Sonar Target Classification with Sonar Fingerprint

Jian Yuan, Guo-Hui Li, Wei Zhu, and Xu Lai

1 Department of System Engineering, School of Information System & Management, National University of Defense Technology, Yanwachi St.47#, 410073, Changsha, Hunan, China
yuanjian@vip.163.com, guohli@nudt.edu.cn, zhuwei_jim@163.com, lailai_boy@yahoo.com.cn

Abstract. To recognize underwater target precisely is always a hot potato to navies due to the complicated watery environment. It’s different from the aerial circumstance. There is much more interfere under the sea. Sonar is the most efficient way to detect underwater world at present time. In this paper, a genetic-based classifier system is designed which recognizes targets by sonar fingerprints. This method will release the sonarman to a certain degree. Experiments show that the system gains acceptable speed and accuracy in the classifying operation. The proposed underwater target classifier system is highly automatic, with quite finite hardware requirements in operating.

1 Introduction

Surface ships are the core power of navy; their severest danger is come from both the air and the submarine. To detect the aerial target depends on the radar system, and it works based on the electromagnetic wave. Electromagnetic wave spread in the air, there is less interferes than the underwater world; consequently, abundant methods in the area of radar target recognition had been put forward. For example, the Monterey University in the California, US, only in the year 1983 to 1990, there were more than 40 papers about recognizing the targets with their pole were embodied into the AD Report [9], [11]. But the underwater world needs different methods despite the two are looked similarly apparently. Radar operates over high frequency electromagnetic wave, while sonar operates over sound wave. It’s in a much lower band about 5kHz—30kHz, and the newest information even shows that to detect the most advanced sub should use infrasound wave [8], [13]. The Signal-to-Noise of radar’s echo is normally very high, and the dynamic range is small, however, the dynamic range of sonar is large, often between several micro volts to tens of volts, and SNR can range from minus decibel to tens of decibel [8].

Lacking of advanced method to recognizing the aqueous target leads to ridiculous phenomenon, presently the recognition still relies on the experienced sonarman’s monitoring and listening. How can we trust the reliability? The weapon’s manufacturing technique has gone through its evolution; no country can collect all samples in the water. Under this finite condition, the sonar system must obtain the ability to classify.
Sonar fingerprint technique maps sonar echo into fingerprint segment, which can identify the original echo uniquely. It’s only composed with very limited data. It highly decreases the difficulty of signal processing and the time consume. Furthermore, the fingerprint has a thick skin to the store way or the format, i.e. no matter how the sound clips are stored, perceptual similarity always leads to approximate fingerprint [6].

This paper describes a valid underwater target recognition system’s architecture and working procedure. After focusing on the system’s working precondition, sonar fingerprint’s extracting guiding principle and the algorithm in Section 2 and the whole system’s architecture in Section 3, we elaborating on the proposed special improvements to the system in Section 4 and the system’s working flow in Section 5. Experiments, discussion and the future work are described in the last two sections.

2 Fingerprint Extraction Algorithm

2.1 Target Recognizing Procedure in General

Target’s echo is a time-series; it’s the function of the target’s class, distance and orientation. It contains abundant information of target’s feature. But the sound wave can hardly be processed directly, the data is too much.

Therefore the course of feature extraction and compression must be put up.

![Fig. 1. Target Recognizing Procedure](image)

Figure 1 is the general procedure of target recognition [9], [13]. The pretreatment course accepts the target’s echo; it extracts the feature, and then sends it to the next course to process or transform, if necessary.

The feature compression course compresses the extracted high-dimensional feature vector into low-dimensional feature vector. In this paper, this course is referred to the detector (a.k.a. fingerprint extractor). And its task can be described resumptively like this: extracts sonar fingerprints and outputs bit strings so that the classifier system can do the training or the classification.
2.2 Extraction Algorithm

Sonar fingerprint intend to capture the relevant perceptual features of echo. At the same time extracting fingerprints should be fast and easy, preferably with a small granularity to allow usage in some extreme applications (the warfare circumstance).

First the echo signal is segmented into overlapping frames. For every frame a set of features is computed. Preferably the features are chosen such that they are invariant (at least to a certain degree) to signal degradations. Features that have been proposed are well known audio features such as Fourier coefficients [4], Mel Frequency Cepstral Coefficients (MFCC) [7], spectral flatness, sharpness, Linear Predictive Coding (LPC) coefficients and others [1]. Also derived quantities such as derivatives, means and variances of audio features are used. Generally the extracted features are mapped into a more compact representation by using classification algorithms, such as quantization [5]. The compact representation of a single frame will be referred to as a sub-fingerprint. The global fingerprint procedure converts a stream of echo into a stream of sub-fingerprints. One sub-fingerprint usually does not contain sufficient data to identify an echo clip. The basic unit that contains sufficient data to identify an echo clip (and therefore determining the granularity) will be referred to as a fingerprint-block.

The overlapping frames have a length of 0.37 seconds and are weighted by a Hamming window with an overlap factor of 31/32. The extraction scheme extracts 32-bit sub-fingerprints for every interval of 11.6 milliseconds. A fingerprint block consists of 256 subsequent sub-fingerprints, corresponding to a granularity of only 3 seconds. In the worst-case scenario the frame boundaries used during identification are 5.8 milliseconds. The large overlap assures that even in this worst-case scenario the sub-fingerprints of the echo clip to be identified are still very similar to the sub-
fingerprints of the same clip in the database. Due to the large overlap subsequent sub-fingerprints have a large similarity and are slowly varying in time.

The most important perceptual audio features live in the frequency domain. Therefore a spectral representation is computed by performing a Fourier transform on every frame.

In order to extract a 32-bit sub-fingerprint value for every frame, 33 non-overlapping frequency bands are selected. These bands lie in the range from 300Hz to 2000Hz (most echo sound’s energy lives in this range and the most relevant spectral range for the HAS) but not the wholly band and have a logarithmic spacing. The logarithmic spacing is chosen, because the HAS operates on approximately logarithmic bands. Experimentally it was verified that the sign of energy differences (simultaneously along the time and frequency axes) is a property that is very robust to many kinds of processing.

If we denote the energy of band m of frame n by \(e(n,m)\) and the m-th bit of the sub-fingerprint of frame n by \(f(n, m)\), the bits of the sub-fingerprint are formally defined as:

\[
f(n, m) = \begin{cases} 
1 & \text{if } e(n, m) - e(n, m+1) - (e(n-1, m) - e(n-1, m+1)) > 0 \\
0 & \text{if } e(n, m) - e(n, m+1) - (e(n-1, m) - e(n-1, m+1)) \leq 0
\end{cases}
\]  

(1)

![Fingerprint block](image)

**Fig. 3.** (a) Fingerprint block of original echo clip, (b) fingerprint block of a compressed version, (c) the difference between a and b showing the bit errors in black (BER=0.034).

Figure 3 shows an example of 256 subsequent 32-bit sub-fingerprints (i.e. a fingerprint block), extracted with the above scheme from an echo of a submarine. A ‘1’ bit corresponds to a white pixel and a ‘0’ bit to a black pixel. Figure 3a and Figure 3b show a fingerprint block from an original echo sound and its compressed version of the same excerpt, respectively. The usage of compressed version is to symbolize the degradation of the echo caused by the watery environment. Ideally these two figures
should be identical, but due to the compression (corresponding to the attenuation in the water) some of the bits are retrieved incorrectly. These bit errors, which are used as the similarity measure for our fingerprint scheme, are shown in black in Figure 3c.

The computing resources needed for the proposed algorithm are limited. Since the algorithm only takes into account frequencies below 2 kHz the echo is first down sampled to a mono stream with a sampling rate of 5 kHz. The sub-fingerprints are designed innately robust against signal degradations, therefore very simple down sample filters can be used. The most computationally demanding operation is the Fourier transform of every frame. In the down sampled signal a frame has a length of 2048 samples.

3 The Classifier System Architecture

In order to get the correct class of target, a well-designed classifier is necessary. The classifier receives the fingerprint information and tells which class the current target is through comparing with the well-trained fingerprint-rule set.

In the underwater target recognition application, a well-defined system has the modules below in figure 4:

![Classifier System Architecture](image)

Fig. 4. Classifier System Architecture

Firstly, the detector, that is, the sonar fingerprint extractor distils the sonar fingerprint according to the CS’s requirements, which means the echo sound is turned into bit string in fixed length, once they are sent to the CS, they become encoded environmental information. Then the information becomes messages, a piece of message is delivered to the message queue, in the queue, maybe the message would trigger the rule (we call it classifier). A rule’s form is showed here [2]:

\[
\text{IF} \ <\text{condition}> \ \text{THEN} \ <\text{action}> \quad (3)
\]

It means when the condition is met, the rule will act. The triggered classifier will send message to the message queue, and the message will probably trigger another classifier or bring an action (where there is no more classifier to be triggered). Finally the action will act on the environment. That is:

\[
<\text{environment information}> \rightarrow <\text{classifier}1>:<\text{message}1> \rightarrow \ldots \rightarrow <\text{classifier} \ n>:<\text{message} \ n> \rightarrow <\text{action}> \quad (4)
\]

Now five main components included in this system will be elaborated:
3.1 Detector (Fingerprint Extractor)

It is the interface of the system which directly deals with the echoes. It extracts the sonar fingerprint and encodes it in shape like this:

\[ M^i = [x^i, y^i] \]  \hspace{1cm} (5)

Every piece of message is a dualistic group. In this formula, ‘i’ is the sequence number of the message, ‘x’ is the condition part, it contains the codes which embodies all the classifying information, \( x^i \in \{0,1\}^m \), ‘y’ is the conclusion part, \( y^i \in \{0,1\}^n \), that is the class of the object, this part is added manually for training after encoding. For example, \([(10001011), (1011)]\) is a piece of message which composed by 8-bit condition and 4-bit conclusion.

3.2 Message List

It’s the system’s repository, it contains all the messages. It offers the raw materials that are needed in the following operation.

3.3 Classifier System

Rules are different in form with the environment messages. They should have enhanced adaptability, so they may cover more information. Consequently the condition part of the rule contains wildcard ‘#’. It means that a bit can be either ‘0’ or ‘1’, it cannot be decided. Enhanced adaptability will lead to increased rule set. In the followed procedure a method is designed to avoid the redundancy caused by this strategy.

The classifier is generated by GA subsystem. There are two kinds of classifier in the system, one is Working Classifier, when the learning begins, it chooses scheduled number of messages and mutates them in a certain degree. The following learning and evolution operating carry and achieve in it; the other is Refinement Classifier, all the useful rules in the Working Classifier and collisional rules obtained from message list form it. Some material handling take place in it, this can merge the redundant rules.

Relative to traditional rule, the newly proposed rule is a triplet; its form is as follows:

\[ C^i = [U^i, V^i, fitness^i] \]  \hspace{1cm} (6)

\( U^i \) is the condition part, \( U^i \in \{0,1,\#\}^m \), \# is the wildcard, \( V^i \) is the conclusion part, \( V^i \in \{0,1\}^n \), \( fitness^i \) is a rule’s fitness, it’s two-tuples itself, it likes in form:

\[ fitness^i = [fit1, fit2] \]  \hspace{1cm} (7)

\( fit1 \) and \( fit2 \) are positive integers, they separately means the quantity that a piece of message matches the rule’s conclusion or not within the rule’s coverage area. For
example, \([(1##0#0#1), (1011), (20, 4)]\) means the rule’s condition part \((1##0#0#1)\) covered 24 samples, among these, there are 20 samples hold the same conclusion with the rule, and 4 at variance.

3.4 Testing List

All the testing examples make up of it. A testing example \(T^i\) likes a piece of message, and one difference exists in its conclusion part, \(y^i \in \{*\}^n\), * means it is not confirmed yet. After testing by the CS, it will obtain the form that same with the message and the example itself will turn to a new message. Its conclusion can both act on the environment and feedback the new message through the environment. So that the system may continue its learning and can adapt the environment better.

3.5 Actor

It turns the classifying result to the truly output value, then act on the environment. It’s the reverse process of Detector, in fact, it’s a decode course. In our application, we can learn which class of an underwater target is after the whole training and classifying procedure from the actor.

4 Improvements on Classifier System

Considering the application the system will meet, speediness and accuracy is badly needed. Here, the Comparing and Matching Algorithm (CMA), Hyperplasia Operator (HO), Refining Classifier (RC) and alterable mutation probability \((P_m)\) is designed to meet this requirement.

4.1 Comparing and Matching Algorithm

Comparing with the commonly used Bucket Brigade Algorithm (BBA), CMA is more convenient for users to get the fitness, it endues the fitness value with more explicit statistical meaning, and this allows users to explain the rule with background knowledge. Furthermore, the fitness which is educed by the CMA can decide if the rule should be merged in the RC.

The designing idea of the CMA is as followed: Match the rules and the messages in the \(ML\) one by one, it modifies the rule’s fitness bases the success or failure of the match. The fitness has been described in formula 8. The ultimately purpose is to ensure the survival of the better rule and the elimination of the worse rules.

Here is the approach:

1. Initialize the rule’s fitness, i.e. \(fit1 \rightarrow 0\), \(fit2 \rightarrow 0\);
2. Select a piece of message from \(ML\), match the rules in the Working Classifier one by one.

If both the condition and the conclusion are matched,
ATEN $fit1 \leftarrow fit1 + 1$;
IF the condition matches while the conclusion not,
THEN $fit2 \leftarrow fit2 + 1$;
IF the condition doesn’t match,
THEN $fitness \leftarrow fitness$
Return to Step 2, do the match until the $ML$ is out.

4.2 Hyperplasia Operator

The Hyperplasia Operator is designed for the avoidance of the situation that no rule could match current message. Therefore it gives the system persistent learning ability.

Suppose this situation, when the message matches the rule, it is found that there is no rule can distinguish the message, what can we do? Here the $HO$ will generate a new rule which can match the message.

The approach is, mutate the message’s condition part in every bit with a fixed probability. While mutating, the 1 or 0 on this bit will be changed into “#”. Make the result the condition part of a new rule, and then give a specific class artificially.

4.3 Refining Classifier

The purpose of $RC$ (also called Merge Operator) is to reduce the quantity of the redundant rules. The smaller scale of the rule set, the rapider the classification achieves.
1. Toward every rule in the original population, if its corresponding $fit1 \neq 0$ and $fit2=0$, the $RC$ will reserve it, otherwise the $RC$ will eliminate it.
2. Match the reserved rules each other. Suppose $R_1$ and $R_2$ are two rules that survived.
   IF $R_1 \supseteq R_2$ and $fit1(R_1) = fit1(R_2)$,
   THEN keep $R_2$, eliminate $R_1$;
   IF $R_1 \supseteq R_2$ and $fit1(R_1) > fit1(R_2)$,
   THEN keep $R_1$, eliminate $R_2$.

4.4 Alterable Mutation Probability

In normal $GA$, the value of the parameter $P_m$ is invariable. The classifier mutates according to the fixed probability from beginning to end. Experiments show that set an alterable mutation probability correctly may increase the speed and the accuracy of classification.

In the prophase of the evolution, all rule’s fitness are relatively low, high mutation probability would probably destroy some potential rules. Lower mutation probability should be used in order to keep the stability of the evolution, and this can speed up the evolution either. In the anaphase the fitness has already become steady; amplified mutation probability will enlarge the variety.

Further more, the training and the testing go hand in hand in the system, know from normal classifier system. In general, the training samples cannot cover all the
situations in reality, and the proportion of positive and negative samples is difficult in equaling with the fact. However, the policy of simultaneous processing makes the system catch the persistent learning ability. So, the samples decide the obtaining procedure inappreciably, and consequently the rules are in phase with the reality better.

5 Working Procedure of Whole System

The whole system’s working flow can be sum up as follows:
1) Initialize all preset parameters (for example: rule number $n$, crossover probability $P_c$, mutation probability $P_m$, etc). Initialize Working Classifier, set up initial population $F$, generate $n$ rules randomly, and evaluate equal fitness to each rule.
2) Encodes the training set into binary messages and put into Message List [$M$].
3) Call CMA, modify every fitness of initial population $F$. If there is no rule can match the messages, call the HO, generate an assorted rule, and then put this rule into population $F$.
4) Call Merge Operator, the population after merging become population $M$.
5) If $M$ becomes convergent, copy all the rules into $RC$, then return to step 8).
6) Call GA, creates a new generation population $L$, merges $L$ and $M$, then sends them to population $F$, and renews the initial population $F$.
7) Return to step 3).
8) Call rules in $RC$, examines the Testing List [$T$], and makes its conclusion part.
9) Send [$T$] to actor, it is the output value, and can decide the class of the target.

In addition, the detector executes fingerprint extracting procedure in parallel course.

6 Experiment and Discussion

The class is specified compulsively when an echo is being a training example. If the system is well-trained, the recognizing time won’t be long. In practice, the main consume is the processing to the echoes. If the Fourier transform is implemented as a fixed point real-valued FFT, then the fingerprinting algorithm would run more efficiently.

In order to validate the performance, we take 8 kinds of different targets into identification; they were submarines or surface ships, woodiness ships, buoyages, fish torpedoes, trail streams, submerged rocks or rockiness benthal, shoals and benthal reverberations or fake echoes. Each kind of target contained 50 echoes, we choose 40 randomly as training examples, the rest were test examples. The result of the classification is shown in table 1.

Table 1. Result of the classification

<table>
<thead>
<tr>
<th>Training set</th>
<th>Nr</th>
<th>TR</th>
<th>Testing set</th>
<th>C</th>
<th>Er</th>
<th>Er%</th>
<th>TE</th>
</tr>
</thead>
<tbody>
<tr>
<td>320(40×8)</td>
<td>86</td>
<td>100%</td>
<td>80(10×8)</td>
<td>74</td>
<td>6</td>
<td>7.5%</td>
<td>92.5%</td>
</tr>
</tbody>
</table>

©GESTS-Oct.2005
Sonar Target Classification with Sonar

Nr is the rule quantity after training, C is the amount of the correctly recognized testing targets, TR is the recognizing right rate of training set, TE is the recognizing right rate of testing set, Er is the wrongly recognized target; Er% is the error rate. In our opinion, the classification is successful only until TR > 95% and TE > 70%.

The difference between the improvements supplied and absent is shown in table 2. Without the hyperplasia operation, a large number of training examples covered all kinds of classes are need to be collected. Now the hyperplasia operation can generate them gradually; thereby the system possesses the persistent learning ability.

Table 2. Improvements can affect the system much

<table>
<thead>
<tr>
<th></th>
<th>With RC</th>
<th>Without RC</th>
<th>Alterable $P_m$</th>
<th>Invariable $P_m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Rules</td>
<td>86</td>
<td>231</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average Fitness</td>
<td></td>
<td></td>
<td>34.3</td>
<td>28.7</td>
</tr>
<tr>
<td>Training Time(Minute)</td>
<td>6</td>
<td>65</td>
<td>6</td>
<td>9</td>
</tr>
</tbody>
</table>

7 Conclusion and Future Work

Despite there are some achievements, there are still some problems for future work:
- Parallel identification. Sonar’s feedback has particular meaning, it’s the decision support info, and the synthetical information will help to increase the accuracy.
- Concise sonar fingerprint extracting algorithm. In order to reduce unnecessary process and extracting complexity, farther study to the echo should be carried out.
- Extend the system’s function. It’s a promising idea that stores the well-trained rules and the correlative equipment information into database. Associated information can be shown after a class is recognized. The commander could make more scientific decision on how to deal with the target under the system’s support.

References


Biography

▲ Name: Jian Yuan
Address: Yanwachi St. 47#, Changsha, Hunan, P.R.China.
Education & Work experience: Jian Yuan received the B.S. and M.S. degrees in strategies from Guangzhou Naval Academy, Guangzhou, Guangdong, P.R.China, in 2000 and 2003, respectively. He is currently pursuing the Ph.D. degree in the school of Information System & Management. His research interest includes multimedia audio processing, multimedia data mining and machine learning.
Tel: +86-13574849643
E-mail: yuanjian@vip.163.com

▲ Name: Guo-Hui Li
Address: Yanwachi St.47#, Changsha, Hunan, P.R.China
Education & Work experience: Guohui Li received his B.S.E.E. degree from Changsha Institute of Technology (CIT), Hunan, China, in 1983, and the M.S. degree in communication and electrical engineering from the CIT in 1986. He received PhD degree in 2001. He is the member of China Computer Association, multimedia division. He has joined the Multimedia Laboratory in Department of System Engi-
Sonar Target Classification with Sonar Engineering and Mathematics, CIT since 1986. He has become full professor in CIT since 1999. Prof. Li has received the top award of Science and technology advancement from NSEM in 1994 and 1997 in China for his major contribution to the two projects: hypermedia system and distributed hypermedia database system, and get several second and third-grade awards on distributed multimedia system, content-based information retrieval, image and document management system. He has authored or coauthored three books, more than 50 journal and conference papers and received one of the best paper awards from the National Conference on Multimedia Technology in 1995. His research interests include (1) Multimedia information management, which include multimedia database system, content-based retrieval, digital library, hypermedia, multimedia security; (2) Multimedia communication and network, which includes content delivery, video conferencing, video transmission and coding, multimedia wireless communications and networking. He is currently the principle investigator of a multimedia database system funded by NSEM.

Tel: +86-731-4576437
E-mail: guohli@nudt.edu.cn

▲ Name: Wei Zhu
Address: Yanwachi St.47#, Changsha, Hunan, P.R.China
Education & Work experience: Wei Zhu received the B.S. and M.S. degrees in system engineering from National University of Defence Technology, Changsha, Hunan, P.R.China, in 2001 and 2003, respectively. He is currently pursuing the Ph.D. degree in the school of Information System & Management. His research interest includes intelligent media processing.
Tel: +86-13787241204
E-mail: zhuwei_jim@163.com

▲ Name: Xu Lai
Address: Yanwachi St.47#, Changsha, Hunan, P.R.China
Education & Work experience: Xu Lai received the B.S. and M.S. degrees in system engineering from National University of Defence Technology, Changsha, Hunan, P.R.China, in 2001 and 2003, respectively. He is currently pursuing the Ph.D. degree in the school of Information System & Management. His research interest includes multimedia data mining.
Tel: +86-13467539563
E-mail: lailai_boy@yahoo.com.cn
Other information:

©GESTS-Oct.2005