Proceedings of The Institute of Acoustics

Sea bottom quality determination using an inexpensive echo-sounder and microprocessor.

D.Q.M. Fay.

Computer Science Department, Queen's University of Belfast.

Introduction

This paper summarises some results of applying a pattern recognition algorithm to data, the collection of which was reported in an earlier paper [1]. The results seem to be surprisingly good and therefore worth reporting despite the simple method used. In order to obtain reasonable real-time computation speeds, simple algorithms and integer arithmetic were essential.

Pattern recognition algorithm

Following Minneman's hypothesis [2] that the correct selection of a feature set is much more important than the mathematics used on those features, it was decided to identify features that could be computed easily but gave good classification separations. Two suitable features were found to be :-

x - the integral of the signal strength values in a window containing the bottom echo ;

y - a count of the number of data peaks in that window.

More than two feature axes were found to be unnecessary.

A simple archetype clustering algorithm was also used. A large number of the published algorithms [3] involved long computing times in recognise mode. Computer time was kept reasonably short in learn mode, and as short as possible in recognise mode; whenever possible computing tasks were done in learn mode rather than in recognise mode.

The data for each of up to ten archetypes comprised :-

x_{max}, x_{min}, y_{max}, y_{min}: integer values representing Cartesian co-ordinate limits for that archetype ;

r_{max}, r_{min}, θ_{max}, θ_{min}: integer values representing polar co-ordinate limits for that archetype ;

(x_i, y_i): integer pairs for every point learnt for that archetype.

The learn mode was entered only when the operator pressed a key; a dialogue was used to determine whether the point was another one for an existing archetype or was the first point for a new archetype. The archetype data was updated whenever the learn mode was entered.

In the recognise mode the point was compared with all of the archetypes to determine if its (x, y) lay within that archetype's Cartesian co-ordinate limits, and if its (r, θ) lay within the polar limits. A good match was reported if the same archetype match was found unambiguously using both criteria. Other confidence levels were reported depending on the degree of ambiguity. If there was no archetype matching or if there was unresolved ambiguity, a separate algorithm was used to determine which of the previously learnt points was closest to the point being recognised. The closest point was considered to be the one with the minimum value of :-

$$\frac{x}{x_i} + \frac{x_i}{x} + \frac{y}{y_i} + \frac{y_i}{y} .$$

The program outline was as follows :-

```
program bottompattern ;
    :
begin
initialise ;
while more data do begin
                    ping ;
                    collect one set of echo signal strengths ;
                    look for bottom echo ; display depth ;
                    select data window around bottom ;
                    normalise data for standard depth ;
                    compute feature values ; put point into x-y display on
                    screen ;
                    RECOGNISE ; display bottom type  and confidence level ;
                    if keypress then LEARN
                    end ; (* if no keypress this takes 0.5 sec approximately *)
terminate
end.
```

Figure 1 shows the confidence levels that would be reported for various points, given the two archetypes shown.

Results

Figure 2 shows the data that was reported in an earlier paper [1], and some additional results, plotted on axes using the features described above. Good grouping for particular bottom types is seen. Since the program is intended to be interactive, a dynamic display would illustrate the algorithm more effectively.

Comments

The ADC was saturated with bottom echoes from rock in shallow water, thus decreasing the number of peaks in the window around the bottom. Fortuisly this seems to increase the clustering rather than decease it!

The system has been thoroughly tested with real data collected earlier and read from disk, and with partially randomly generated data. Results have been good. Sea trials using real-time data are expected to produce good results.

The data was mostly collected in Strangford Lough which has a very good variety of bottom types and a reasonable depth range; its bottom types are well mapped.

The program described is essentially interactive. The archetypes are improved using interaction with the operator as the system continues to be used. In recognise mode the system works at the rate of about two pings per second; in learn mode the rate is determined by the operator's response.

Programming notes

The program was written in Pascal on an Apple computer.

Integer arithmetic including integer division was used throughout the program.

Sea bottom quality determination using an inexpensive echo-sounder and microprocessor.

The recognition algorithm ran in about one fifth of a second, not including the depth determination.

Set operators and nested if..then..else statements were used to minimise computing times when comparing the archetype matches found using the two different matching techniques.

Instead of using true polar co-ordinates (r, θ), the computing was done with $((\frac{x}{s})^2 + y^2) = \surd r$ and $\frac{x}{sy} = \tan^{-1}\theta$. This considerably decreased the computing times without affecting the validity of the algorithm since $\surd$ and $\tan^{-1}$ are monotonic functions in the appropriate ranges.

A window size of 127 values was used. The maximum possible value for y was 62. Eight bit sample values were used. The maximum value for x was 32512. The factor s was chosen to be 512. The maximum value of $\frac{x}{s}$ was 63. The maximum integer size with Apple Pascal is 32767.

References

1 D. Fay "Digital computer analysis of echo sounder data for fish identification". Inst. Acoustics, Proc. Conf. Acoustics in Fisheries, Hull 1978.

2 M.J. Minneman "Hand written character recognition" IEEE Trans SSC-32, No 2 Dec 1966.

3 A review of some literature on pattern recognition is given in an unpublished honours degree project report by Mr W. Oswald, Q.U.B. 1979.

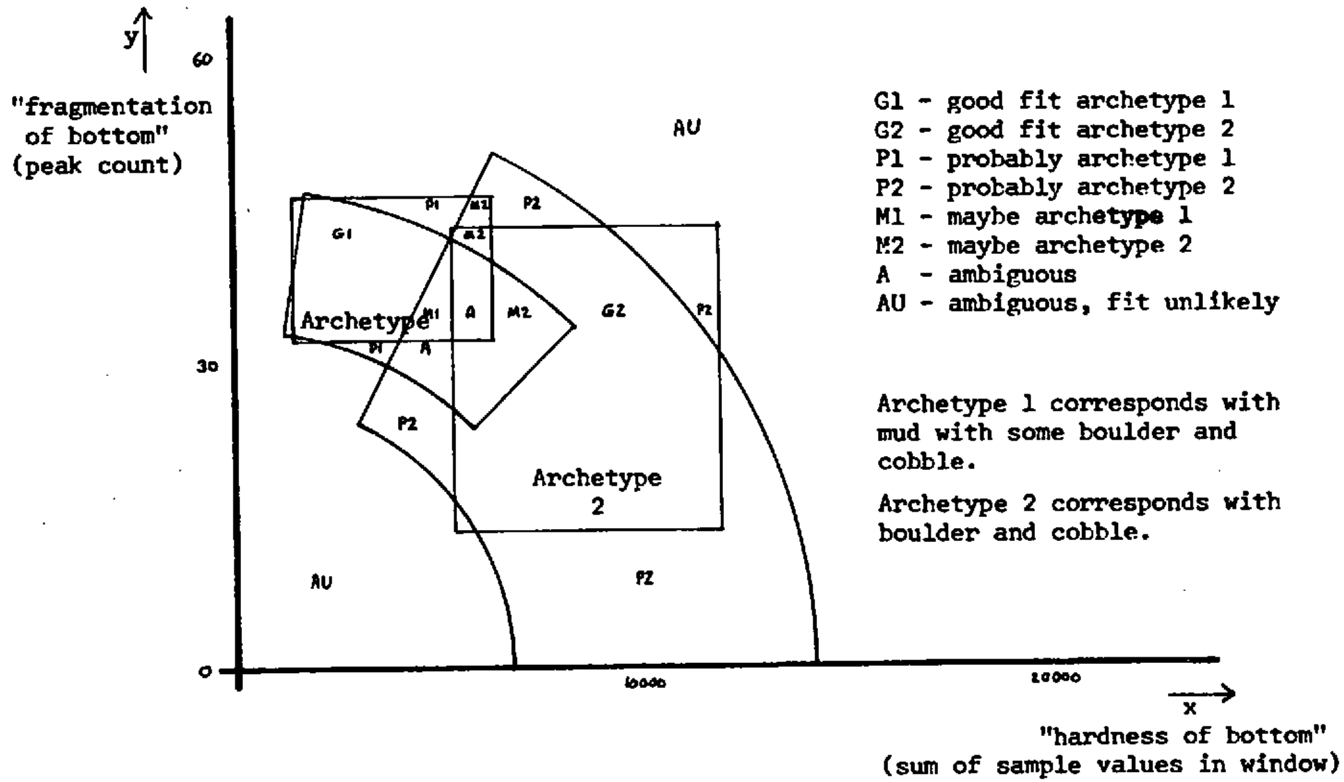

Figure 1 To show how pattern matching algorithm would behave if archetypes overlapped.

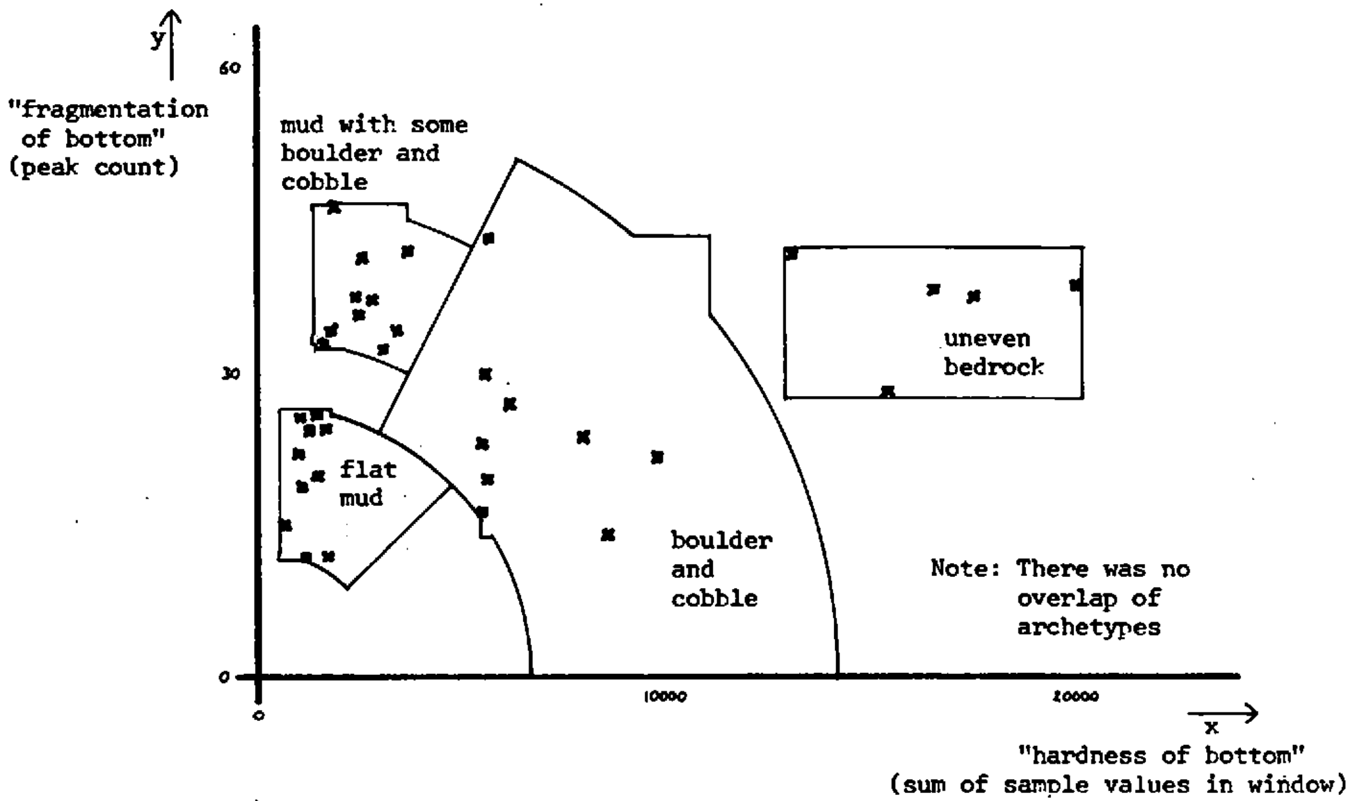

Figure 2 Clustering of data from different echo recordings for particular bottom types.