)$ in $\mathbb{R}^{n \times n}$ and inheriting the Euclidean distance there. As an alternative, one can use an intrinsic distance on $SO(n)$ given by $d(O_1, O_2) = \| \log(O_1 O_2^T) \| / \sqrt{2}$, where log is the matrix log. To understand the difference between two distances, consider any two points on a circle. The extrinsic distance corresponds to the chord length between them, whereas the intrinsic distance corresponds to the shortest arc length between them. An analysis resulting from the use of this intrinsic distance is called an intrinsic analysis. In most situations, an intrinsic approach is more natural for statistical inferences. However, in view of the simplicity of analysis and a lower computational cost (see [22]), we describe only the extrinsic analysis in this chapter.

4.1 Minimum Mean Squared Error Estimator

Given an observed image I^D and the target type α, the MMSE estimate of the target orientation, also called the Hilbert-Schmidt estimate, is defined as follows.

Definition 1 *A Hilbert-Schmidt estimate (HSE) is given by* $\hat{s}$: $\mathcal{I}^D \rightarrow SO(n)$, *such that*

$$\hat{O}(I^D) = \underset{O \in SO(n)}{\operatorname{argmin}} \int_{SO(n)} ||O - O'||^2 P(O'|I^D)\gamma(dO'), \tag{3}$$

where γ is the infinitesimal volume element on $SO(n)$.

The HSE can be interpreted as a MMSE estimator when the error is computed using the HS-norm. As the norm squared $|| \cdot ||^2$ is continuous in its entries and $SO(n)$ is compact, the minimizer is attained in $SO(n)$, and hence the estimator is well defined. It should be noted that if the minimum is attained at multiple points, all these points are MMSE estimates (i.e., the estimator is then set-valued). Instead of choosing the MMSE criterion, other error functions (such as absolute difference, step function, etc.) can also be used, resulting in the corresponding estimators.

Due to the geometric properties of rotation matrices, the evaluation of the HSE simplifies to the following form:

$$\hat{O}(I^D) = \underset{O \in SO(n)}{\operatorname{argmax}} \, trace(OA^\dagger) \tag{4}$$

$$= \begin{cases} UV^\dagger, & \text{if } determinant(A) \geq 0 \\ ULV^\dagger, \ L = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & & & \\ 0 & 0 & \dots & -1 \end{bmatrix}, & \text{if } determinant(A) < 0, \end{cases} \tag{5}$$

where

$$A = \int_{SO(n)} OP(O|I^D)\gamma(dO) \tag{6}$$

and where $A = U\Sigma V^\dagger$ is the standard singular value decomposition of A (as described in [8]). The matrices U, Σ, V are arranged such that the singular values occur in decreasing order along the diagonal of Σ. Equation 6 can be interpreted as element by element integration in $\mathbb{R}^{n^2}$ with non-zero contributions only from the rotation matrices. This integral can be computed using one of several numerical integration techniques: a Monte-Carlo sampling technique is presented in [24], and the trapezoidal integration is utilized in [9] to compute $\hat{O}$, the orientation estimate.

4.2 Lower Bound on Expected Error

The next issue is to define a quantity that can be used to assess any given estimator in terms of its expected estimation errors. For example, in the case of Euclidean parameters Cramer-Rao lower bounds are often used to establish the optimum performance and the estimators are judged through these comparisons. In the context of orientation estimation in ATR, we will derive Hilbert-Schmidt bounds (HSBs), which provide a way of comparing different algorithms. The HSB is defined to be the minimum error attainable when the error is specified using the HS-norm.

Definition 2 *Define the HSB as the quantity* $\int_{\mathcal{I}^D} \varrho(I^D)P(I^D)dI^D$, *where dI^D is the base measure on $\mathcal{I}^D$, and*

$$\varrho(I^D) = \int_{SO(n)} ||\hat{s} - s'||^2 P(s'|I^D)\gamma(ds') \, . \tag{7}$$

The importance of the HSB stems from the fact that for any estimator $\tilde{s}$: $\mathcal{I}^D \rightarrow SO(n)$,

$$E||\tilde{O} - O||^2 \geq E||\hat{O} - O||^2 \equiv HSB, \tag{8}$$

where $\hat{O}$ is the HSE as defined earlier. The expectation is over both the randomness in the data and the randomness in the unknown parameters according to

$$E||\tilde{O} - O||^2$$
$$= \int_{\mathcal{I}^D} \left(\int_{SO(n)} ||\tilde{O}(I^D) - O'||^2 P(O'|I^D)\gamma(dO') \right) P(I^D)dI^D.$$

For the proof of this result please refer to [9]. Due to the structure of $SO(n)$, the HSB takes the form:

$$\int_{\mathcal{I}^D} \varrho(I^D)p(I^D)dI^D,$$

where $\rho(I^D) = 2(n - trace(A^\dagger \hat{O}))$ for A as defined in Equation 6. We shall say that the HSE is efficient in the sense that it has HS-efficiency= 1 with the HS-efficiency of an arbitrary estimator $\tilde{s}$: $\mathcal{I}^D \rightarrow SO(n)$ defined as the ratio

$$\text{efficiency}(\hat{O}) = \frac{E||\hat{O} - O||^2}{E||\tilde{O} - O||^2}.$$

Shown in the top panel of Fig. 8 is a plot of the HSB for estimating the truck orientation, in $SO(2)$, as a function of the noise standard deviation, σ. To avoid some symmetry issues (please refer to [21] for a discussion on symmetry issues), this bound is computed by considering only the half circle. The zero expected error implies perfect orientation estimation; the maximal expected error of 1.45 implies completely unreliable estimation of the truck orientation. Superimposed on the error plot are three x's, corresponding to three noise levels. The three truck images in the bottom panels are samples at the noise levels corresponding to the x's. The bottom left corresponds to low-noise resulting in a perfect pose estimation; however, notice the rapid increase in the estimation error as the noise level increases.

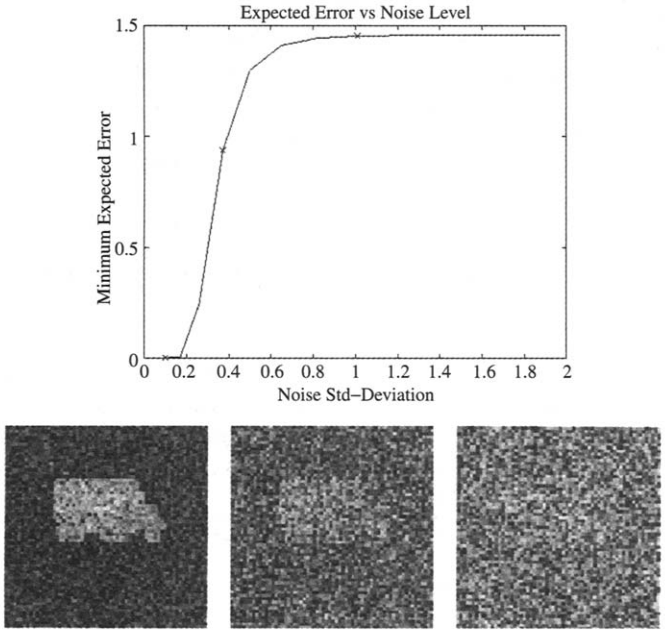

FIGURE 8 Top panel shows the bound for estimating the orientation of a truck using video data. The bottom three panels show sample images of the truck at three noise levels consistent with the x's in top left panel.

To explain the performance curves, at a given noise level, say at noise standard deviation 0.4, the HSB value of the video sensor is 1.0 (i.e., the minimum expected error in estimating truck orientation in this environment is 1.0). Also, for noise level < 0.2, the HSB $\simeq 0$ and for deviation > 0.6, the error is maximal. Errorless estimation is, thus, possible in the case of the video sensor for noise level ≤ 0.2 and reasonable estimates (HSB, 0–1.2) are possible for deviations in the range $[0.2, 0.6]$; beyond that the data are too noisy to provide any information for inference on target orientation. To illustrate the significance of the HSB = 1.0, consider Fig. 9. The bottom-right panel shows a noisy image of the truck at the noise level corresponding to HSB = 1.0. At this particular noise level, the estimation is degraded to such a point that, on average, the estimates span 1.0 HSB units around the mean. Four sample orientations, all within 1.0 HSB units of the orientation shown in the top-left panel, are shown in the other panels. Naturally, the target geometry should determine the bound associated with pose estimation via a given sensor suite, as is depicted in Fig. 10. Shown here are HSB curves for two different targets: tank and truck, when imaged by a video camera. This curve shows that, in low-noise situations the tank orientation estimates are better than the truck estimates via the video sensor while at higher noise levels the performance is similar.

Sensor fusion occurs automatically in this setting. The increased number of data observations $I_1^D, I_2^D, \ldots$ increases the

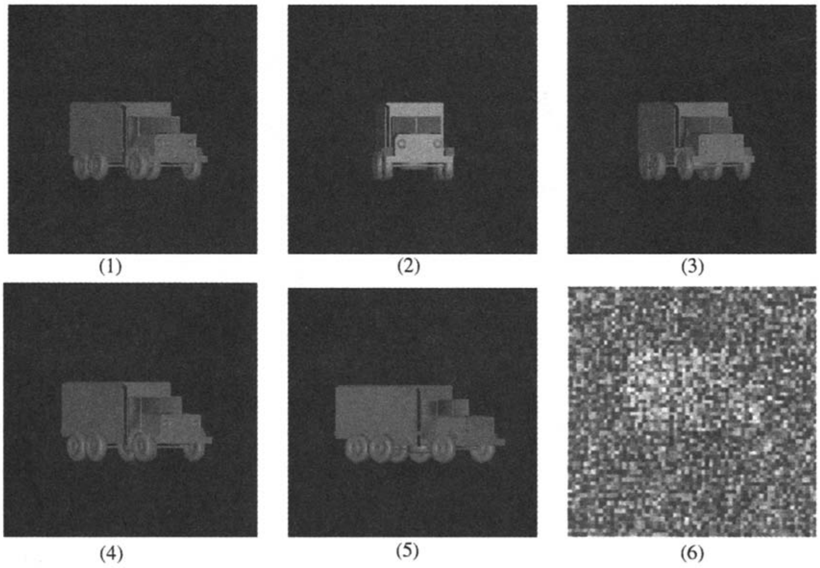

FIGURE 9 Panels 2-4 showing four different trucks orientations within HSB = 1.0 of the orientation in top left. Panel 6 shows the associated imagery with this uncertainty ball.

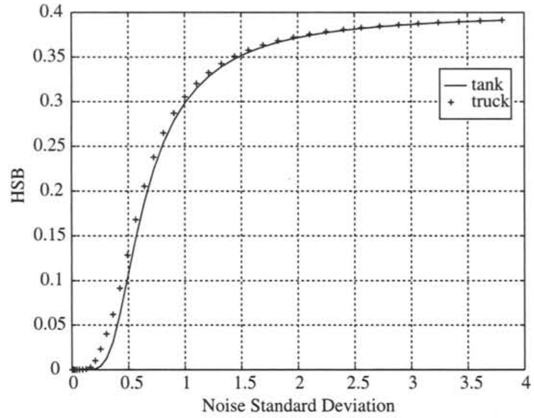

FIGURE 10 Panel shows the variation of the Hilbert-Schmidt bounds with noise for two targets: truck and tank. For low noise levels the bounds are different; at higher noise levels the performance is identical.

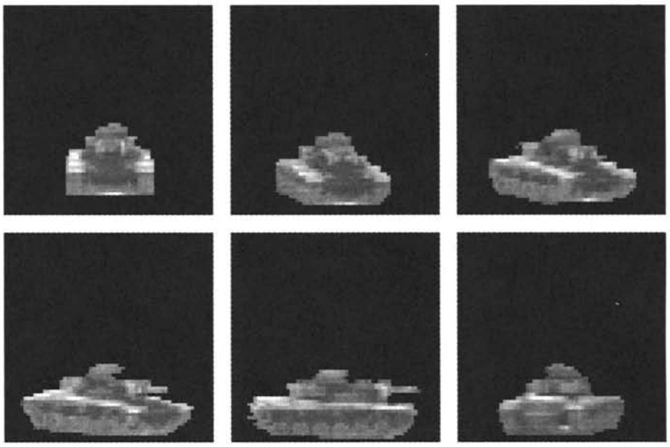

FIGURE 12 Sample images from a dataset of real video images of a tank (data courtesy Dr. Richard Sims of Army Missile Command). The images are down-scaled to 64 × 64 for the results described in this chapter.

accuracy of the estimator. Shown in Fig. 11 are the plots of the HSB on expected error versus noise level in estimating tank orientation for the two sensors: the broken line plots the HSB for HRR, the solid line shows the HSB for video, and the x's display the HSB for the joint case. The right panel shows the HSB curves for the tank pose estimation by three individual sensors: the solid line for FLIR, the broken line for HRR, and the dotted line for video. The HSB for the joint bound, we have utilized a dataset involving real video images of a tank, mounted on a pedestal and imaged at 120 different orientations. This dataset is obtained courtesy of Dr. Richard Sims at Army Missile Command. Shown in Fig. 12 are six sample images from this dataset. Shown in Fig. 13 is the variation of the cumulative position and orientation error [on $SE(2)$] versus the sensor noise. This error bound can be used to analyze multisensor, multitarget situations.

5 Target Recognition and Performance

Having established a framework for target orientation and location estimation, we now focus on the main task: finding the index α that best matches a given image I^D. As described earlier, in a Bayesian framework the estimated target type is given by the index with MAP. It becomes an M-ary hypothesis test. That is

$$\hat{\alpha} = \underset{\alpha \in \mathcal{A}}{\operatorname{argmax}}\, P(\alpha | I^D), \qquad (9)$$

where the posterior is calculated using the Bayes' rule,

$$P(\alpha | I^D) = \frac{P(I^D|\alpha)P(\alpha)}{P(I^D)}.$$

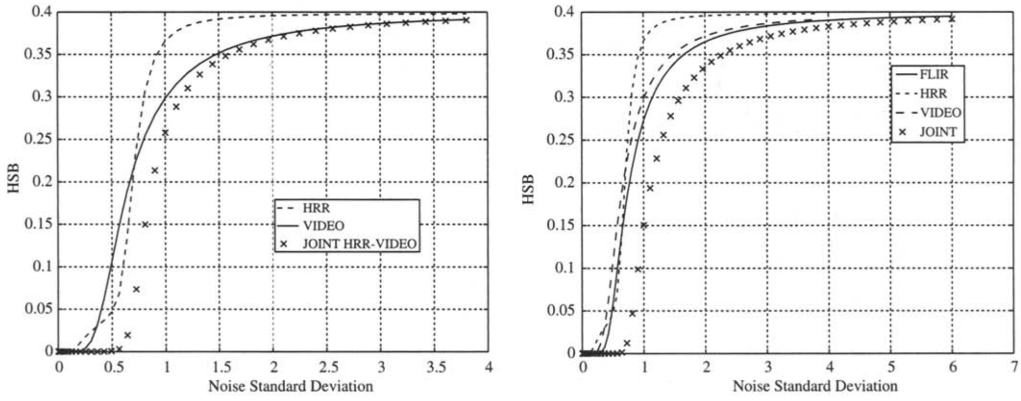

FIGURE 11 Left panel shows the plots of the Hilbert-Schmidt bounds (HSBs) on expected error versus noise level in estimating tank orientation for the two sensors: the broken line plots the HSB for high-range resolution (HRR), the solid line shows the HSB for video, and the x's display the HSB for the joint case. Right panel shows the HSB curves for the tank pose estimation by three individual sensors: the solid line for forward-looking infrared, the broken line for HRR, and the dotted line for video. The HSB for the joint case is shown by the crosses.

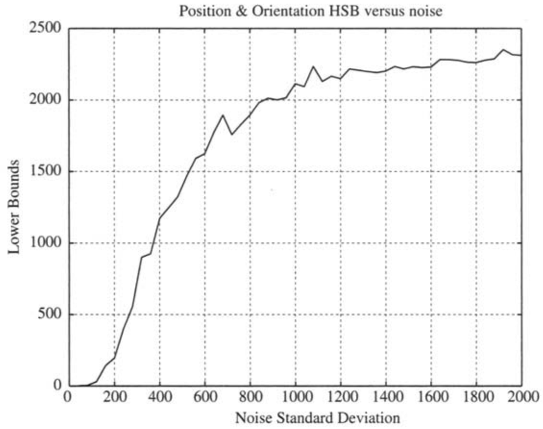

FIGURE 13 Hilbert-Schmidt bounds for joint pose-location estimation on *SE*(2).

The term $P(I^D|\alpha)$ is the likelihood of observing I^D given that the true target is α and can be evaluated as the integration over all transformations

$$P(I^D|\alpha) = \int_S P(I^D|s,\alpha)P(s|\alpha)ds.$$

In the context of selecting α and ATR, s can be considered as a nuisance parameter. This important integral governs the relationship between target recognition (selecting α) and the pose-location estimation (estimating s). It is intuitively clear that recognition and pose estimation are inherently linked; accuracy of target recognition is directly determined by the accuracy of pose estimation.

In most practical situations, the integrand is too complicated to be computed analytically and one of several approximations, numerical and analytical, can be used. To illustrate some of these methods we simplify to **binary** target recognition. That is, given an observed image our task is to select one of the two targets: α_0 or α_1. For binary decision and $S = SO(n)$, the nuisance integral can be evaluated using one of the following three methods:

1. **Quadrature integration:** Since $SO(n)$ is compact, one can compute the integral approximately by evaluating the integrand at some sampled points and using one of the many established formulas (trapezoidal, Simpson's, Gauss-quadrature). As an example, for ground targets ($n = 2$), we have evaluated the integral using the trapezoidal rule and performed hypothesis selection for target recognition. Shown in Fig. 14 are the results from binary recognition for α_0 = truck and α_1 = tank. A video image was simulated for α_0 at some orientation s_0 with respect to the sensor, the integral computed for that image and a decision is made following Bayes'

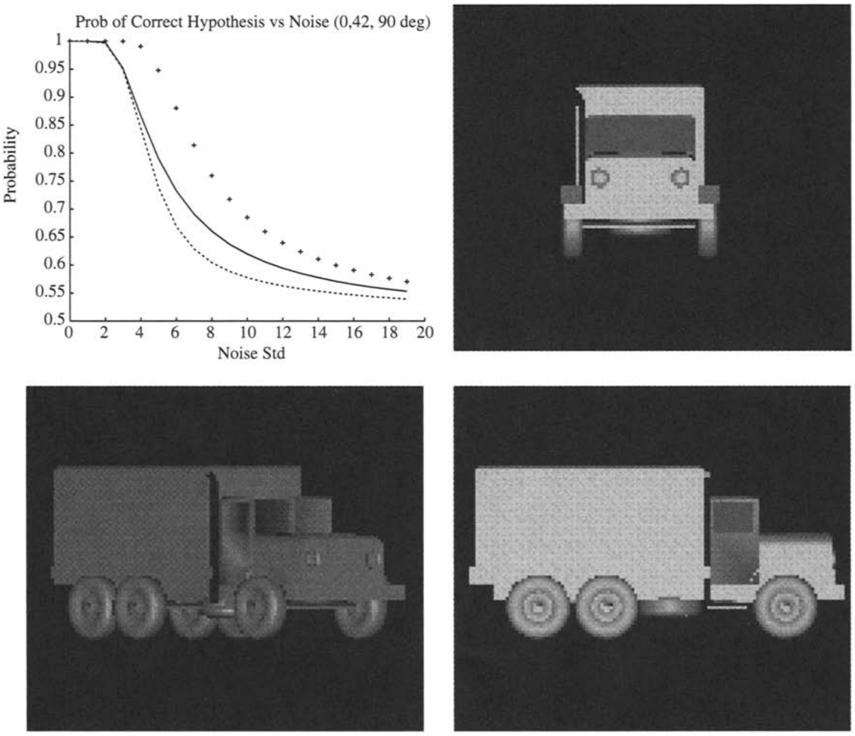

FIGURE 14 The left plot shows the probability of correct hypothesis in binary Bayesian identification plotted against increasing noise for three underlying orientations: the crosses for 90, the solid line for 42, and the broken line for 0 deg. The other three panels show the three underlying truck orientations: **top-right**: (0), **bottom left**: (42); and **bottom right**: (90).

selection. Plotted in the top-left panel are the probabilities of selecting the correct target, α_0, studied against the sensor noise for three different target orientations. Notice that when the target is broadside with most of the pixels in the image, the probability of recognizing it is the highest.

2. **Generalized likelihood ratio:** In this procedure the integral value is approximated by the maximum value of the integrand as a function of the integration variable [26]. The test is given by

$$\frac{\max_{s\in S} P(I^D|s,\alpha_1)}{\max_{s\in S} P(I^D|s,\alpha_0)} \underset{\alpha_0}{\overset{\alpha_1}{\gtrless}} \frac{P(\alpha_0)}{P(\alpha_1)}.$$

   In other words, the maximum likelihood estimation of s is calculated for both the hypotheses, and the ratio of maximum likelihoods compared to the ratio of prior probabilities decides the hypothesis selection.

3. **Asymptotics:** To obtain analytic expressions, which are often more useful than the numeric approximations, asymptotic approximations using Laplace's method [19] can be derived. The basic approach is to assume a very large signal-to-noise ratio, either through large sample size or small sensor noise, and approximate the integrand using normal approximation of the integrand [11]. This result is then used in computing the likelihood ratio and, furthermore, the probability of error in the hypothesis selection. The error probability decreases exponentially with the decrease in the sensor noise, with the rate depending on the accuracy in pose estimation. This highlights the relevance of transformation estimation accuracy in hypothesis testing. A more accurate pose estimator can lead to a better recognition system.

6 Discussion

In this chapter, we have described a model-based, Bayesian approach to automated target recognition. Models for targets are developed using a deformable template approach in which each target occurrence in a given scene is modeled using a template and a transformation. The transformations associated with ATR are Lie groups and have curved geometry. Using the Hilbert-Schmidt norm we have defined a MMSE estimator for pose and pose-location estimates, and also lower bounded the expected squared error for any estimator. Pose/location estimates are incorporated in target recognition which is performed using Bayesian hypothesis selection. The posterior calculation includes an integration over the nuisance parameters and several methods are presented to perform this numerically. The asymptotic technique leads to an analytic expression for the performance analysis by providing the probability of errors in recognition.

7 Acknowledgment

This research was supported in part by the grants ARO DAAD19-99-1-0267, NSF (FRG) DMS-0101429, NMA 201-01-2010, ARO W911NF-04-01-0268, and NSF (ACT) DMS-0345242.

References

[1] J. K. Agarwal and S. Shah, "Object recognition and performance bounds," *Lecture Notes in Computer Science: Image Analysis and Processing* 343–360 (1997).

[2] O. Barndorff-Nielsen, J. Kent, and M. Sorensen, "Normal variance-mean mixtures and z distributions," *Int. Stat. Rev.* 50, 145–159 (1982).

[3] W. M. Boothby, *An Introduction to Differential Manifolds and Riemannian Geometry.* (Academic Press, New York, 1986).

[4] M. Cooper and M. Miller, "Information measures for object recognition accommodating signature variability," *IEEE Trans. Inf. Theory*, 46, 1896–1907 (2000).

[5] D. E. Dudgeon, "ATR performance modeling and estimation," in *MIT Lincoln Labs Technical Report 1051* (Lexington, MA, December 1998).

[6] D. E. Dudgeon and R. T. Lacoss, "An overview of automatic target recognition," *MIT Lincoln Lab. J.* 6, 3–10 (1993).

[7] F. Garber and E. Zelnio, "On some simple estimates of atr performance, and initial comparisons for a small data set," in *Proc. SPIE*, 3070, 150–161 (1997).

[8] G. H. Golub and C. F. Van Loan, *Matrix Computations*, (Johns Hopkins University Press, Baltimore, 1989).

[9] U. Grenander, M. I. Miller, and A. Srivastava, "Hilbert-Schmidt lower bounds for estimators on matrix Lie groups for ATR," *IEEE Trans. PAMI*, 20, 790–802 (1998).

[10] U. Grenander and A. Srivastava, "Probability models for clutter in natural images," *IEEE Trans. Patt. Anal. Mach. Intell.* 23, 424–429 (2001).

[11] U. Grenander, A. Srivastava, and M. I. Miller, "Asymptotic performance analysis of bayesian object recognition," *IEEE Trans. Inf. Theory*, 46, 1658–1666 (2000).

[12] S. P. Jacobs and J. A. O'Sullivan, "Automated target recognition using sequences of high resolution radar range profiles," *IEEE Trans. Aerospace Electr. Syst.* 36, 364–382 (2000).

[13] B. Jahne, H. Haubecker, and P. Geibler, *Handbook of Computer Vision and Applications, Volume 2.* (Academic Press, New York, 1999).

[14] A. D. Lanterman, M. I. Miller, and D. L. Snyder, "Implementation of jump-diffusion processes for understanding FLIR scenes," in F. A. Sadjadi, ed., *Automatic Object Recognition V*, 2485, 309–320, Orlando, FL, April 1995. SPIE.

[15] M. Lindenbaum, "Bounds on shape recognition performance," *IEEE Trans. Patt. Anal. Mach. Intell.* 17, 666–680 (1995).

[16] M. Loizeaux, A. Srivastava, and M. I. Miller, "Pose/location estimation of ground targets," *Signal Processing, Sensor Fusion, and Target Recognition, Proceedings of SPIE,* 3720, 140–151 (1999).

[17] D. Mumford, "Empirical investigations into the statistics of clutter and the mathematical models it leads to," *A lecture for the review of ARO Metric Pattern Theory Collaborative,* 2000.

[18] L. M. Novak, G. R. Benitz, G. J. Owirka, and L. A. Bessette, "ATR performance using enhanced resolution sar," in *Proc. SPIE*, 2757, 332–337 (1996).

[19] G. Polya and G. Szego, *Problems and Theorems in Analysis: Translated by D. Aeppli.* (Springer-Verlag, 1976).

[20] D. L. Snyder, A. M. Hammoud, and R. L. White, "Image recovery from data acquired with a charge-coupled-device camera," *J. Opt. Soc. Am. A*, 10, 1014–1023 (1993).

[21] A. Srivastava and U. Grenander, "Metrics for target recognition," in *Proc. SPIE, Applications of Artificial Neural Networks in Image Processing III*, 3307, 29–37 (1998).

[22] A. Srivastava and E. Klassen, "Monte Carlo extrinsic estimators for manifold-valued parameters," *IEEE Trans. Signal Process.* 50, 299–308 (2001).

[23] A. Srivastava, X. Liu, and U. Grenander, "Universal analytical forms for modeling image probability," *IEEE Trans. Patt. Anal. Mach. Intell.* 28, 1200–1214 (2002).

[24] A. Srivastava, M. I. Miller, and U. Grenander, *Ergodic Algorithms on Special Euclidean Groups for ATR. Systems and Control in the Twenty-First Century: Progress in Systems and Control,* Vol. 22. (Birkhauser, 1997).

[25] M. Steinberger, *Algebra.* (PWS Publishing Company, Boston, 1994).

[26] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Volume I.* (John Wiley, New York, 1971).

[27] A. C. Williams and B. Clark, "Evaluation of SAR ATR," in *Proc. SPIE*, 2755, 36–45 (1996).