PERGAMON

Pattern Recognition 36 (2003) 1213–1231

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Comparative analysis of different approaches to target differentiation and localization with sonar

Billur Barshan*, Birsel Ayrulu

Department of Electrical Engineering, Bilkent University, Bilkent, TR-06533 Ankara, Turkey

Received 29 November 2001; received in revised form 5 July 2002; accepted 5 July 2002

Abstract

This study compares the performances of different methods for the differentiation and localization of commonly encountered features in indoor environments. Differentiation of such features is of interest for intelligent systems in a variety of applications such as system control based on acoustic signal detection and identification, map building, navigation, obstacle avoidance, and target tracking. Different representations of amplitude and time-of-flight measurement patterns experimentally acquired from a real sonar system are processed. The approaches compared in this study include the target differentiation algorithm, Dempster–Shafer evidential reasoning, different kinds of voting schemes, statistical pattern recognition techniques (k-nearest neighbor classifier, kernel estimator, parameterized density estimator, linear discriminant analysis, and fuzzy c-means clustering algorithm), and artificial neural networks. The neural networks are trained with different input signal representations obtained using pre-processing techniques such as discrete ordinary and fractional Fourier, Hartley and wavelet transforms, and Kohonen's self-organizing feature map. The use of neural networks trained with the back-propagation algorithm, usually with fractional Fourier transform or wavelet pre-processing results in near perfect differentiation, around 85% correct range estimation and around 95% correct azimuth estimation, which would be satisfactory in a wide range of applications.

Keywords: Target classification; Target differentiation; Target localization; Dempster-Shafer evidential reasoning; Majority voting; Kernel estimator; Nearest-neighbor classifier; Parameterized density estimation; Linear discriminant analysis; Fuzzy c-means clustering; Artificial neural networks; Sonar sensing

1. Introduction

Intelligent systems, especially those which interact with, or act upon their surroundings need the model of the environment in which they operate. They can obtain this model partly or entirely using one or more sensors and/or viewpoints. An important example of such systems is fully or partly autonomous mobile robots. For instance, considering typical indoor environments, a mobile robot must be able to differentiate planar walls, corners, edges, and cylinders for map-building, navigation, obstacle avoidance, and target-tracking applications.

Reliable differentiation is crucial for robust operation and is highly dependent on the mode(s) of sensing employed. Sonar sensing is one of the most useful and cost-effective modes of sensing. The fact that sonar sensors are light, robust, and inexpensive devices has led to their widespread use in applications such as navigation of autonomous vehicles through unstructured environments [1–3], map building [4–6], target tracking [7], and obstacle avoidance [8]. Although there are difficulties in the interpretation of sonar data due to poor angular resolution of sonar, multiple and higher-order reflections, and establishing correspondence between multiple echoes on different receivers [9,10], these difficulties can be overcome by employing accurate physical models for the reflection of sonar. Sonar ranging systems commonly employ only the *time-of-flight* (TOF) information, recording the time elapsed between

* Corresponding author. Tel.: +90-312-290-2161; fax: +90-312-266-4192.
E-mail address: billur@ee.bilkent.edu.tr (B. Barshan).

0031-3203/03/$30.00
PII: S0031-3203(02)00167-X

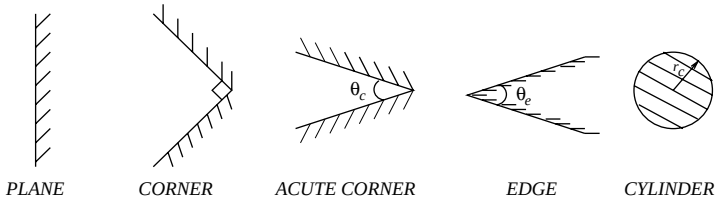

Fig. 1. Horizontal cross sections of the target primitives/features differentiated in this study.

the transmission and reception of a pulse [11]. A review of work using this approach can be found in Refs. [12,13].

The purpose of this paper is to present a comprehensive comparison of a diverse array of methods for differentiating and localizing targets based on returns from inexpensive sonar sensors. We consider several variations of each method and determine their optimal operating configurations and parameters. The methods considered are the target differentiation algorithm (TDA), Dempster-Shafer (D-S) evidential reasoning, simple majority voting (SMV) and various other voting schemes with preference ordering and reliability measures, statistical pattern recognition techniques (ordinary and generalized k-nearest neighbor (k-NN) classifiers, kernel estimator (KE), parameterized density estimator (PDE), linear discriminant analysis (LDA), and fuzzy c-means clustering (FCC) algorithm), and artificial neural networks (ANNs).

In this paper, we consolidate the results of our studies of the above methods, spanning a period of 5 years. Results associated with some of the methods listed above were published in Refs. [12–15], sometimes in the context of a specific application. This paper presents these results uniformly together with previously unpublished methods and results. To the best of our knowledge, there is no previously published work undertaking such a uniform comparison of any substantial set of the methods considered here with any comparable degree of generality. Given the attractive performance for cost of sonar-based systems, we believe that the results of this study will be of great usefulness for those designing and implementing sonar systems as well as researchers in this area.

The paper is organized as follows: Section 2 describes the sensing configuration used in this study and introduces the target primitives. In Section 3, the TDA used in earlier work [14] is reviewed. The use of two non-parametric classification methods, D-S evidential reasoning and majority voting, is described in Sections 4 and 5, respectively. In Section 6, statistical pattern recognition techniques are considered. In Section 7, we focus on ANNs. In Section 8, the performances of all these classification schemes in target classification and localization are compared based on experimental data. In the last section, concluding remarks are made.

2. Sonar sensing

The basic target types or features differentiated in this study are *plane*, *corner*, *acute corner*, *edge* and *cylinder* (Fig. 1). In particular, we have employed a planar target, a corner of $\theta_c = 90°$, an acute corner of $\theta_c = 60°$, an edge of $\theta_e = 90°$, and cylinders with radii $r_c = 2.5, 5.0$ and 7.5 cm, all made of wood. Detailed reflection models of these are provided in Ref. [14].

The most common sonar ranging system is based on TOF which is the time elapsed between the transmission and the reception of a pulse. In commonly used TOF systems, an echo is produced when the transmitted pulse encounters an object and a range measurement $r = ct_0/2$ is obtained (Fig. 2) by *simple thresholding* [16]. Here, t_0 is the TOF and c is the speed of sound in air (at room temperature, $c = 343.3$ m/s).

The major limitation of sonar sensors comes from their large beamwidth. Although these devices return accurate range data, they cannot provide direct information on the angular position of the object from which the reflection was obtained. Sensory information from a single sonar sensor has poor angular resolution and is usually not sufficient to differentiate more than a small number of target primitives [17]. Improved target classification can be achieved by using multiple sensors and by employing both amplitude and TOF information. However, a major problem with using the amplitude information of sonar signals is that the amplitude is very sensitive to environmental conditions. For this reason, and also because the standard electronics used in practical work typically provide only TOF data, amplitude information is rarely used. Barshan and Kuc's early work on the use of amplitude information [17] has been extended to a variety of target types in Ref. [14] using both amplitude and TOF information. In the present paper, amplitude and TOF information from a pair of identical ultrasonic transducers a and b with center-to-center separation $d = 25$ cm is employed to improve the angular resolution [15].

Panasonic transducers [18] with aperture radius $a = 0.65$ cm, resonance frequency $f_0 = 40$ kHz, and beamwidth 108° are used in our experiments. The entire sensing unit is mounted on a small 6 V computer-controlled stepper motor with step size 1.8°. Data acquisition from the sonars is through a 12-bit 1 MHz PC A/D card. Starting at the transmit time, 10,000 samples of each echo signal are collected to record the peak amplitude and the TOF.

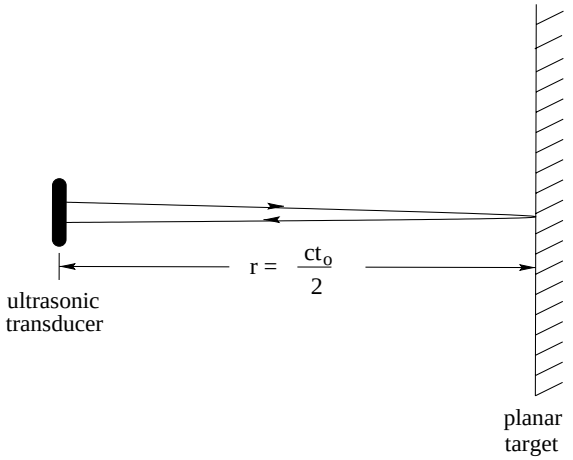

Fig. 2. Reflection of ultrasonic echoes from a planar target.

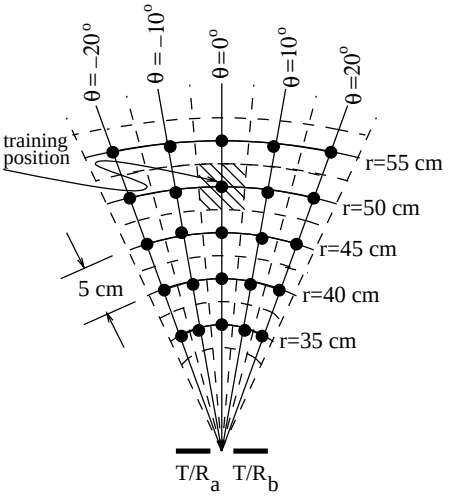

Fig. 3. Discrete training locations. T/R_a and T/R_b denote the two transmitting/receiving transducers.

Amplitude and TOF patterns of the targets are collected in this manner at 25 different locations (r, θ) for each target, from $\theta = -20^\circ$ to 20° in 10° increments, and from $r = 35$ to 55 cm in 5 cm increments (Fig. 3). The target located at range r and azimuth θ is scanned by the rotating sensing unit for scan angles $-52^\circ \leqslant \alpha \leqslant 52^\circ$ with 1.8° increments (determined by the step size of the motor). The angle α is always measured with respect to $\theta = 0^\circ$ as shown in Fig. 4.

At each step of the scan (for each value of α), four sonar echo signals are acquired. The echo signals are in the form of slightly skewed wave packets [13]. A_{aa}, A_{bb}, A_{ab}, and A_{ba} denote the peak values of the echo signals, and t_{aa}, t_{bb}, t_{ab}, and t_{ba} denote their TOF delays (extracted by simple thresholding). The first subscript indicates the transmitting transducer, the second denotes the receiver. At each

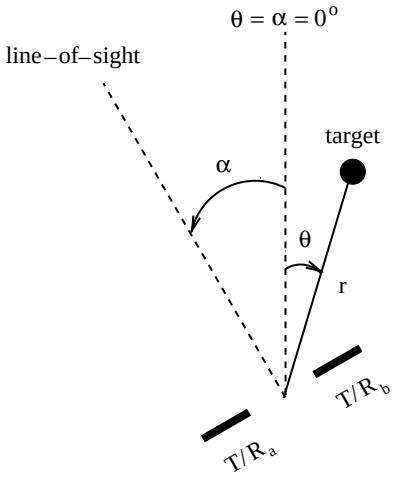

Fig. 4. The scan angle α and the target azimuth θ.

step of the scan, only these eight amplitude and TOF values extracted from the four echo signals are recorded. For the given scan range and motor step size, $58(=(2 \times 52^\circ)/1.8^\circ)$ angular samples of each of the amplitude and TOF patterns $A_{aa}(\alpha)$, $A_{bb}(\alpha)$, $A_{ab}(\alpha)$, $A_{ba}(\alpha)$, $t_{aa}(\alpha)$, $t_{bb}(\alpha)$, $t_{ab}(\alpha)$, and $t_{ba}(\alpha)$ are acquired at each target location.

Since the cross terms $A_{ab}(\alpha)$ and $A_{ba}(\alpha)$ (or $t_{ab}(\alpha)$ and $t_{ba}(\alpha)$) should ideally be equal due to reciprocity, it is more representative to employ their average. Thus, 58 samples each of the following six functions are taken collectively as acoustic signatures embodying shape and position information of a given target:

$$A_{aa}(\alpha),\ A_{bb}(\alpha),\ \frac{A_{ab}(\alpha)+A_{ba}(\alpha)}{2},\ t_{aa}(\alpha),\ t_{bb}(\alpha),$$

$$\text{and}\quad \frac{t_{ab}(\alpha)+t_{ba}(\alpha)}{2}. \tag{1}$$

Scans are collected with four-fold redundancy for each target primitive at each location, resulting in 700 (=four-fold redundancy × 25 locations × seven target types) sets of scans to be used for training. This set of 700 data is referred as the *training set* throughout this paper. This training set is used to design decision rules in statistical pattern recognition techniques and to train the ANNs.

In this study, three different *test sets* are acquired to evaluate and compare the different methods. For *test set* I, each target is placed in turn in each of the 25 training positions in Fig. 3. Again, scans are collected with four-fold redundancy for each combination of target type and location, resulting in 700 sets of experimentally acquired scans. While collecting *test set* II, the targets are situated arbitrarily in the continuous estimation space and not necessarily confined to one of the 25 training positions. The values of r, θ corresponding to these locations are randomly and uniformly generated in the range $r \in [32.5\text{ cm}, 57.5\text{ cm}]$ and $\theta \in [-25^\circ, 25^\circ]$. In collecting *test set* III, we employ targets not scanned

during training which are slightly different in size, shape, or roughness than the targets used for training. These are two smooth cylinders of radii 4 and 10 cm, a cylinder of radius 7.5 cm and a plane both covered with blister packaging material, and a 60° smooth edge. The blister packaging material has a honeycomb pattern of uniformly distributed circular bubbles of diameter 1.0 cm and height 0.3 cm, with a center-to-center separation of 1.2 cm.

3. Target differentiation algorithm (TDA)

The TDA has its origins in the plane/corner differentiation algorithm developed in Ref. [17], which is based on the idea of exploiting amplitude differentials in resolving target type. In Ref. [14], the algorithm is extended to include other target primitives using both amplitude and TOF differentials, based on their characteristics presented earlier in Ref. [12]. The extended algorithm may be summarized in the form of rules:

if $[t_{aa}(\alpha) - t_{ab}(\alpha)] > k_t\sigma_t$ and $[t_{bb}(\alpha) - t_{ba}(\alpha)] > k_t\sigma_t$ then *acute corner*,
else if $[A_{aa}(\alpha) - A_{ab}(\alpha)] > k_A\sigma_A$ and $[A_{bb}(\alpha) - A_{ba}(\alpha)] > k_A\sigma_A$ then *plane*,
else if $[\max_\alpha\{A_{aa}(\alpha)\} - \max_\alpha\{A_{bb}(\alpha)\}] < k_A\sigma_A$ and $[\max_\alpha\{A_{aa}(\alpha)\} - \max_\alpha\{A_{ab}(\alpha)\}] < k_A\sigma_A$ then *corner*, else *edge*, *cylinder* or *unknown*.

In the above algorithm, k_A (or k_t) are the number of amplitude (or TOF) noise standard deviations σ_A (or σ_t), respectively, and are employed as safety margins to achieve robustness in the differentiation process. Differentiation is achievable only in those cases where the difference in amplitudes (and TOFs) exceeds $k_A\sigma_A$ (or $k_t\sigma_t$). If this is not the case, a decision cannot be made and the target type remains unknown.

The above algorithm cannot distinguish between edges and cylinders. Edges and cylinders can be differentiated with a similar configuration of transducers using a method based on radius of curvature estimation [19]. For the cylinder, the radius of curvature has two limits of interest. As $r_c \to 0$ the characteristics of the cylinder approach those of an edge.

$$m(c) = \begin{cases} (1-K_4)\dfrac{K_2[A_{ab}(\alpha) - A_{aa}(\alpha)] + K_3[A_{ba}(\alpha) - A_{bb}(\alpha)]}{K_2\max_\alpha[A_{ab}(\alpha) - A_{aa}(\alpha)] + K_3\max_\alpha[A_{ba}(\alpha) - A_{bb}(\alpha)]} & \text{if } K_2 \neq 0 \text{ or } K_3 \neq 0, \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

On the other hand, as $r_c \to \infty$, the characteristics approach those of a plane. By assuming the target is a cylinder first and estimating its radius of curvature [19], it is possible to distinguish edges and cylinders.

After determining the target type using the TDA summarized above, the target range and azimuth can be estimated from the geometry [14]. In addition to the radius r_c of cylinders, the wedge angle θ_c of acute corners can also be estimated [14].

4. Dempster-Shafer (D-S) evidential reasoning

In D-S evidential reasoning, different opinions are represented by belief functions [20]. These are set functions which assign numerical degrees of support on the basis of evidence, but also allow for the expression of ignorance: belief can be committed to a set or proposition without commitment to its complement. In the D-S method, a priori information is not required and the belief assignment is made only when sensor readings provide supportive evidence. Therefore, ignorance can be represented explicitly. Conflict between views is represented by a conflict measure which is used to normalize the sensor belief assignments. Letting Ω represent a finite set of elementary propositions, a *basic probability mass assignment* $m(.)$ maps each subset A of Ω to a number between 0 and 1 such that $m(\emptyset) = 0$ and $\sum_{A\subseteq\Omega} m(A) = 1$. The *belief* (or total support) that is assigned to a subset A of Ω is obtained by summing the basic probability assignments over all subsets of A as $Bel(A) = \sum_{B\subseteq A} m(B)$.

A brief survey of D-S evidential reasoning in classification problems is provided in Ref. [13]. In this study, the different viewpoints of the sensing unit are assigned beliefs using D-S evidential reasoning and they are combined through Dempster's fusion rule. The uncertainty in the measurements is represented by a belief function $BF = \{\text{target}_i; m(\text{target}_i)\}_{i=1}^{c}$, which is a set consisting of targets and their corresponding basic probability mass assignments $m(.)$, and c is the number of different target types. The basic probability mass assignment is the underlying function for decision making using the D-S method. Here, these are defined based on the TDA outlined in Section 3 and are thus dependent on amplitude and TOF differentials such that the larger the differential, the larger the degree of belief [see Eqs. (2)–(4)]. The basic probability mass assignment values are scaled to fall in the interval $[0, 1]$ as given below:

$$m(p) = (1-K_4)K_1 \frac{[A_{aa}(\alpha) - A_{ab}(\alpha)] + [A_{bb}(\alpha) - A_{ba}(\alpha)]}{\max_\alpha[A_{aa}(\alpha) - A_{ab}(\alpha)] + \max_\alpha[A_{bb}(\alpha) - A_{ba}(\alpha)]}, \tag{2}$$

$$m(ac) = K_4\frac{[t_{aa}(\alpha) - t_{ab}(\alpha)] + [t_{bb}(\alpha) - t_{ba}(\alpha)]}{\max_\alpha[t_{aa}(\alpha) - t_{ab}(\alpha)] + \max_\alpha[t_{bb}(\alpha) - t_{ba}(\alpha)]}, \tag{4}$$

where $m(p), m(c)$, and $m(ac)$ correspond to plane, corner, and acute corner assignments, respectively, and K_1, K_2, K_3, and K_4 are the indicator functions of the conditions

given below:

$$K_1 = \begin{cases} 1 & \text{if } [A_{aa}(\alpha) - A_{ab}(\alpha)] > k_A\sigma_A \quad \text{and} \\ & [A_{bb}(\alpha) - A_{ba}(\alpha)] > k_A\sigma_A, \\ 0 & \text{otherwise,} \end{cases}$$

$$K_2 = \begin{cases} 1 & \text{if } [A_{ab}(\alpha) - A_{aa}(\alpha)] > k_A\sigma_A, \\ 0 & \text{otherwise,} \end{cases}$$

$$K_3 = \begin{cases} 1 & \text{if } [A_{ba}(\alpha) - A_{bb}(\alpha)] > k_A\sigma_A, \\ 0 & \text{otherwise,} \end{cases}$$

$$K_4 = \begin{cases} 1 & \text{if } [t_{aa}(\alpha) - t_{ab}(\alpha)] > k_t\sigma_t \quad \text{and} \\ & [t_{bb}(\alpha) - t_{ba}(\alpha)] > k_t\sigma_t, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

The remaining belief represents ignorance, or undistributed probability mass and is given by $m(u) = 1 - [m(p) + m(c) + m(ac)]$. This uncommitted belief is the result of lack of evidence supporting any one target type more than another.

Given two independent sources of information regarding the target type with the following belief functions:

$$\begin{aligned} BF_1 &= \{\text{target}_i; m_1(\text{target}_i)\}_{i=1}^4 \\ &= \{p, c, ac, u; m_1(p), m_1(c), m_1(ac), m_1(u)\}, \\ BF_2 &= \{\text{target}_j; m_2(\text{target}_j)\}_{j=1}^4 \\ &= \{p, c, ac, u; m_2(p), m_2(c), m_2(ac), m_2(u)\}, \end{aligned} \tag{6}$$

the information from the two independent sources are fused (combined) as follows:

$$\begin{aligned} \text{BF}_f &= \text{BF}_1 \oplus \text{BF}_2 = \{\text{target}_k; m_f(\text{target}_k)\}_{k=1}^4 \\ &= \{p, c, ac, u; m_f(p), m_f(c), m_f(ac), m_f(u)\} \end{aligned} \tag{7}$$

where

$$\begin{aligned} m_f(p) &= \frac{m_1(p)m_2(p) + m_1(p)m_2(u) + m_1(u)m_2(p)}{1 - \text{conflict}}, \\ m_f(c) &= \frac{m_1(c)m_2(c) + m_1(c)m_2(u) + m_1(u)m_2(c)}{1 - \text{conflict}}, \\ m_f(ac) &= \frac{m_1(ac)m_2(ac) + m_1(ac)m_2(u) + m_1(u)m_2(ac)}{1 - \text{conflict}}, \\ m_f(u) &= \frac{m_1(u)m_2(u)}{1 - \text{conflict}}. \end{aligned} \tag{8}$$

In these equations, disagreement between the two sources of information is represented by the "conflict" term that represents the degree of mismatch. The conflict measure is expressed as

$$\begin{aligned} \text{conflict} &= m_1(p)m_2(c) + m_1(c)m_2(p) + m_1(p)m_2(ac) \\ &\quad + m_1(ac)m_2(p) + m_1(c)m_2(ac) + m_1(ac)m_2(c). \end{aligned} \tag{9}$$

The denominators of Eq. (8) normalize the beliefs after discounting this conflict. The target type with the maximum belief value in these equations is chosen to be the group decision. Eq. (8) is a special case of the powerful evidence combination rule called Dempster's rule of combination or fusion [20]:

$$\begin{aligned} &m_f(\text{target}_k) \\ &\triangleq \frac{\sum\sum_{\text{target}_k = \text{target}_i \cap \text{target}_j} m_1(\text{target}_i)m_2(\text{target}_j)}{1 - \sum\sum_{\text{target}_i \cap \text{target}_j = \emptyset} m_1(\text{target}_i)m_2(\text{target}_j)}, \end{aligned} \tag{10}$$

where $\sum\sum_{\text{target}_i \cap \text{target}_j = \emptyset} m_1(\text{target}_i)m_2(\text{target}_j)$ is the measure of conflict.

The fusion process can be easily extended from two to n sources of information as $BF_f = (((BF_1 \oplus BF_2) \oplus BF_3) \oplus \cdots \oplus BF_n)$ which is both associative and commutative.

In implementing this method, first, the target type is found by employing the TDA at each angular step of the scan, and its range and azimuth are estimated. Then, the target type decisions made at each of the 58 angular steps are fused using D-S evidential reasoning to reach a single final target type decision for a particular scan. Weighted averages of the 58 r and θ estimates are calculated to find the fused range and the azimuth estimates of the target. The weights used are the ratio of the belief value assigned to the r (or θ) estimate at each angular step (described later in Section 5, see Eq. (11)) to the sum of the belief values assigned to the r (or θ) estimates over all 58 angular steps.

5. Conflict resolution through voting

Voting can take place among multiple sensors and/or complementary views of a single sensor which can give rise to conflicts that must be resolved to reach consensus. Voting, in its simplest form, has the advantages of being computationally inexpensive and, to a certain degree, fault tolerant. The major drawback of voting is the consistency problem of Arrow which states that there is no voting scheme for selecting among more than two alternatives that is locally consistent under all possible conditions [21]. In the present paper, voting takes place among the opinions produced by the sensing unit at different scan angles.

In *simple* majority voting (SMV), the votes are given equal weight and the consensus is taken as the outcome with the largest number of votes. Although SMV provides fast and robust fusion in some problems, there exist some drawbacks that limit its usage. For example, in cases when all outcomes receive equal votes, a consensus cannot be reached. Furthermore, SMV does not take into account the distribution of votes among dissenters (those voting for the losing alternatives) [13].

To overcome these drawbacks and to increase the reliability and consistency of the decision, more sophisticated voting schemes can be employed. For this purpose, the sensing unit assigns preference orders to the possible target types

at each scan angle and a reliability measure is introduced. The preference order at each scan angle is determined according to the belief assignments given in Eqs. (2)–(4) with the largest belief corresponding to the first preference order (most preferred), and so on with decreasing beliefs.

The reliability measure represents how much we can trust a particular piece of information. Our reliability measures will be defined in terms of basic probability mass assignments. The closer the target is to the sensing unit, the more accurate is the range reading, and the closer the target is to the line-of-sight of the sensing unit, the more accurate is the azimuth estimate [22]. This is because signal amplitude decreases with r and $|\theta|$ so that at large ranges or angular deviations (from the line-of-sight), signal-to-noise ratio is smaller. For each scan angle α, a range r_α and azimuth θ_α estimate is obtained from the TOF measurements using the geometry. Then, the basic probability mass assignments are made as

$$m(r_\alpha) = \frac{r_{max} - r_\alpha}{r_{max} - r_{min}}, \quad r_{min} \leqslant r_\alpha \leqslant r_{max},$$

$$m(\theta_\alpha) = \frac{\theta_0 - |\theta_\alpha|}{\theta_0}, \quad 0 \leqslant |\theta_\alpha| \leqslant \theta_0, \tag{11}$$

where r_{min} and r_{max} define the operating range of the sensing unit and θ_0 is the half-beamwidth angle. Note that, $m(r_\alpha)$ takes its maximum value of one when $r_\alpha = r_{min}$ and its minimum value of zero when $r_\alpha = r_{max}$. Similarly, $m(\theta_\alpha)$ is one when $\theta_\alpha = 0^\circ$ and zero when $\theta_\alpha = \pm\theta_0$.

We have considered several different reliability measures:

$$\mathrm{rel}_\alpha^1 = m(r_\alpha)\, m(\theta_\alpha), \quad \mathrm{rel}_\alpha^2 = \min\{m(r_\alpha), m(\theta_\alpha)\},$$

$$\mathrm{rel}_\alpha^3 = \frac{m(r_\alpha) + m(\theta_\alpha)}{2}, \quad \mathrm{rel}_\alpha^4 = \max\{m(r_\alpha), m(\theta_\alpha)\}, \tag{12}$$

which satisfy $\mathrm{rel}_\alpha^1 \leqslant \mathrm{rel}_\alpha^2 \leqslant \mathrm{rel}_\alpha^3 \leqslant \mathrm{rel}_\alpha^4$. All of these measures take values in the interval $[0, 1]$, with 0 being unreliable and 1 being most reliable.

A fifth alternative is to set the reliability measure proportional to the difference between belief values assigned to the first two preferences, as an indicator of how large a margin the first choice is ahead of the second choice. This way, the distribution of the belief assignments to different target types is partially taken into account. This can be expressed as $\mathrm{rel}_\alpha^5 = m(\text{1st choice}_\alpha) - m(\text{2nd choice}_\alpha)$ where the functions $m(.)$ are now those defined in Eqs. (2)–(4).

The final preference ordering for the targets is obtained from the orders at each scan angle by taking the weighted average of the preference orders assigned at each scan angle, with the reliability measure serving as the weighting function [13]. For comparison, we have also considered the use of weighting the preference orders with unit reliability measures.

In SMV, range and azimuth estimates are averaged over the complete scan to obtain the final r and θ estimates of the target. The same is done for voting with preference ordering if the reliability measures are taken as unity. When the reliability measures are taken according to one of the five alternatives above instead, the ratio of the reliability assigned to a particular angular step to the sum of the reliabilities assigned to all 58 angular steps are used as weights.

6. Statistical pattern recognition techniques

We begin by constructing three alternative feature vector representations from the scans of Eq. (1):

$$\mathbf{x}_A : \left[\mathbf{A}_{aa}, \mathbf{A}_{bb}, \frac{\mathbf{A}_{ab} + \mathbf{A}_{ba}}{2}, \mathbf{t}_{aa}, \mathbf{t}_{bb}, \frac{\mathbf{t}_{ab} + \mathbf{t}_{ba}}{2}\right]^{\mathrm{T}},$$

$$\mathbf{x}_B : [\mathbf{A}_{aa} - \mathbf{A}_{ab}, \mathbf{A}_{bb} - \mathbf{A}_{ba}, \mathbf{t}_{aa} - \mathbf{t}_{ab}, \mathbf{t}_{bb} - \mathbf{t}_{ba}]^{\mathrm{T}},$$

$$\mathbf{x}_C : [(\mathbf{A}_{aa} - \mathbf{A}_{ab})(\mathbf{A}_{bb} - \mathbf{A}_{ba}), (\mathbf{A}_{aa} - \mathbf{A}_{ab}) + (\mathbf{A}_{bb} - \mathbf{A}_{ba}),$$
$$(\mathbf{t}_{aa} - \mathbf{t}_{ab})(\mathbf{t}_{bb} - \mathbf{t}_{ba}), (\mathbf{t}_{aa} - \mathbf{t}_{ab}) + (\mathbf{t}_{bb} - \mathbf{t}_{ba})]^{\mathrm{T}},$$

Here, A_{aa} denotes the row vector representing the samples of $\mathbf{A}_{aa}(\alpha)$ at the 58 scan angles. The first feature vector $\mathbf{x}_A$ is taken as the original form of the scans, except for averaging the cross terms. The choice of the second feature vector $\mathbf{x}_B$ has been motivated by the TDA reviewed in Section 3. The third feature vector $\mathbf{x}_C$ is motivated by the differential terms which are used to assign belief values to the target types in D-S evidential reasoning and majority voting [14] discussed in Section 4. Note that the dimensionalities d of these vector representations are $348(=6 \times 58), 232(=4 \times 58)$, and 232, respectively.

We associate a class w_i with each target ($i = 1, \ldots, c$). An unknown target is assigned to class w_i if its *feature vector* $\mathbf{x} = (x_1, \ldots, x_d)^{\mathrm{T}}$ falls in the region Ω_i. A rule which partitions the decision space into regions Ω_i, $i = 1, \ldots, c$ is called a *decision rule*. Each one of these regions corresponds to a different target type. Boundaries between these regions are called *decision surfaces*. Let $p(w_i)$ be the a priori probability of a target belonging to class w_i. To classify a target with feature vector $\mathbf{x}$, the a posteriori probabilities $p(w_i|\mathbf{x})$ are compared and the target is classified into class w_j if $p(w_j|\mathbf{x}) > p(w_i|\mathbf{x})\ \forall i \neq j$. This is known as *Bayes minimum error rule*. However, since these a posteriori probabilities are rarely known, they need to be estimated. A more convenient formulation of this rule can be obtained by using Bayes' theorem: $p(w_i|\mathbf{x}) = p(\mathbf{x}|w_i)p(w_i)/p(\mathbf{x})$ which results in $p(\mathbf{x}|w_j)p(w_j) > p(\mathbf{x}|w_i)p(w_i)\ \forall i \neq j \Rightarrow \mathbf{x} \in \Omega_j$ where $p(\mathbf{x}|w_i)$ are the class-conditional probability density functions (CCPDFs) which are also unknown and need to be estimated in their turn using the training set. The training set consists of several sample feature vectors $\mathbf{x}_n$, $n = 1, \ldots, N_i$ which all belong to the same class w_i, for a total of $N_1 + N_2 + \cdots + N_c = N$ sample feature vectors. The *test set* is then used to evaluate the performance of the decision rule used. This decision rule can be generalized as

$q_j(\mathbf{x}) > q_i(\mathbf{x})\ \forall i \neq j \Rightarrow \mathbf{x} \in \Omega_j$ where the function q_i is called a *discriminant function*.

The various statistical techniques for estimating the CCPDFs from the training set are often categorized as non-parametric and parametric. In non-parametric methods, no assumptions on the parametric form of the CCPDFs are made; however, this requires large training sets. This is because any non-parametric PDF estimate based on a finite sample is biased [23]. There are four major types of non-parametric PDF estimators: histogram, kernel estimator, k-nearest neighbor, and series methods. In parametric methods, specific models for the CCPDFs are assumed and then the parameters of these models are estimated. These parametric methods can be categorized as normal and non-normal models. The most commonly used parametric estimation technique is the maximum likelihood estimator.

6.1. Kernel estimator (KE)

KE is a family of PDF estimators first proposed by Fix and Hodges in 1951 [24]. In the KE method, the CCPDF estimates $\hat{p}(\mathbf{x}|w_i)$ are of the form

$$\hat{p}(\mathbf{x}|w_i) = \frac{1}{N_i h_i^d} \sum_{n=1}^{N_i} K\left(\frac{\mathbf{x}-\mathbf{x}_n}{h_i}\right), \tag{13}$$

where $\mathbf{x}$ is the d-dimensional feature vector at which the estimate is being made and $\mathbf{x}_n$, $n = 1, \ldots, N_i$ are the training set sample feature vectors associated with class w_i. Here, h_i is called the spread or smoothing parameter or the *bandwidth* of the KE, and $K(\mathbf{z})$ is a kernel function which satisfies the conditions $K(\mathbf{z}) \geqslant 0$ and $\int K(\mathbf{z})\,\mathrm{d}\mathbf{z} = 1$.

In this method, the selection of the bandwidth h_i is important. If h_i is selected too small, $\hat{p}(\mathbf{x}|w_i)$ degenerates into a collection of N_i sharp peaks, each located at a sample feature vector. On the other hand, if h_i is selected too large, the estimate is oversmoothed and an almost uniform CCPDF results. Usually, h_i is chosen as a function of N_i such that $\lim_{N_i \to \infty} h(N_i) = 0$. There are various approaches to select h_i if a constant h_i is to be used [25,26].

In the implementation of this method, we employed a d-dimensional Gaussian kernel function. Consider the $N_i = 100$ sample feature vectors corresponding to the ith class. The bandwidth h_i for each class is pre-computed based on the training data as follows: The distance between each of these vectors and its qth nearest neighbor in the same class is found and their average is taken. This is repeated for $1 \leqslant q \leqslant 10$ for all classes. Then, the vectors in the training set are used as test vectors to compute the average misclassification rate for each value of q. The average distances (for each class i) corresponding to the value of q minimizing the misclassification rate (in our case, $q = 4$), are chosen as the values of h_i. After h_i's are computed, a test feature vector $\mathbf{x}$ is classified into that class for which the CCPDF in Eq. (13) is maximized. This requires the training data to be stored throughout testing.

6.2. k-Nearest neighbor (k-NN) method

Consider the k nearest neighbors of a feature vector $\mathbf{x}$ in a set of several feature vectors. Suppose k_i of these k vectors come from class w_i. Then, a k-NN estimator for class w_i can be defined as $\hat{p}(w_i|\mathbf{x}) = k_i/k$, and $\hat{p}(\mathbf{x}|w_i)$ can be obtained from $\hat{p}(\mathbf{x}|w_i)\hat{p}(w_i) = \hat{p}(w_i|\mathbf{x})\hat{p}(\mathbf{x})$. This results in a classification rule such that $\mathbf{x}$ is classified into class w_j if $k_j = \max_i(k_i)$. In other words, the k nearest neighbors of the vector $\mathbf{x}$ in the training set are considered and the vector $\mathbf{x}$ is classified into the same class as the majority of its k nearest neighbors. A major disadvantage of this method is that a pre-defined rule for the selection of the value of k does not exist.

The so-called *generalized* k-NN estimator is related to the KE. Letting $r_k(\mathbf{x})$ be the Euclidean distance from $\mathbf{x}$ to the kth nearest neighbor of $\mathbf{x}$ in the training set, it is defined as [28]

$$\hat{p}(\mathbf{x}|w_i) = \frac{1}{N_i r_k^d(\mathbf{x})} \sum_{n=1}^{N_i} K\left(\frac{\mathbf{x}-\mathbf{x}_n}{r_k(\mathbf{x})}\right). \tag{14}$$

Note that this is similar to Eq. (13) for the KE. The main difference between the KE and the generalized k-NN estimator is that here, the bandwidth $r_k(\mathbf{x})$ is a function of $\mathbf{x}$ instead of being constant for each class as in the KE.

In the implementation of the k-NN and the generalized k-NN methods, k values varying between 1 and 10 have been considered. We present results for $k = 1$ which is the value giving the best results. Again, the training data must be stored during testing.

6.3. Parameterized density estimation (PDE)

In this method, the CCPDFs are assumed to be d-dimensional normal:

$$p(\mathbf{x}|w_i) = \frac{1}{(2\pi)^{(d/2)}|\boldsymbol{\Sigma}_i|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^{\mathrm{T}}\boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\right], \quad i = 1, \ldots, c, \tag{15}$$

where the $\boldsymbol{\mu}_i$'s denote the class means, and the $\boldsymbol{\Sigma}_i$'s denote the class-covariance matrices, both of which must be estimated based on the training set. The most commonly used estimation technique is the maximum likelihood estimator [29] which is also used in this study.

In PDE, d-dimensional homoscedastic and heteroscedastic normal models are used for the CCPDFs. In the homoscedastic case, the covariance matrices are selected equal for all classes, usually taken as a weighted average of the individual class-covariance matrices: $\sum_{i=1}^{c}(N_i/N)\hat{\boldsymbol{\Sigma}}_i$ [30]. In the heteroscedastic case, they are individually calculated for each class.

In this study, both homoscedastic and heteroscedastic normal models have been implemented to estimate the means and the covariances of the CCPDF for each class (i.e., target type) using the maximum likelihood estimator, for each of

the three feature vector representations. Then, the test feature vector is classified into the class for which Eq. (15) is maximum.

6.4. Linear discriminant analysis (LDA)

To describe this method, we first consider the case where there are only two targets. Let the training set include N_1 feature vectors which are obtained from the first target and thus which should remain in the region associated with class w_1. Similar considerations apply for the N_2 feature vectors obtained from the second target. We wish to choose a weight vector $\mathbf{a} = (a_0, a_1, \ldots, a_d)^{\mathrm{T}}$ so that the plane defined by $\mathbf{a}^{\mathrm{T}}\mathbf{z}$ divides the d-dimensional space into two, such that the rate of misclassification is minimized. Here $\mathbf{z} = (1, \mathbf{x}^{\mathrm{T}})^{\mathrm{T}}$ is the augmented feature vector in terms of which the linear discriminant function is defined as

$$q(\mathbf{x}) = = a_0 + \sum_{l=1}^{d} a_l x_l = \mathbf{a}^{\mathrm{T}}\mathbf{z}. \tag{16}$$

We require

(i) $q(\mathbf{x}_n) = \mathbf{a}^{\mathrm{T}}\mathbf{z}_n > 0$, whenever $\mathbf{x}_n$ is a sample feature vector from class w_1,
(ii) $q(\mathbf{x}_n) = \mathbf{a}^{\mathrm{T}}\mathbf{z}_n < 0$, whenever $\mathbf{x}_n$ is a sample feature vector from class w_2.

To reduce the problem to a single equation, we define a new vector $\mathbf{y}_n$ such that:

(i) $\mathbf{y}_n \triangleq \mathbf{z}_n$, whenever $\mathbf{x}_n$ is a sample feature vector from class w_1, and
(ii) $\mathbf{y}_n \triangleq -\mathbf{z}_n$, whenever $\mathbf{x}_n$ is a sample feature vector from class w_2.

Now, the above two conditions are reduced to the single condition $\mathbf{a}^{\mathrm{T}}\mathbf{y}_n > 0$, $\forall n$, $n = 1, \ldots, N$ where $N = N_1 + N_2$. The decision surface is the hyperplane $\mathbf{a}^{\mathrm{T}}\mathbf{y}_n = 0$. Unless these two classes are linearly separable, a weight vector $\mathbf{a}$ which satisfies the above condition cannot be found. Therefore, we aim to satisfy $\mathbf{a}^{\mathrm{T}}\mathbf{y}_n > 0$ as much as possible. There are various criteria to find the linear surface which best discriminates two classes. Two of the most widely used ones are the perceptron criterion and Fisher's criterion [27].

A third approach is to seek a weight vector $\mathbf{a}$ that satisfies $\mathbf{a}^{\mathrm{T}}\mathbf{y}_n = b_n$ as closely as possible in the least-squares sense where b_n's are positive constants whose choice is discussed below. This set of N equations can be put in standard matrix form $\mathbf{Ya} = \mathbf{b}$ with $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N)^{\mathrm{T}}$ being an $N \times (d+1)$ matrix and $\mathbf{b} = (b_1, \ldots, b_N)^{\mathrm{T}}$. For a given $\mathbf{b}$, the value of $\mathbf{a}$ which minimizes $(\mathbf{Ya} - \mathbf{b})^{\mathrm{T}}(\mathbf{Ya} - \mathbf{b})$ is given by $\mathbf{a} = (\mathbf{Y}^{\mathrm{T}}\mathbf{Y})^{-1}\mathbf{Y}^{\mathrm{T}}\mathbf{b}$.

In generalizing the LDA from two to c classes, we used $c-1$ two-class decision rules, each one separating Ω_i, $i = 1, \ldots, c-1$ from *all* Ω_j, $j = 1, \ldots, c$ where $j \neq i$. First, the $\mathbf{a}$ vectors separating each class from all the others are calculated using the least-squares approach, with $\mathbf{b}$ chosen as $[(N/N_1)\mathbf{u}_1, (N/N_2)\mathbf{u}_2]^{\mathrm{T}}$, where $\mathbf{u}_1$ and $\mathbf{u}_2$ are row vectors of N_1 and N_2 ones, respectively. The least-squares approach with this choice of $\mathbf{b}$ results in exactly the same solution obtained with Fisher's criterion. Then, for each test vector $\mathbf{x}$, Eq. (16) is evaluated and the vector is classified into that class for which $q(\mathbf{x})$ takes a positive value.

6.5. Fuzzy c-means clustering (FCC) algorithm

Clustering tries to identify the relationships among patterns in the training data set by organizing the patterns into a number of clusters, where the patterns in each cluster show a certain degree of closeness or similarity. In *hard clustering*, cluster boundaries are assumed to be well defined so that each feature vector in the data set belongs to one of the clusters with a degree of membership equal to one. However, this type of clustering may not be suitable when the cluster boundaries are not well defined. In such cases, *fuzzy clustering* is more useful where a feature vector $\mathbf{x}$ is assigned to each cluster i with a degree of membership $\mu_i(\mathbf{x}) \in [0, 1]$. It is possible to use fuzzy clustering as the basis for hard clustering, by assigning feature vector $\mathbf{x}$ to cluster j (in the hard sense) if $\mu_j(\mathbf{x}) \geqslant \mu_i(\mathbf{x})$, $\forall i = 1, \ldots, c$ where $c \geqslant 2$ is the total number of clusters. However, it should be noted that these sets may not be disjoint when more than one maximum exists.

The FCC algorithm has been developed by Dunn [31] and extended by Bezdek [32]. It minimizes the following objective function with respect to fuzzy memberships $\mu_i(\mathbf{x}_j)$ and cluster centers $\mathbf{v}_i$:

$$J_\gamma = \sum_{i=1}^{c}\sum_{j=1}^{N} \mu_i^\gamma(\mathbf{x}_j) \| \mathbf{x}_j - \mathbf{v}_i \|_{\mathbf{A}}^2 \quad \text{where } \|\mathbf{x}\|_{\mathbf{A}}^2 = \mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{x}, \tag{17}$$

where $\mathbf{A}$ is a $d \times d$ positive definite matrix, d is the dimension of the input patterns $\mathbf{x}_j$, N is the total number of training feature vectors, and $\gamma > 1$ is the weighting exponent for $\mu_i(\mathbf{x}_j)$ which controls the fuzziness of the resulting clusters. In this study, we have taken $\mathbf{A}$ as a $d \times d$ identity matrix and $\gamma = 1.3$. The FCC algorithm can be summarized as [32]:

(1) Initialize the memberships $\mu_i(\mathbf{x}_j)$'s such that $\sum_{i=1}^{c} \mu_i(\mathbf{x}_j) = 1$, $j = 1, \ldots, N$.
(2) Compute the cluster center $\mathbf{v}_i$ for $i = 1, \ldots, c$ using $\mathbf{v}_i = \sum_{j=1}^{N} \mu_i^\gamma(\mathbf{x}_j)\mathbf{x}_j / \sum_{j=1}^{N} \mu_i^\gamma(\mathbf{x}_j)$.
(3) Update the memberships $\mu_i(\mathbf{x}_j)$ using $\mu_i(\mathbf{x}_j) = (\|\mathbf{x}_j - \mathbf{v}_i\|_A^2)^{-1/(\gamma-1)} / \sum_{k=1}^{c} (\|\mathbf{x}_j - \mathbf{v}_k\|_{\mathbf{A}}^2)^{-1/(\gamma-1)}$.
(4) Repeat the second and third steps until the value of J_γ no longer decreases.

Then, a test feature vector $\mathbf{x}$ is classified into the class for which $\mu_i(\mathbf{x})$ is maximum.

7. Artificial neural networks (ANNs)

ANNs have been widely used in areas such as target detection and classification [33], speech processing [34], system identification [35], control theory [36], medical applications [37], and character recognition [38]. In this study, ANNs are employed to identify and resolve parameter relations embedded in the characteristics of sonar echo returns from all seven target types considered, for their differentiation and localization in a robust manner in real time. ANNs consist of an input layer, one or more hidden layers to extract progressively more meaningful features, and a single output layer, each comprised of a number of units called neurons. The model of each neuron includes a smooth nonlinearity, here a sigmoid function of the form $\varphi(v) = (1 + \mathrm{e}^{-v})^{-1}$. Due to the presence of distributed nonlinearity and a high degree of connectivity, theoretical analysis of ANNs is difficult. These networks are trained to compute the boundaries of decision regions in the form of connection weights and biases by using training algorithms. Performance of ANNs is affected by the choice of parameters related to the network structure, training algorithm, and input signals, as well as parameter initialization [39]. In this study, two training algorithms are employed, namely, back-propagation (BP) and generating-shrinking (GS) algorithms.

With the BP algorithm, a set of *training patterns* is presented to the network and the error between the resulting signal at the output and the desired signal is minimized with a gradient-descent procedure. The two adjustment parameters of the algorithm, namely the learning rate and the momentum constant [40] are chosen to be 0.01 and 0.9, respectively, and training with the BP algorithm is stopped either when the average error is reduced to 0.001 or if a maximum of 10,000 epochs is reached, whichever occurs earlier. The second case occurs very rarely. The number of hidden-layer neurons is determined by enlarging [41].

The GS algorithm first builds and then shrinks or prunes a feed-forward neural network, offering fast convergence rates and 100% correct classification on the training set [42]. The network used in Ref. [42] consists of two hidden layers with equal numbers of neurons, initially set equal to the number of training patterns. Pre-determined initial connection weights are assigned, with the consequence that the generalization behavior of the network is analytically known. Then, the hidden layers are pruned while preserving 100% correct classification on the training set. Only one output neuron takes the value one (the winning neuron) and the remaining output neurons take the value zero. At the input layer, a pre-fixed reference number $n_r \in (0, \infty)$ is used as an additional input to control the generalization capability of the network. The algorithm achieves scale-invariant generalization behavior as n_r approaches zero, and behaves like a nearest-neighborhood classifier as it tends to infinity. We employ the relatively small value $n_r = 0.01$ in order to enhance scale invariance. A comparison with the BP algorithm [42] indicates that the GS algorithm does not have the convergence problems of the BP algorithm and has several hundred times faster convergence rate and improved generalization capability.

7.1. Pre-processing of the input signals

The results obtained depend on which form the observed signals are presented to the ANNs. Therefore, we have considered several different pre-processing techniques.

7.1.1. Ordinary Fourier transform

The Fourier transform is widely used in signal processing to study the spectral behavior of a signal. The discrete Fourier transform (DFT) of a signal $f(n)$ is defined as

$$F(k) = \mathscr{F}\{f(n)\} \triangleq \frac{1}{N}\sum_{n=0}^{N-1} f(n)\mathrm{e}^{-\mathrm{i}2\pi nk/N}, \tag{18}$$

where N is the length of the discrete signal $f(n)$.

7.1.2. Fractional Fourier transform

The ath-order fractional Fourier transform is a generalization of the ordinary Fourier transform such that the first-order fractional Fourier transform is the ordinary Fourier transform and the 0th-order fractional Fourier transform corresponds to the function itself [43]. The transform has been studied extensively since the early 1990s with applications in wave propagation and optics [44–47], time-frequency analysis, pattern recognition, and digital signal [48,49] and image processing [50,51]. Most applications are based on replacing the ordinary Fourier transform with the fractional transform. Since the latter has an additional degree of freedom (the order parameter a), it is often possible to generalize and improve upon previous results. The ath-order fractional Fourier transform $f_a(u)$ of $f(u)$ is defined for $0 < |a| < 2$ as [49]

$$f_a(u) \triangleq \int_{-\infty}^{\infty} A_\phi \exp[\mathrm{i}\pi(u^2\cot\phi - 2uu'\csc\phi + u'^2\cot\phi)] f(u')\,\mathrm{d}u',$$

where

$$A_\phi = \frac{\exp[-\mathrm{i}(\pi\,\mathrm{sgn}(\phi)/4 - \phi/2)]}{|\sin\phi|^{1/2}} \quad \text{and} \quad \phi = \frac{a\pi}{2}. \tag{19}$$

The $f_a(u)$ approaches $f(u)$ and $f(-u)$ as a approaches 0 and ± 2, respectively, and is defined as such at these values. The fractional Fourier transform reduces to the ordinary Fourier transform when $a = 1$. The transform is linear and index additive: the a_1th-order transform of the a_2th-order transform is equal to the $(a_1 + a_2)$ th-order transform. Digital implementation of the fractional Fourier transform is as efficient as that of the ordinary Fourier transform; it can also be computed in the order of $N \log N$ time [43].

With a similar notation as in the case of DFT, the ath-order discrete fractional Fourier transform (DFRT) of **f**, denoted

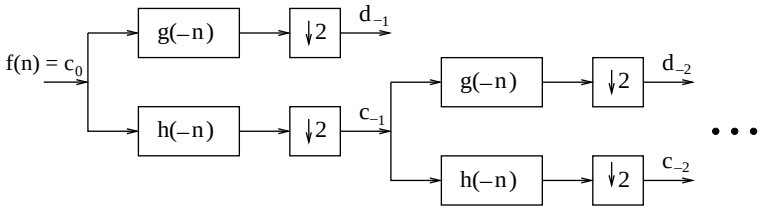

Fig. 5. Block diagram of the DWT. The square boxes represent down-sampling.

$\mathbf{f_a}$, can be expressed as $\mathbf{f_a} = \mathbf{F^a f}$ where $\mathbf{F^a}$ is the $N \times N$ DFRT matrix which corresponds to the ath power of the ordinary DFT matrix $\mathbf{F}$ and $\mathbf{f}$ is an $N \times 1$ column vector [52]. However, we note that there are certain subtleties and ambiguities in defining the power function [52].

7.1.3. Hartley transform

Hartley transform [53] is a widely used technique in signal processing applications such as image compression [54] and adaptive filtering [55]. The discrete Hartley transform (DHT) of $f(n)$ is defined as

$$H(k) = \mathscr{H}\{f(n)\} \triangleq \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} f(n) \operatorname{cas}\left(\frac{2\pi}{N} nk\right), \qquad (20)$$

where $cas(x) \triangleq \cos(x) + \sin(x)$. If the DFT of a signal $f(n)$ is expressed as $F(k) = F_R(k) - \mathrm{i}F_I(k)$, then its DHT is given by $H(k) = F_R(k) + F_I(k)$. The DHT can also be represented in matrix notation as $\mathbf{h}_1 = \mathbf{Hf}$, where $\mathbf{H}$ is the $N \times N$ DHT matrix, and $\mathbf{h}_1$ is the DHT of $\mathbf{f}$.

7.1.4. Wavelet transform

We describe the discrete wavelet transform (DWT) [56] by referring to Fig. 5, where the operations performed on the input signal $f(n)$ of length N are shown as a block diagram. $h(-n)$ and $g(-n)$ are referred to as the *scaling filter* and the *wavelet filter*, respectively, where $g(n) \triangleq (-1)^n h(M - n - 1)$. Mathematically,

$$c_j(k) = \sum_m h(m - 2k) c_{j+1}(m),$$

$$d_j(k) = \sum_m g(m - 2k) c_{j+1}(m), \quad k = 0, 1, \ldots, (2^j N - 1) \quad \text{and} \quad j = -1, -2, \ldots, \qquad (21)$$

where for $j = -1$ we associate $c_0(.)$ with $f(.)$ and these equations describe the left part of Fig. 5. When $j = -2$, they describe the right part of the same figure. More generally, these equations allow us to obtain the coefficients at scale j from the coefficients at scale $j + 1$. We have employed the value $M = 23$ and the scaling filter whose coefficients $h(n)$ are given below:

$$\begin{array}{lllllll} h(n)= & [-0.002 & -0.003 & 0.006 & 0.006 & -0.013 \\ 0.012 & -0.030 & 0.023 & -0.078 & -0.035 & 0.307 \\ 0.542 & 0.307 & -0.035 & -0.078 & 0.023 & -0.030 \\ 0.012 & -0.013 & 0.006 & 0.006 & -0.003 & -0.002] \end{array}$$

for $n = 0, \ldots, M - 1$. This filter is known as the Lamaire wavelet [57]. After down-sampling, the total number of samples in the concatenation of c_j and d_j is equal to the number of samples of c_{j+1}. In principle, the concatenation of c_j and d_j for any resolution level $j = -1, -2, \ldots$ can be used as an input to the neural network. However, values of j further than -2 were not found to be advantageous in our implementations as discussed later.

7.1.5. Self-organizing feature map

Self-organizing ANNs are generated by unsupervised learning algorithms that have the ability to form an internal representation of the network, modeling the underlying structure of the input data. These networks are commonly used to solve the scale-variance problem encountered in supervised learning. However, it is not recommended to use them by themselves for pattern classification or other decision-making processes [41]. Best results are achieved with these networks when they are used as feature extractors prior to a linear classifier or a supervised learning procedure. The most commonly used algorithm for generating self-organizing ANNs is Kohonen's self-organizing feature-mapping (KSOFM) algorithm [58]. In this algorithm, weights are adjusted from the input layer towards the output layer where the output neurons are interconnected with local connections. These output neurons are geometrically organized in one, two, three, or even higher dimensions. This algorithm can be summarized as follows: (i) initialize the weights randomly, (ii) present new input from the training set, (iii) find the winning neuron at the output layer, (iv) select the neighborhood of this output neuron, (v) update weights from input towards selected output neurons, (vi) continue with the second step until no considerable changes in the weights occur (see Ref. [41] for further details).

7.2. Input signals

In this work, many different signal representations are considered as alternative inputs to the ANNs. In addition to the pre-processing methods discussed, different combinations of the amplitude and TOF patterns are also considered. Specifically, we employed the following 30 alternative inputs to the ANNs:

I_1: samples of $A_{aa}(\alpha), A_{bb}(\alpha), [A_{ab}(\alpha) + A_{ba}(\alpha)]/2$, $t_{aa}(\alpha), t_{bb}(\alpha)$, and $[t_{ab}(\alpha) + t_{ba}(\alpha)]/2$,

I_2: samples of $A_{aa}(\alpha) - A_{ab}(\alpha), A_{bb}(\alpha) - A_{ba}(\alpha)$, $t_{aa}(\alpha) - t_{ab}(\alpha)$, and $t_{bb}(\alpha) - t_{ba}(\alpha)$,

I_3: samples of $[A_{aa}(\alpha) - A_{ab}(\alpha)][A_{bb}(\alpha) - A_{ba}(\alpha)], [A_{aa}(\alpha) - A_{ab}(\alpha)] + [A_{bb}(\alpha) - A_{ba}(\alpha)]$

$[t_{aa}(\alpha) - t_{ab}(\alpha)][t_{bb}(\alpha) - t_{ba}(\alpha)]$,

and $[t_{aa}(\alpha) - t_{ab}(\alpha)] + [t_{bb}(\alpha) - t_{ba}(\alpha)]$,

$I_4 - I_{12}$: DFT of I_1, I_2, I_3, its low-frequency component (LFC), and its magnitude

$(\mathscr{F}(I_i), \mathrm{LFC}(\mathscr{F}(I_i)), |\mathrm{LFC}(\mathscr{F}(I_i))|,\ i = 1, 2, 3)$,

$I_{13} - I_{15}$: DFRT of I_1, I_2, I_3 at different orders $(\mathscr{F}^a(I_i),\ i = 1, 2, 3)$,

$I_{16} - I_{18}$: DHT of I_1, I_2, I_3 $(\mathscr{H}(I_i),\ i = 1, 2, 3)$,

$I_{19} - I_{27}$: DWT of I_1, I_2, I_3 and its low-frequency components at different resolutions,

$(\mathrm{DWT}(I_i), \mathrm{LFC}(\mathrm{DWT}(I_i))_1, \mathrm{LFC}(\mathrm{DWT}(I_i))_2,$ $i = 1, 2, 3)$,

$I_{28} - I_{30}$: features extracted by using KSOFM (KSOFM(I_i), $i = 1, 2, 3$).

The sampled sequences I_1, I_2, I_3 correspond to the feature vectors $\mathbf{x}_A, \mathbf{x}_B$, and $\mathbf{x}_c$ defined and used in Section 6 for statistical pattern recognition techniques. Here, they have been used both in their raw form and after taking their discrete ordinary and fractional Fourier, Hartley, and wavelet transforms, as well as after feature extraction by KSOFM. The transforms are performed on the six parts of I_1 and the four parts of I_2 and I_3, separately.

DWTs of each signal at different resolution levels j have been considered. Initially, DWT of each signal at resolution level $j = -1$ is used as the input: DWT(I_i), $i = 1, 2, 3$. Secondly, only the low-frequency component of the DWT, the c_{-1}'s, are employed: LFC(DWT(I_i))$_1$. Finally, the low-frequency component of DWT at resolution $j = -2$, the c_{-2}'s, are used: LFC(DWT(I_i))$_2$. Use of the low-frequency components helps eliminate high-frequency noise. However, more negative values of j, which correspond to fewer samples of c_j and d_j, and thus lower resolutions, lead to deterioration in the performance of the network beyond $j = -2$. The value $j = -2$ corresponds to the frequency-domain information between 0 and $\pi/4$ of the original patterns. To make a fair comparison, the low-frequency component of the DFT, LFC$(\mathscr{F}(I_i))$, corresponding to the same frequency interval as LFC(DWT(I_i))$_2$ is also considered. We also employed the magnitude of the low-frequency component of the DFT, $|\mathrm{LFC}(\mathscr{F}(I_i))|$. The ath-order DFRTs of the three input signal representations, for values of a varying from 0.05 to 0.95 with 0.05 increments have been considered. The features extracted by using KSOFM are used both prior to ANNs trained with the two training algorithms and prior to linear classifiers designed by using a least-squares approach.

Initially, a single integrated ANN is trained by using the BP algorithm to both classify and localize the targets for each of the above input signals. Next, modular network structures for each type of input signal have been considered in which three separate networks for target type, range, and azimuth, each trained with the BP algorithm, are employed. Neural networks using the same input signal representations are also trained with the GS algorithm. This algorithm can only be applied for target type classification since here only one output neuron takes the value one (the winning neuron) and the others are zero. For this reason, range and azimuth estimation cannot be made with this approach [12].

8. Results

All of the methods considered determine the target type and estimate its range and azimuth except statistical pattern recognition techniques and ANNs trained with the GS algorithm.

For TDA, D-S, and voting, the resulting average percentages over all target types for correct classification, correct range and azimuth estimation are given in Table 1 for test sets I–III. A range or azimuth estimate is considered correct if it is within an error tolerance of ε_r of the actual range or ε_θ of the actual azimuth. Use of preference orders and assignment of reliability measures always brings some improvement compared to the results of SMV. The fifth reliability measure gives the highest percentage of correct differentiation, and is followed by the third, fourth, first, and second measures. These five reliability measures always result in better classification performance than a uniform reliability measure assignment. In addition, their performances are also better than that of D-S evidential reasoning which is in turn better than TDA. The results associated with test set II are about 3–4% worse than with test set I. Those obtained with test set III are about 0–2% lower. Note that these methods do not involve training; therefore the training data are not used. In general, the azimuth estimation results are slightly better than the range estimation results.

For the statistical pattern recognition techniques, the resulting average percentages of correct classification over all target types for the three test sets are given in Table 2. In this table, the percentage of correct classification presented for k-NN and generalized k-NN are the best results over different k values ($1 \leqslant k \leqslant 10$). With few and insignificant

Table 1
The percentages of correct classification, range (r) and azimuth (θ) estimation for TDA, D–S, SMV, and majority voting schemes employing preference ordering without/with reliability measures for the three test sets (I–II–III)

Method	% of correct classif.	% of correct r estimation				% of correct θ estimation			
		Error tolerance ε_r				Error tolerance ε_θ			
		± 0.125 cm	± 1 cm	± 5 cm	± 10 cm	$\pm 0.25^\circ$	$\pm 2^\circ$	$\pm 10^\circ$	$\pm 20^\circ$
TDA	61-57-61	16-16-16	36-35-35	72-60-62	80-77-74	19-19-19	41-41-40	59-60-56	95-95-86
D-S	89-85-87	17-20-16	36-39-35	72-61-62	80-77-74	32-31-26	61-56-54	98-98-99	98-98-99
SMV	82-79-80	16-16-16	36-36-35	72-60-62	80-77-74	19-19-19	41-42-40	61-62-56	98-98-86
$\mathrm{rel}_\alpha = 1$	88-84-85	16-16-16	36-36-35	72-60-62	80-77-74	19-19-19	41-42-40	61-62-56	98-98-86
rel_α^1	90-86-88	29-28-23	48-47-43	82-83-72	89-85-81	32-31-26	61-56-54	98-98-99	98-98-99
rel_α^2	90-86-88	29-28-23	48-47-43	82-83-72	89-85-81	32-31-26	61-56-54	98-98-99	98-98-99
rel_α^3	92-88-91	17-20-16	36-40-35	72-61-62	80-77-74	20-23-19	44-48-40	67-78-56	97-97-86
rel_α^4	91-87-89	17-20-16	36-39-35	72-61-62	80-77-74	20-23-19	42-47-40	63-77-56	97-96-86
rel_α^5	94-91-92	16-16-16	36-35-35	72-60-62	80-77-74	19-19-19	41-41-40	59-60-56	96-96-86

Table 2
The percentage of correct classification with the three alternative feature vectors $\mathbf{x}_A$, $\mathbf{x}_B$, $\mathbf{x}_C$ for different statistical target recognition for the three test sets (I–II–III)

Method	$\mathbf{x}_A$	$\mathbf{x}_B$	$\mathbf{x}_C$
KE	99-93-71	99-89-68	94-78-65
Ordinary k-NN	97-83-70	98-74-67	91-67-63
Generalized k-NN	99-95-71	99-90-69	99-82-67
PDE (homoscedastic NM)	76-74-57	71-63-61	62-54-56
PDE (heteroscedastic NM)	81-79-66	79-74-59	68-64-62
LDA	71-56-50	58-39-41	57-42-41
FCC	98-94-91	93-92-92	93-92-92

exceptions, $\mathbf{x}_A$ is seen to be the feature vector of choice, followed by $\mathbf{x}_B$ and $\mathbf{x}_C$ in that order. Thus narrowing our attention to $\mathbf{x}_A$, we observe that the best results are obtained with KE, generalized k-NN, and FCC (give or take one percentage point) for test sets I and II. However, for test set III, FCC is clearly superior.

In most cases, the percentages of correct classification obtained with the heteroscedastic normal model are slightly higher than those obtained with the homoscedastic normal model; however, these are both inferior to the methods mentioned above (expected since the superior methods are non-parametric in which no assumptions on the underlying PDFs are made, whereas in PDE the CCPDFs are assumed to be Gaussian, imposing an unnecessary restriction). Worst classification performance is obtained with LDA, indicating that the different functional forms of the amplitude and TOF patterns of the targets are not suitable for linear separation. The optimal results for test sets I, II, and III are 99%, 95%, and 92% respectively, showing that the degradation in performance for the latter test sets is not large.

As already mentioned, ANNs trained with the BP algorithm estimate the target type, range, and azimuth, whereas those trained with the GS algorithm determine only the target type. For non-modular and modular networks trained with the BP algorithm, the resulting average percentages over all target types for correct type classification, correct range and azimuth estimation are given in Table 3. In this three-part table, the numbers before the parentheses are for non-modular networks, whereas the numbers in the parentheses are for modular networks. For the DFRT, results are given for the corresponding optimal value of a [59]. For test set I, the highest average percentage of correct classification of 100% is obtained with the input signal $\mathscr{F}^a(I_1)$ for non-modular networks, and 99% with $\mathrm{LFC}(\mathrm{DWT}(I_1))_2$ for modular networks. For non-modular networks, the highest average percentages of correct range estimation lie in the range 79–97% as the error tolerance ε_r varies between 0.125–10 cm. The optimal pre-processing method is one of $I_3, \mathscr{F}^a(I_1)$, or $\mathscr{F}(I_1)$. The highest average percentages of correct azimuth estimation lie in the range 93–100% as the error tolerance ε_θ varies between 0.25° and 20°. The optimal pre-processing method is usually $\mathscr{F}^a(I_1)$ or $\mathrm{LFC}(\mathrm{DWT}(I_1))_2$. For modular networks, the highest average percentage of correct range estimation varies between 80% and 96% as ε_r varies between 0.125–10 cm. This is obtained with either $I_2, \mathscr{F}(I_1)$, or $\mathrm{LFC}(\mathrm{DWT}(I_1))_2$. The highest average percentage of correct azimuth estimation varies between 95% and 100% as the error tolerance level ε_θ varies between 0.25° and 20°. The optimal pre-processing method is one of $I_2, \mathscr{F}(I_1), \mathrm{LFC}(\mathscr{F}(I_1))$, or $\mathrm{LFC}(\mathrm{DWT}(I_1))_2$.

In general, straightforward use of DWT pre-processing does not offer any improvements with respect to no pre-processing. However, the low-frequency part of the DWT does offer better performance, with the resolution level ($j = -1$ or -2) to be used depending on whether we use I_1, I_2, or I_3. Employing the low-frequency part of the Fourier transform gives better classification and estimation performance than employing the whole Fourier transform for the input signals I_2 and I_3, while giving comparable

Table 3
Average percentages of correct classification, range (r) and azimuth (θ) estimation for ANNs trained with the BP algorithm. The three panels correspond to test sets I, II, and III, respectively

Input to ANN	% of correct classif.	% of correct r estimation				% of correct θ estimation			
		Error tolerance ε_r				Error tolerance ε_θ			
		± 0.125 cm	± 1 cm	± 5 cm	± 10 cm	$\pm 0.25°$	$\pm 2°$	$\pm 10°$	$\pm 20°$
Test set I									
I_1	88(88)	30(33)	41(46)	63(70)	86(87)	65(65)	76(72)	87(84)	97(97)
I_2	95(95)	74(73)	77(88)	87(93)	93(96)	89(95)	92(96)	95(97)	97(99)
I_3	86(88)	79(73)	82(75)	89(83)	94(91)	83(87)	89(91)	95(95)	97(98)
$\mathscr{F}(I_1)$	97(98)	64(72)	69(73)	86(87)	96(95)	86(94)	93(96)	96(98)	100(100)
LFC($\mathscr{F}(I_1)$)	96(97)	56(70)	64(73)	86(88)	95(97)	84(92)	90(96)	96(96)	100(99)
$\lvert$LFC($\mathscr{F}(I_1)$)$\rvert$	88(86)	28(45)	35(52)	68(77)	88(93)	65(55)	70(59)	86(79)	95(90)
$\mathscr{F}(I_2)$	93(89)	59(60)	64(65)	79(78)	89(90)	76(73)	81(86)	88(91)	93(96)
LFC($\mathscr{F}(I_2)$)	99(95)	63(68)	72(74)	85(86)	94(92)	91(89)	93(91)	96(96)	99(98)
$\lvert$LFC($\mathscr{F}(I_2)$)$\rvert$	86(95)	35(54)	42(60)	73(80)	96(94)	39(56)	50(65)	71(86)	86(95)
$\mathscr{F}(I_3)$	86(90)	54(62)	61(65)	77(77)	89(89)	70(77)	76(82)	85(88)	94(94)
LFC($\mathscr{F}(I_3)$)	91(85)	60(60)	68(65)	82(78)	92(90)	77(78)	81(83)	88(89)	96(96)
$\lvert$LFC($\mathscr{F}(I_3)$)$\rvert$	74(82)	34(41)	42(49)	65(72)	85(90)	30(53)	39(60)	62(78)	83(90)
$\mathscr{F}^a(I_1)$	100(96)	75(62)	79(66)	89(86)	97(96)	93(76)	96(79)	97(92)	100(99)
$\mathscr{F}^a(I_2)$	98(98)	67(68)	71(76)	83(87)	92(95)	80(86)	84(89)	90(95)	96(98)
$\mathscr{F}^a(I_3)$	90(93)	61(59)	68(62)	83(80)	92(90)	76(75)	82(79)	88(88)	95(94)
$\mathscr{H}(I_1)$	99(97)	59(54)	68(60)	85(81)	94(94)	84(84)	89(87)	95(95)	98(99)
$\mathscr{H}(I_2)$	98(97)	67(62)	72(68)	85(80)	93(90)	80(84)	85(86)	91(93)	96(99)
$\mathscr{H}(I_3)$	87(81)	59(46)	66(51)	80(69)	90(89)	73(79)	80(84)	89(90)	95(95)
DWT(I_1)	82(74)	15(21)	30(27)	59(59)	80(82)	46(51)	58(63)	77(80)	94(94)
LFC(DWT(I_1))$_1$	85(98)	18(21)	28(33)	58(59)	82(79)	54(59)	65(62)	80(79)	95(94)
LFC(DWT(I_1))$_2$	98(99)	71(80)	76(82)	87(91)	95(96)	90(92)	93(93)	97(98)	100(100)
DWT(I_2)	92(96)	63(64)	69(69)	84(82)	93(92)	85(87)	88(90)	93(94)	96(96)
LFC(DWT(I_2))$_1$	95(97)	65(66)	70(71)	84(84)	94(91)	87(88)	90(90)	94(94)	97(96)
LFC(DWT(I_2))$_2$	89(84)	28(32)	34(44)	58(68)	84(88)	58(53)	68(61)	86(80)	95(92)
DWT(I_3)	86(89)	58(58)	62(62)	76(76)	93(89)	85(76)	88(80)	93(88)	96(94)
LFC(DWT(I_3))$_1$	82(91)	56(61)	60(66)	75(78)	89(87)	73(79)	77(83)	86(89)	93(94)
LFC(DWT(I_3))$_2$	83(79)	29(33)	37(44)	63(69)	83(88)	53(41)	65(52)	78(75)	87(89)
KSOFM(I_1)	75(74)	17(14)	25(23)	49(46)	80(72)	64(61)	67(64)	81(79)	90(89)
KSOFM(I_2)	78(76)	22(19)	28(28)	59(57)	88(81)	69(66)	73(71)	86(85)	92(93)
KSOFM(I_3)	66(63)	24(21)	30(31)	57(55)	84(81)	51(49)	54(51)	78(75)	89(87)
Test set II									
I_1	88(88)	17(18)	32(30)	55(56)	78(83)	37(38)	47(47)	75(74)	91(94)
I_2	90(93)	59(60)	63(69)	78(83)	88(88)	70(71)	75(76)	92(97)	94(98)
I_3	58(59)	63(60)	63(62)	76(76)	83(85)	66(69)	74(73)	93(93)	94(97)
$\mathscr{F}(I_1)$	96(98)	53(57)	54(57)	81(75)	91(88)	69(72)	77(77)	89(98)	98(98)
LFC($\mathscr{F}(I_1)$)	96(97)	52(59)	58(62)	82(83)	89(89)	69(69)	75(74)	83(83)	98(98)
$\lvert$LFC($\mathscr{F}(I_1)$)$\rvert$	86(82)	20(37)	28(45)	64(72)	86(88)	57(53)	66(59)	78(74)	88(88)
$\mathscr{F}(I_2)$	89(92)	52(51)	53(52)	67(68)	80(80)	60(59)	65(68)	81(92)	83(95)
LFC($\mathscr{F}(I_2)$)	98(95)	54(56)	57(58)	74(70)	83(80)	69(69)	72(73)	95(90)	97(92)
$\lvert$LFC($\mathscr{F}(I_2)$)$\rvert$	83(90)	21(42)	31(50)	64(74)	86(92)	39(51)	49(60)	71(75)	81(87)
$\mathscr{F}(I_3)$	84(87)	48(51)	52(53)	65(68)	77(80)	57(60)	63(65)	82(84)	89(86)
LFC($\mathscr{F}(I_3)$)	90(85)	56(53)	56(54)	74(73)	85(85)	61(62)	65(67)	87(86)	91(91)
$\lvert$LFC($\mathscr{F}(I_3)$)$\rvert$	74(81)	25(36)	34(43)	57(60)	80(86)	30(48)	39(56)	62(78)	81(87)
$\mathscr{F}^a(I_1)$	100(96)	59(53)	60(55)	79(79)	89(88)	70(63)	75(68)	97(97)	100(99)
$\mathscr{F}^a(I_2)$	92(92)	55(56)	55(59)	67(71)	78(83)	62(65)	67(68)	85(91)	90(92)
$\mathscr{F}^a(I_3)$	83(85)	53(52)	53(54)	72(71)	81(79)	61(60)	70(65)	85(80)	89(88)
$\mathscr{H}(I_1)$	92(96)	52(51)	55(54)	76(77)	87(89)	68(67)	74(73)	93(95)	96(99)
$\mathscr{H}(I_2)$	93(95)	55(52)	58(52)	71(68)	83(82)	62(66)	68(71)	86(94)	90(96)
$\mathscr{H}(I_3)$	77(79)	50(44)	51(45)	72(66)	83(83)	60(61)	65(68)	81(86)	87(87)
DWT(I_1)	82(74)	12(14)	24(20)	50(53)	76(79)	26(29)	37(38)	64(64)	87(89)

Table 3 (*continued*)

LFC(DWT(I_1))$_1$	85(98)	11(13)	22(22)	50(53)	75(75)	33(31)	41(43)	70(71)	87(91)
LFC(DWT(I_1))$_2$	98(99)	60(64)	60(64)	76(79)	91(89)	71(72)	77(77)	96(94)	96(95)
DWT(I_2)	92(93)	53(54)	53(57)	72(71)	85(81)	65(66)	67(69)	87(90)	92(92)
LFC(DWT(I_2))$_1$	91(94)	53(56)	53(56)	70(71)	80(80)	68(66)	72(70)	91(88)	91(90)
LFC(DWT(I_2))$_2$	86(80)	16(20)	28(29)	51(60)	80(79)	33(28)	40(34)	74(72)	86(88)
DWT(I_3)	82(85)	49(51)	53(52)	68(67)	78(81)	57(59)	63(65)	85(85)	87(88)
LFC(DWT(I_3))$_1$	80(86)	52(54)	52(54)	68(65)	80(77)	60(62)	67(68)	85(86)	88(90)
LFC(DWT(I_3))$_2$	80(78)	21(20)	30(32)	60(62)	81(83)	28(23)	38(31)	65(66)	84(84)
KSOFM(I_1)	75(73)	12(10)	19(18)	45(41)	77(69)	38(34)	40(37)	75(69)	88(86)
KSOFM(I_2)	78(76)	19(16)	23(21)	53(52)	82(78)	39(38)	45(42)	77(76)	88(87)
KSOFM(I_3)	65(61)	21(19)	26(25)	51(51)	78(73)	29(27)	34(33)	69(67)	81(80)
Test set III									
I_1	85(73)	18(21)	28(32)	49(55)	74(76)	35(40)	45(45)	61(56)	80(72)
I_2	78(80)	59(60)	59(65)	72(77)	83(84)	68(70)	73(75)	75(76)	76(80)
I_3	57(54)	60(59)	60(59)	69(69)	80(80)	64(68)	72(75)	73(78)	74(79)
$\mathscr{F}(I_1)$	77(74)	56(58)	57(58)	73(76)	87(83)	68(69)	78(77)	81(77)	81(77)
LFC($\mathscr{F}(I_1)$)	77(78)	52(59)	56(59)	73(73)	82(85)	69(68)	75(74)	83(83)	85(85)
|LFC($\mathscr{F}(I_1)$)|	68(68)	19(31)	25(36)	57(62)	81(82)	55(50)	60(56)	73(66)	78(74)
$\mathscr{F}(I_2)$	79(76)	50(53)	52(54)	59(67)	76(80)	54(61)	62(70)	71(77)	76(82)
LFC($\mathscr{F}(I_2)$)	84(81)	54(56)	57(58)	73(70)	83(79)	69(68)	72(73)	85(80)	87(86)
|LFC($\mathscr{F}(I_2)$)|	63(70)	21(35)	30(41)	60(67)	84(88)	34(42)	40(52)	68(65)	80(83)
$\mathscr{F}(I_3)$	74(76)	47(52)	48(52)	63(62)	76(75)	62(63)	69(72)	78(76)	81(79)
LFC($\mathscr{F}(I_3)$)	77(74)	52(53)	55(54)	68(66)	79(76)	61(62)	65(67)	71(73)	85(87)
|LFC($\mathscr{F}(I_3)$)|	65(70)	23(30)	31(38)	57(60)	79(82)	29(40)	36(48)	53(72)	72(79)
$\mathscr{F}^a(I_1)$	83(89)	61(55)	63(55)	77(72)	90(82)	67(67)	71(70)	71(80)	71(83)
$\mathscr{F}^a(I_2)$	81(79)	55(56)	56(57)	68(70)	79(79)	64(65)	70(72)	71(73)	73(77)
$\mathscr{F}^a(I_3)$	77(79)	52(53)	53(53)	65(74)	76(72)	62(62)	69(67)	73(75)	77(80)
$\mathscr{H}(I_1)$	89(87)	53(51)	54(52)	71(70)	79(80)	69(72)	76(76)	80(83)	80(83)
$\mathscr{H}(I_2)$	80(81)	56(52)	58(53)	72(66)	85(79)	65(66)	70(69)	74(74)	75(81)
$\mathscr{H}(I_3)$	75(72)	51(45)	53(45)	66(60)	78(78)	62(64)	69(70)	73(75)	73(76)
DWT(I_1)	78(75)	12(15)	23(19)	50(51)	75(80)	27(32)	35(45)	53(62)	79(80)
LFC(DWT(I_1))$_1$	69(84)	12(14)	18(27)	47(50)	78(70)	32(33)	41(40)	59(60)	80(77)
LFC(DWT(I_1))$_2$	85(83)	56(63)	58(63)	68(74)	82(85)	67(71)	69(75)	76(80)	76(80)
DWT(I_2)	82(80)	53(54)	54(55)	71(68)	85(83)	65(69)	69(73)	77(75)	79(73)
LFC(DWT(I_2))$_1$	80(84)	53(55)	56(56)	69(68)	79(79)	67(68)	71(72)	78(73)	78(73)
LFC(DWT(I_2))$_2$	76(74)	16(19)	22(28)	48(51)	75(74)	32(33)	39(39)	59(57)	73(73)
DWT(I_3)	73(75)	49(50)	49(50)	59(63)	75(75)	67(61)	72(67)	76(76)	79(78)
LFC(DWT(I_3))$_1$	74(80)	50(52)	50(52)	60(63)	73(75)	62(63)	68(69)	73(72)	76(74)
LFC(DWT(I_3))$_2$	73(72)	17(20)	26(30)	51(52)	73(75)	30(23)	40(32)	56(52)	72(68)
KSOFM(I_1)	73(72)	9(7)	13(12)	35(33)	60(56)	32(31)	34(32)	51(50)	65(65)
KSOFM(I_2)	75(74)	17(15)	21(21)	56(55)	85(81)	44(43)	47(46)	67(66)	76(76)
KSOFM(I_3)	66(64)	16(14)	19(20)	47(46)	73(71)	32(31)	36(35)	60(59)	83(82)

results for I_1. (The ordinary Fourier transform can be considered as a special case of the DFRT.)

For test set II (Table 3), the maximum correct target classification percentages of 100% (non-modular) and 99% (modular) are obtained when the input signals $\mathscr{F}^a(I_1)$ and LFC(DWT(I_1))$_2$ are used, respectively. These values are the same as those achieved with test set I. However, the percentages for correct range and azimuth estimates are generally 3–16% and 0–30% lower than test set I, respectively. Noting that the networks are trained only at 25 locations and at grid spacings of 5 cm and 10°, it can be concluded from the percentage of correct range and azimuth estimates obtained at error tolerances of $|\varepsilon_r| = 0.125$ and 1 cm and $|\varepsilon_\theta| = 0.25°$ and 2°, that the networks demonstrate the ability to interpolate between the training grid locations. Thus, the neural network maintains a certain spatial continuity between its input and output and does not haphazardly map positions which are not drawn from the 25 locations of Fig. 3. The correct target type percentages are just as good (99–100%) and the accuracy of the range/azimuth estimates would be acceptable for most applications. If better estimates are required, this can be achieved by reducing the training grid spacing in Fig. 3. Finally, we add that the results for the modular networks are slightly better than those for the non-modular

Table 4
Average percentages of correct classification, range (r) and azimuth (θ) estimation for KSOFM used prior to a linear classifier for the three test sets (I–II–III)

Input to ANN	% of correct classif.	% of correct r estimation				% of correct θ estimation			
		Error tolerance ε_r				Error tolerance ε_θ			
		± 0.125 cm	± 1 cm	± 5 cm	± 10 cm	$\pm 0.25°$	$\pm 2°$	$\pm 10°$	$\pm 20°$
KSOFM(I_1)	81-81-78	33-21-20	37-27-23	61-55-50	85-79-74	75-65-46	76-68-46	88-88-68	94-91-77
KSOFM(I_2)	85-85-77	41-26-28	44-30-30	71-59-58	90-84-80	80-65-47	82-68-48	93-88-63	97-88-76
KSOFM(I_3)	73-73-67	42-28-28	45-34-30	69-60-59	86-78-81	64-58-44	67-63-46	85-81-69	94-84-84

networks. Furthermore, use of modular networks has the additional advantage that one can independently optimize the pre-processing method and the parameters.

For test set III (Table 3), a maximum correct target classification percentage of 89% for both non-modular and modular network structures is obtained when the input signals $\mathscr{H}(I_1)$ (non-modular network structure) and $\mathscr{F}^a(I_1)$ (modular structure) are used, respectively. In most cases, $\mathscr{F}^a(I_1)$ gives the best range and azimuth estimates. Overall, we can conclude that the networks are fairly robust to variations in target shape, size, and roughness.

As an across-the-board conclusion, we may state that the fractional Fourier transform of I_1 with optimal order and low-frequency part of the wavelet transform of I_1 generally represent the best pre-processing options and offer substantial improvements with respect to no pre-processing.

The results obtained with KSOFM used prior to linear classifiers are given in Table 4. This combination results in better classification performance than when KSOFM is employed prior to ANNs (last three rows of Table 3). The classification and azimuth estimation performances are comparable to those obtained with the corresponding unprocessed signals (first three rows of Table 3). However, range estimation results are inferior compared to the results obtained with unprocessed signals. In any event, this approach is overshadowed by the best pre-processing methods in Table 3.

For networks trained with the GS algorithm, the resulting average percentages of correct type classification over all target types are given in Table 5. (Recall that this approach cannot produce localization results.) The maximum average percentage of correct classification is 97–98% for both test sets I and II, and can be obtained with either of the input signals $\mathscr{F}(I_1)$, LFC($\mathscr{F}(I_1)$), $|\text{LFC}(\mathscr{F}(I_1))|$, $\mathscr{F}^a(I_1)$, $\mathscr{H}(I_1)$, LFC$(\text{DWT}(I_1))_1$, or LFC$(\text{DWT}(I_1))_2$. It is 91–92% for test set III which can be obtained with either of the input signals $\mathscr{F}(I_1)$, $\mathscr{F}^a(I_1)$, or $\mathscr{H}(I_1)$. We see that the fractional Fourier and low-frequency wavelet transforms again give the best results, though several pre-processing alternatives also give comparable results in this case. Use of KSFOM results in exceptionally poor target differentiation. While the GS algorithm does not offer an advantage over the BP algorithm for test set I, it does offer better results for test set II; the classification results obtained with test set II are almost always as good as those with test set I with the GS algorithm, which means that it is accomplishing a very good task of spatial interpolation.

Table 5
The percentages of correct classification for ANNs trained with the GS algorithm for the three test sets (I–II–III)

Input to ANN	% of correct diff.
I_1	95-95-89
I_2	90-90-78
I_3	76-76-68
$\mathscr{F}(I_1)$	97-97-92
LFC($\mathscr{F}(I_1)$)	98-98-86
$\lvert\text{LFC}(\mathscr{F}(I_1))\rvert$	97-97-84
$\mathscr{F}(I_2)$	95-95-82
LFC($\mathscr{F}(I_2)$)	96-96-81
$\lvert\text{LFC}(\mathscr{F}(I_2))\rvert$	94-94-75
$\mathscr{F}(I_3)$	83-83-69
LFC($\mathscr{F}(I_3)$)	88-88-75
$\lvert\text{LFC}(\mathscr{F}(I_3))\rvert$	83-83-71
$\mathscr{F}^a(I_1)$	97-97-91
$\mathscr{F}^a(I_2)$	96-96-83
$\mathscr{F}^a(I_3)$	84-83-71
$\mathscr{H}(I_1)$	97-97-91
$\mathscr{H}(I_2)$	95-95-81
$\mathscr{H}(I_3)$	83-83-71
DWT(I_1)	95-95-89
LFC$(\text{DWT}(I_1))_1$	97-97-89
LFC$(\text{DWT}(I_1))_2$	97-97-88
DWT(I_2)	91-91-80
LFC$(\text{DWT}(I_2))_1$	90-90-79
LFC$(\text{DWT}(I_2))_2$	90-90-79
DWT(I_3)	75-75-67
LFC$(\text{DWT}(I_3))_1$	77-77-68
LFC$(\text{DWT}(I_3))_2$	80-80-71
KSOFM(I_1)	5-8-5
KSOFM(I_2)	13-11-9
KSOFM(I_3)	8-5-6

Table 6 summarizes the results for all of the methods considered, allowing their overall comparison. In this

Table 6
Overview of the methods compared (test set I). The numbers before/in the parentheses are for non-modular/modular ANNs trained with the BP algorithm, and the numbers in brackets are for ANNs trained with the GS algorithm

Method	No. of targets diff.	Differen. accuracy (%)	Correct pos. est. (%)				
			Range (0.125 cm/5 cm)	Azimuth (0.25°/10°)	Training data	Learning	Parametric
TDA	3	61	Yes 16/72	Yes 19/59	Not used	No	No
D-S evidential reasoning	3	89	Yes 17/72	Yes 32/98	Not used	No	No
Voting	3		Yes	Yes	Not used	No	No
SMV		82	16/72	19/61			
With pref. ordering and rel. measures							
$\mathrm{rel}_\alpha = 1$		88	16/72	19/61			
$\mathrm{rel}_\alpha^1 = m(r_\alpha)\ m(\theta_\alpha)$		90	29/82	32/98			
$\mathrm{rel}_\alpha^2 = \min\{m(r_\alpha)\ m(\theta_\alpha)\}$		90	29/82	32/98			
$\mathrm{rel}_\alpha^3 = \frac{m(r_\alpha) + m(\theta_\alpha)}{2}$		92	17/72	20/67			
$\mathrm{rel}_\alpha^4 = \max\{m(r_\alpha)\ m(\theta_\alpha)\}$		91	17/72	20/63			
$\mathrm{re1}_\alpha^5 = m(\text{1st choice}_\alpha) - m(\text{2nd choice}_\alpha)$		94	16/72	19/59			
Statistical pattern recognition	7		No	No			
KE		99			Used, stored	No	No
Ordinary k-NN		98			Used, stored	No	No
Generalised k-NN		99			Used, stored	No	No
PDE (homoscedastic NM)		76			Used, not stored	No	Yes
PDE (heteroscedastic NM)		81			Used, not stored	No	Yes
LDA		71			Used, not stored	No	No
FCC		98			Used, not stored	Yes	No
Neural networks	7		Yes (yes) [no]	Yes (yes) [no]	Used, not stored	Yes	No
Raw signal		95 (95) [95]	79/89 (73/93)	89/95 (95/97)			
DFT		99 (98) [98]	64/86 (72/88)	91/96 (94/98)			
DFRT		100 (98) [97]	75/89 (68/87)	93/97 (86/95)			
DHT		99 (97) [97]	67/85 (62/81)	84/95 (84/95)			
DWT		98 (99) [97]	71/87 (80/91)	90/97 (92/98)			
KSOFM		78 (76) [13]	24/59 (21/57)	69/86 (66/85)			
KSOFM with linear classifier	7	85	Yes 42/71	Yes 80/93	Used, not stored	Yes	No

summary table, the correct range and azimuth estimation percentages are given for error tolerances $\varepsilon_r = 0.125$ and 5 cm and $\varepsilon_\theta = 0.25^\circ$ and 10°, and the presented results correspond to the optimal variation and/or parameter values which give the best differentiation accuracy for each entry. For instance, the entry labeled "DWT" more specifically corresponds to the $\mathrm{LFC}(\mathrm{DWT}(I_1))_2$ result. TDA, D-S evidential reasoning, and the various voting strategies can differentiate only three target types while the other methods can differentiate all of the seven targets considered in this paper. This indicates that these latter methods, which involve training, are making more effective use of the available data. Statistical pattern recognition techniques and ANNs trained with the GS algorithm do not provide range and azimuth estimates.

A 100% correct differentiation is achieved with the non-modular ANN trained with the BP algorithm employing DFRT pre-processing. Better range and/or azimuth accuracy can be obtained with some of the other pre-processing methods at the cost of slightly poorer differentiation accuracy. In general, which method is best depends on the relative importance we attach to minimizing errors in differentiation, range, and azimuth. Nevertheless, a compromise which balances both differentiation and localization is obtained with DWT pre-processing using modular networks trained with the BP algorithm and offers 99% differentiation accuracy, 80% or 91% range estimation accuracy and 92% or 98% azimuth estimation accuracy, for $\varepsilon_r = 0.125$ and 5 cm and $\varepsilon_\theta = 0.25^\circ$ and 10°, respectively.

9. Conclusion

In this study, a comprehensive comparison of a large number of methods for target classification and localization

with sonar has been presented. Many variations of the methods have been considered and their optimal configurations and parameters determined. Different pre-processing methods, structures, and training algorithms for ANNs have been implemented, among which the method leading to the best results emerged.

The performance of all the methods considered have been compared for three different test sets. The first test set is based on targets situated at the training locations. The second is based on targets situated at arbitrary locations; it has been observed that ANNs are able to achieve considerable spatial interpolation. The third is based on targets which are not used for training and are somewhat different in size, shape, or roughness than those used for training; it has been observed that the methods are fairly robust in identifying these modified targets.

The results are summarized in Table 6 which provides a comparison of the various methods optimized within themselves. In terms of the number of targets that can be differentiated, correct differentiation percentage, and correct range and azimuth estimation, the use of modular ANNs trained with the BP algorithm, usually with DFRT or DWT pre-processing, gives the best results. With the best optimized methods, it is possible to obtain near perfect differentiation, around 85% correct range estimation and around 95% correct azimuth estimation, which would be satisfactory in a wide range of applications. While the GS algorithm does not offer an advantage over the BP algorithm for test set I, it does offer better results for test set II; the classification results obtained with test set II are almost always as good as that with test set I, which means that the GS algorithm is accomplishing a very good task of spatial interpolation.

Use of preference orders and assignment of reliability measures always brings some improvement compared to the results of SMV. The fifth reliability measure gives the highest percentage of correct differentiation, and is followed by the third, fourth, first, and second measures. These five reliability measures always result in better classification performance than a uniform reliability measure assignment. In addition, their performances are also better than that of D-S evidential reasoning which is in turn better than the TDA. For statistical pattern recognition techniques, the target classification performances of most non-parametric approaches are better than that of PDE; the best results are obtained with generalized k-NN, generally followed by KE and FCC or k-NN.

Given the attractive performance for cost of sonar-based systems, we believe that the results of this study will be of great usefulness for engineers designing or implementing sonar systems and researchers investigating algorithms and performance evaluation of such systems. While we have concentrated on sonar sensing, the techniques evaluated and compared in this paper may be useful for other sensing modalities and environments where information from a multiplicity of partial viewpoints must be combined.

References

[1] A. Elfes, Sonar based real-world mapping and navigation, IEEE Trans. Robotics Autom. RA-3 (3) (1987) 249–265.

[2] R. Kuc, B.V. Viard, A physically-based navigation strategy for sonar-guided vehicles, Int. J. Robotics Res. 10 (2) (1991) 75–87.

[3] R. Kuc, M.W. Siegel, Physically-based simulation model for acoustic sensor robot navigation, IEEE Trans. Pattern Anal. Mach. Intell. PAMI-9 (6) (1987) 766–778.

[4] A. Kurz, Constructing maps for mobile robot navigation based on ultrasonic range data, IEEE Trans. Systems Man Cybernet. B 26 (2) (1996) 233–242.

[5] J.J. Leonard, H.F. Durrant-Whyte, Directed Sonar Navigation, Kluwer Academic Press, London, 1992.

[6] Ö. Bozma, R. Kuc, A physical model-based analysis of heterogeneous environments using sonar—ENDURA method, IEEE Trans. Pattern Anal. Mach. Intell. 16 (5) (1994) 497–506.

[7] R. Kuc, Three-dimensional tracking using qualitative bionic sonar, Robotics Autonomous Systems 11 (2) (1993) 213–219.

[8] J. Borenstein, Y. Koren, Obstacle avoidance with ultrasonic sensors, IEEE Trans. Robotics Autom. RA-4 (2) (1988) 213–218.

[9] H. Peremans, K. Audenaert, J.M. Van Campenhout, A high-resolution sensor based on tri-aural perception, IEEE Trans. Robotics Autom. 9 (1) (1993) 36–48.

[10] L. Kleeman, R. Kuc, Mobile robot sonar for target localization and classification, Int. J. Robotics Res. 14 (4) (1995) 295–318.

[11] B. Barshan, B. Ayrulu, Performance comparison of four methods of time-of-flight estimation for sonar waveforms, Electron. Lett. 34 (16) (1998) 1616–1617.

[12] B. Ayrulu, B. Barshan, Neural networks for improved target differentiation and localization with sonar, Neural Networks 14 (3) (2001) 355–373.

[13] B. Ayrulu, B. Barshan, Reliability measure assignment to sonar for robust target differentiation, Pattern Recognition 35 (6) (2002) 1403–1419.

[14] B. Ayrulu, B. Barshan, Identification of target primitives with multiple decision-making sonars using evidential reasoning, Int. J. Robotics Res. 17 (6) (1998) 598–623.

[15] B. Barshan, B. Ayrulu, S.W. Utete, Neural network based target differentiation using sonar for robotics applications, IEEE Trans. Robotics Autom. 16 (4) (2000) 435–442.

[16] B. Barshan, Fast processing techniques for accurate ultrasonic range measurements, Meas. Sci. Technol. 11 (1) (2000) 45–50.

[17] B. Barshan, R. Kuc, Differentiating sonar reflections from corners and planes by employing an intelligent sensor, IEEE Trans. Pattern Anal. Mach. Intell. 12 (6) (1990) 560–569.

[18] Panasonic Corp., Ultrasonic ceramic microphones, 12 Blanchard Rd., Burlington, MA, 1989.

[19] B. Barshan, A.Ş. Sekmen, Radius of curvature estimation and localization of targets using multiple sonar sensors, J. Acoust. Soc. Am. 105 (4) (1999) 2318–2331.

[20] G. Shafer, A Mathematical Theory of Evidence, Princeton University Press, Princeton, NJ, 1976.

[21] K.J. Arrow, Social Choice and Individual Values, Wiley, New York, 1951.

[22] B. Barshan, A sonar-based mobile robot for bat-like prey capture. Ph.D. Thesis, Yale University, Department of Electrical Engineering, New Haven, CT, December 1991.
[23] M. Rosenblatt, Remarks on some non-parametric estimates of a density function, Ann. Math. Stat. 27 (3) (1956) 832–837.
[24] E. Fix, J.L. Hodges, Discriminatory analysis, nonparametric discrimination, consistency properties, Technical Report 21-49-004 4, United States Air Force, School of Aviation Medicine, Randolph Field, TX, 1951.
[25] M.P. Wand, M.C. Jones, Kernel Smoothing, Chapman & Hall, London, 1995.
[26] S.J. Sheather, The performance of six popular bandwidth selection methods on some real data sets, Comput. Stat. 7 (1992) 225–250.
[27] D.J. Hand, Discrimination and Classification, Wiley, New York, 1986.
[28] B.W. Silverman, Density Estimation for Statistics and Data Analysis, Chapman & Hall, New York, 1986.
[29] V.K. Rohatgi, An Introduction to Probability Theory and Mathematical Statistics, Wiley, New York, 1976.
[30] G.J. McLachlan, Discriminant Analysis and Statistical Pattern Recognition, Wiley, New York, 1992.
[31] J.C. Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters, J. Cybernet. 3 (1974) 32–57.
[32] J.C. Bezdek, Fuzzy mathematics in pattern classification, Ph.D. Thesis, Cornell University, Department of Applied Mathematics, Ithaca, NY, 1973.
[33] B.C. Bai, N.H. Farhat, Learning networks for extrapolation and radar target identification, Neural Networks 5 (3) (1992) 507–529.
[34] M. Cohen, H. Franco, N. Morgan, D. Rumelhart, V. Abrash, Context-dependent multiple distribution phonetic modelling with MLPs, in: S.J. Hanson, J.D. Cowan, C.L. Giles (Eds.), Advances in Neural Information Processing Systems, Morgan Kaufmann, San Mateo, CA, 1993, pp. 649–657.
[35] K.S. Narendra, K. Parthasarathy, Gradient methods for the optimization of dynamic systems containing neural networks, IEEE Trans. Neural Networks 2 (2) (1991) 252–262.
[36] M.I. Jordan, R.A. Jacobs, Learning to control an unstable system with forward modeling, in: D.S. Touretzky (Ed.), Advances in Neural Information Processing Systems, Vol. 2, Morgan Kaufmann, San Mateo, CA, 1990, pp. 324–331.
[37] M. Galicki, H. Witte, J. Dörschel, M. Eiselt, G. Griessbach, Common optimization of adaptive processing units and a neural network during the learning period: application in EEG pattern recognition, Neural Networks 10 (6) (1997) 1153–1163.
[38] Y. LeCun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel, Handwritten digit recognition with a back-propagation network, in: D.S. Touretzky (Ed.), Advances in Neural Information Processing Systems, Vol. 2, Morgan Kaufmann, San Mateo, CA, 1990, pp. 396–404.
[39] E. Alpaydın, Multiple networks for function learning, in: Proceedings of IEEE International Conference on Neural Networks, San Francisco, March 1993, pp. 9–14.
[40] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning representations by back-propagation errors, Nature 323 (1986) 533–536.
[41] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice-Hall, New Jersey, 1994.
[42] Y.Q. Chen, D.W. Thomas, M.S. Nixon, Generating-shrinking algorithm for learning arbitrary classification, Neural Networks 7 (9) (1994) 1477–1489.
[43] H.M. Ozaktas, Z. Zalevsky, M.A. Kutay, The Fractional Fourier Transform with Applications in Optics and Signal Processing, Wiley, New York, 2001.
[44] D. Mendlovic, H.M. Ozaktas, Fourier transforms and their optical implementation: I, J. Opt. Soc. Am. A 10 (9) (1993) 1875–1881.
[45] H.M. Ozaktas, D. Mendlovic, Fourier transforms of fractional order and their optical interpretation, Opt. Commun. 101 (3,4) (1993) 163–169.
[46] H.M. Ozaktas, D. Mendlovic, Fractional Fourier transforms and their optical implementation: II, J. Opt. Soc. Am. A 10 (12) (1993) 2522–2531.
[47] H.M. Ozaktas, D. Mendlovic, Fractional Fourier optics, J. Opt. Soc. Am. A 12 (4) (1995) 743–751.
[48] H.M. Ozaktas, B. Barshan, D. Mendlovic, Convolution and filtering in fractional Fourier domains, Opt. Rev. 1 (1) (1994) 15–16.
[49] H.M. Ozaktas, B. Barshan, D. Mendlovic, L. Onural, Convolution, filtering, and multiplexing in fractional Fourier domains and their relation to chirp and wavelet transforms, J. Opt. Soc. Am. A 11 (2) (1994) 547–559.
[50] B. Barshan, M.A. Kutay, H.M. Ozaktas, Optimal filtering with linear canonical transformations, Opt. Commun. 135 (1–3) (1997) 32–36.
[51] I.Ş. Yetik, H.M. Ozaktas, B. Barshan, L. Onural, Perspective projections in the space-frequency plane and fractional Fourier transforms, J. Opt. Soc. Am. A 17 (12) (2000) 2382–2390.
[52] C. Candan, M.A. Kutay, H.M. Ozaktas, The discrete fractional Fourier transform, IEEE Trans. Signal Process. 48 (5) (2000) 1329–1337.
[53] R.N. Bracewell, The Hartley Transform, Oxford University Press, New York, 1986.
[54] R.C. Gonzalez, P. Wintz, Digital Image Processing, Addison-Wesley, Reading, MA, 1987.
[55] D.F. Marshall, W.K. Jenkins, J.J. Murphy, The use of orthogonal transforms for improving performance of adaptive filters, IEEE Trans. Circuits Systems 36 (1989) 474–484.
[56] C.K. Chui, An Introduction to Wavelets, Academic Press, San Diego, CA, 1992.
[57] S.G. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, IEEE Trans. Pattern Anal. Mach. Intell. 11 (7) (1989) 674–693.
[58] T. Kohonen, Self-organized formation of topologically correct feature maps, Biol. Cybernet. 43 (1) (1982) 59–69.
[59] B. Barshan, B. Ayrulu, Fractional Fourier transform pre-processing for neural networks and its application to object recognition, Neural Networks 15 (1) (2002) 131–140.

About the Author— BILLUR BARSHAN received B.S. degrees in both Electrical Engineering and in Physics from Boğaziçi University, Istanbul, Turkey and the M.S. and Ph.D. degrees in electrical engineering from Yale University, New Haven, Connecticut, in 1986, 1988, and 1991, respectively. Dr. Barshan was a research assistant at Yale University from 1987 to 1991, and a postdoctoral researcher at the Robotics Research Group at University of Oxford, UK from 1991 to 1993. She joined Bilkent University, Ankara in 1993 where she is currently associate professor at the Department of Electrical Engineering. Dr. Barshan is the founder of the Robotics and Sensing Laboratory in the same department. She is the recipient of the 1994 Nakamura Prize awarded to the most outstanding paper in 1993 IEEE/RSJ Intelligent Robots and Systems International Conference, 1998 TÜBİTAK Young Investigator Award, and 1999 Mustafa N. Parlar Foundation Research Award. Dr. Barshan's current research interests include intelligent sensors, sonar and inertial navigation systems, sensor-based robotics, and multi-sensor data fusion.

About the Author—BIRSEL AYRULU received the B.S. degree in Electrical Engineering from Middle East Technical University and the M.S. and Ph.D. degrees in Electrical Engineering from Bilkent University, Ankara, Turkey in 1994, 1996, and 2001, respectively. Her current research interests include intelligent sensing, sonar sensing, sensor data fusion, learning methods, target differentiation, and sensor-based robotics.