Factors Affecting The Training Of A WISARD Classifier For Monitoring Fish Underwater

D Chan[1,2], S Hockaday[2], R D Tillett[1], L G Ross[2]
[1]Silsoe Research Institute, Wrest Park Silsoe, Bedfordshire, MK45 4HS, UK
[2]Institute of Aquaculture, University of Stirling, FK9 4LA, Scotland
dickson.chan@bbsrc.ac.uk

Abstract

A non-invasive system for monitoring fish underwater is described. A 3D point distribution model (PDM) model can be fitted on stereo images for fish examination. Currently the model fitting algorithm requires manual initialisation. Therefore experiments were carried out to investigate the usefulness of an n-tuple classifier as a tool to initiate the model fitting method automatically. The experiments were designed to identify factors that will affect the performance and the usefulness of the classifier under the requirements of the fish inspection application. Experimental results show that the classifier is a useful tool for interpreting underwater fish images and could be used as a tool to aid the application in question.

1 Introduction

Knowledge of the status of the salmon stocks in a fish farm is essential for farmers to manage their farm efficiently. However, current fish stock examination procedures are labour intensive and most importantly, cause physical damage and physiological stress to the fish. The damage that is caused by the inspection process will affect the health, appetite and even lead to death of the stock. Therefore it is a very desirable idea to have a non-invasive, stress-free monitoring system that will allow the farmers to examine their stock while they are still underwater.

A system that was designed to meet these needs has been described by Beddow et al [1], Chan et al [2], Hockaday et al [4] and Ruff et al [7]. The system utilised an underwater stereo vision system, that would allow real world lateral length measurements to be extracted from a fish in the stereo images. Therefore biomass or other information concerning the stock can be derived.

One aspect of the system consisted of a model fitting procedure for a 3D Point Distribution Model (PDM) [5], which could be used to extract the lateral length measurement automatically from the images without extensive user interaction to locate individual landmark points on the fish.

However, as described by Chan et al [3], there are certain limitations associated with the current fitting procedure. Therefore experiments on using an n-tuple classifier to overcome some of the limitations associated with the PDM fitting algorithm have been carried out by the authors. Provisional results shows that the n-tuple classifier has a great potential for speeding up the whole estimation process by it's unique characteristic of flexibility, simplicity and efficiency of execution.

This paper will first give a brief overview of how salmon biomass could be estimated using underwater stereo imaging techniques. Secondly a short review of the n-tuple classifier, which is based on the WISARD architecture [8] and how it can be fitted into the frame work of the salmon biomass estimation project will be described. Finally experiments are described on how the performance of the classifier is tested according to the requirements of the application and how well the n-tuple classifier can be used as a tool for aiding the PDM fitting process.

2 Measuring Fish Using an Underwater Stereo Imaging System

A non-invasive method for monitoring salmon stock in a fish farm is proposed by Beddow et al [1]. The system Beddow et al suggested consisted of the use of stereo imaging techniques combined with salmon morphology to extract important information about the stock in the salmon farm. Further suggestions were also made by Chan et al [2] to utilise a 3D Point Distribution Model (PDM) that was developed by McFarlane et al [5]. Chan suggested that the lateral measurements of fish in a stereo image could be extracted automatically using the unique properties of the PDM, the landmarks points on the model, to represent the truss network (figure 1) that is vital for any estimation to be made on the fish.

However, the authors [3] have also discussed the draw backs that are associated with the current implementation of the 3D fish PDM and the quality of images captured in the variable underwater environment.

The problem that is associated with the current PDM fitting algorithm, is that it will require user interaction to locate roughly the position of the fish in the image before the model fitting procedures can be commenced. This is a very labour intensive job and it will be beneficial if this procedure can be automated. Another major challenge that concerns this application is the variability of the underwater environment. Figure 2 is a stereo image pair that was captured in a fish farm on the Western coast of Scotland. The image pair shows how the images of the same scene vary with different camera positions. The image quality is further affected by water clarity, lighting condition of the environment and suspended particles present in the water. Due to the complexity of the underwater environment, simple image segmentation techniques such as thresholding and edge detection will not perform well under such variable conditions. High-level and more complex segmentation techniques can be employed for this task, however these methods will often required large amount of computation resources.

The authors have suggested the problems could be overcome by the use of an n-tuple classifier, which is based on the implementation of the WISARD architecture [8]. Provisional experimental results are promising and further experimental results have shown that the classifier could be used to solve the problems in question.

3 Image Segmentation

The grey level image in figure 2 is too complicated for the classifier to be applied to. Therefore the images are first segmented and a binary image is produced as shown in figure 3. The segmentation process is based on simple image subtraction and thresholding algorithms [3], as shown in equations (1) and (2):

$$R(i,j) = \begin{matrix} 1 & if & D(i,j) \geq T \\ 0 & if & D(i,j) < T \end{matrix} \qquad (1)$$

$$D(i,j) = I_n(i,j) - I_{n+k}(i,j) \qquad (2)$$

where R = resultant image; D = difference of two images; I_n = the n^{th} image in the sequence; k = constant; T = pre-defined threshold value. For these experiments, the value of k and T were set to 3 and 15 respectively.

As a result of the segmentation process, the noticeable "moon" shape of the fish head appears as a very constant feature in the binary image (figure 3). The fish head shape is a binary pattern that can be easily recognised by using a simple classifier, such as WISARD. The advantage of segmenting the images using the above method is that the background of the image will stay almost constant, therefore only the swimming motion of the fish will be detected in the resultant image. Hence, an n-tuple classifier was developed and trained to identify this distinctive fish head shape.

4 N-tuple Classifier – WISARD

The n-tuple classifier is one of the oldest practical pattern recognition methods based on distribution computation and amenable to description in term of neural network metaphors. Although the n-tuple classifier is not famously popular compared to some other methods, such as multilayer perceptrons, the n-tuple classifier does have its own advantages over a variety of pattern recognition algorithms [6]. The most noticeable advantages that are offered by the classifier are its speed of execution and simplicity of implementation. The training of the classifier is a one-shot memorisation process; computationally simple compared to other equation solving and minimising methods; implementation of the algorithm is relatively simple and straightforward. The n-tuple classifier implemented for this experiment is a simulation of the WISARD adaptive image classifier that was developed by Stonham [8].

WISARD is a trainable binary pattern classifier, consisting of a Look Up Table (LUT) which holds information about the pattern that the classifier tries to recognise. The LUT training is a two-stage method. Firstly, a set of n-tuples is produced from the binary training image. Each tuple is made up of n elements/pixels from the image (n = 8 for this experiment) which represent an address in the LUT. The choice of which pixels in the image are used to make up each tuple is defined once (pixel-tuple map). Each element of a tuple is selected randomly and each pixel only contributes to one tuple. Finally, the corresponding LUT address is marked to indicate a particular n-tuple has been encountered during training. These steps are repeated by applying the same pixel-tuple map to each image in the training set.

During the recognition process, the content of the test image is broken down into n-tuples according to the pixel-tuple map. A test is then performed on each test tuple against their corresponding LUT entry. A tuple is said to be fired, when the address denoted by the tuple was marked during training. A score, R_v, of the test image can be calculated:

$$R_v = \frac{F_v}{T} \tag{3}$$

$$F_v = \sum_{i=1}^{T} f(\alpha_i(u), \alpha_i(v)) \tag{4}$$

where R = score of the test image; T = total number of tuples in the LUT; F = total number of tuples fired; f(a,b) = 1 if a = b and 0 otherwise; $\alpha_i(x)$ is the i-th n-tuple of the pattern x; u = the pattern embedded in the LUT; v = test pattern.

R_v indicates how closely correlated the test image is with the training set. Finally R_v is tested against a pre-defined acceptance threshold or confidence level (CL), in order to decide if the pattern is presented in the test image.

$$P_v = \begin{cases} 1 & if \quad R_v \geq CL \\ 0 & otherwise \end{cases} \tag{5}$$

where P = classification result of the test image; CL = acceptance confidence level.

5 Experiment

The aims of the experiments are to assess the usefulness of the n-tuple classifier as an automatic detector for initiating the PDM fitting algorithm. They also investigate how the performance of the classifier varies according to the use of different training set and the amount of prior knowledge that is available to the classifier.

5.1 Description of Images

Underwater stereo image sequences of free swimming salmon were captured in a fish farm on the Western coast of Scotland. Images were captured (25 FPS) and stored onto a computer hard drive. The images in each sequence were then pre-processed according to the segmentation procedures described in the previous section. In total 11 sequences were captured and each sequence consisted of 12 pre-processed binary images.

5.2 Description of Training Example Images

The Look Up Tables (LUT) in these experiments were trained with different sets of training examples, in order to allow the assessment of how the performance of the n-tuple classifier varied according to the training examples. The LUT was trained with examples such as the ones shown in figure 4. The resolution of each training image is 96x38.

Five LUTs were constructed for these experiments using different sets of training images:

LUT name	Description	Abbreviated as
LUT1	5 training images from a fish in a single sequence.	5shs
LUT2	5 training images from different fishes in different sequences.	5dhd
LUT3	8 training images from different fishes in different sequences.	8dhd
LUT4	Same set of training image as LUT3, with noise removed from the training images manually.	8dhdc
LUT5	9 training images from different fishes in different sequences, with noise removed from the training images.	9dhdc

Table 1, LUT definitions.

5.3 Experiment Procedures

The experiments were carried out in two parts using the LUTs listed in the previous section. In the first part, the classifier will only perform matching and classification on a test pattern with tuples that contain at least one black element/pixel. N-tuples that contain only white pixels (these tuples will be referred to as white-tuples, and tuples consisting of at least one black pixel will be referred to as black-tuples in the rest of the paper) will be ignored by the classifier. The reason for ignoring the white-tuples is because otherwise the classifier will return a high match score, R_v, when the classifier is applied onto a completely white image [3].

Then the experiment will be repeated taking into account the number of white pixels that are expected to be in the test image. This was achieved by simply recording the number of white-tuples that were encountered during the training stage. During the matching process the number of white-tuples encountered from the test image will be recorded, and a test will be carried out to see if the number of white-tuples from the test image exceed the expected number of white-tuples that was generated during the training stage. That is the condition in (5) is replaced by (8):

$$W_v = \sum_{i=1}^{T} w(\alpha_i(v)) \tag{7}$$

$$TW_v = \frac{W_v}{T} \tag{6}$$

$$P_v = \begin{cases} 1 \quad if \quad (R_v \geq CL) \,\&\, (TW_v \leq EW) \\ \quad 0 \qquad otherwise \end{cases} \tag{8}$$

where W = number of white-tuples generated from the test image; T = total number of tuples in the LUT; TW = ration of white-tuples generated from the test pattern; w(a) = 1 if a = white-tuple and 0 otherwise; $\alpha_i(x)$ is the i-th n-tuple of the pattern x; P = classification result of the test image; CL = acceptance confidence level; EW = expected ratio of white-tuples; v = test pattern.

The purpose of splitting the experiment into two parts as described above is to assess how the performance of the classifier will vary according to the amount of the prior knowledge on the pattern that is available to the classifier.

6 Results and Discussion

For each image in the test set, the highest "fish-head" and noise score were recorded as shown in figure 5. A graph was drawn for the performance of the classifier according to the LUT used. The graph shows the distribution of the "fish-head" score and the noise score when the classifier was applied to the test data. The results are summarised as shown in table 2.

Graph No.	Description of LUT used and matching method	Noise ratio (N)
1	Using LUT1 (5shs), match only on black-tuples	0.329
2	Using LUT2 (5dhd), match only on black-tuples	0.231
3	Using LUT3 (8dhd), match only on black-tuples	0.197
4	Using LUT4 (8dhdc), match only on black-tuples	0.194
5	Using LUT5 (9dhdc), match only on black-tuples	0.155
6	Using LUT2 (5dhd), match on all tuples	0.104
7	Using LUT3 (8dhd), match on all tuples	0.095
8	Using LUT4 (8dhdc), match on all tuples	0.063
9	Using LUT5 (9dhdc), match on all tuples	0.045

Table 2, Description of results of the n-tuple classifier

The graphs show how the fish-head and noise scores are distributed with respect to how the classifier is trained and used during the classification process. The ideal results should show the two distribution separated from each other without much overlapping. That is, the classifier will be able to identify the fish head in the image by a straightforward thresholding value, CL, that can be defined by drawing a decision boundary at the point where the two distributions overlapped. Finally the noise ratio of the results are calculated for comparing the performance of the classifier (the lower the noise ratio, the better the performance).

$$N = \frac{F_{+ve}}{T_{+ve}} \tag{9}$$

where N = noise ratio; F_{+ve} = the area under the noise graph on the RHS of the decision boundary; T_{+ve} = the area under the fish-head graph on the RHS of the decision boundary.

Graphs 1 and 2 show that the classifier will perform better if the training set is extracted from different fish over different sequences. The noise ratio, N, for graph 1 and 2 are 0.329 and 0.231 respectively. The observation obtained can be explained by the fact that the pattern embedded in LUT2 captured the variability of the fish head shape from different fish. Therefore when the LUT is applied to the test pattern, the classifier will be able to coping with the fish head shape variations in a more flexible manner than LUT1.

Graphs 2 and 3 show that the classifier executed the recognition process with better results if the LUT is trained using a large training set (N for graph 3 is equal to 0.197). The reason for such a performance boost by simply increasing the size of the training set is obvious because the more the classifier can learn about a pattern from examples, the better it should be able to perform the test and hence only accept patterns which are highly correlated to the embedded pattern in the LUT. However as pointed out by Rohwer et al [6], the classifier can also be easily over-trained by using a training set which is too large or where too much variation is present in the examples. Therefore extra caution is required to make sure the LUT is not over-trained.

Graphs 3 to 5 show that the classifier performance is improved by a cleaner set of training images. The images used in LUT4 and LUT5 have been clean up manually to remove random noise from the image. This procedure should allow the LUT to be trained only on the emphasis of the distinctive "moon" fish-head shape and no extra noise information will be recorded into the LUT. Therefore during the recognition process, the classifier can "concentrate" on the pattern that is of interest and ignore all the unnecessary information caused by noise. The noise ratio for graphs 3, 4 and 5 are 0.197, 0.194 and 0.155 respectively.

In graphs 6 to 9, the experiment was repeated, but now also using information concerning the number of white-tuples that are present in the training pattern and in the test pattern. The reader should notice how much further the two distributions are separated from each other in these graphs when compared with the results that were generated without taking into account the white-tuples in the images. The noise ratio of graph 6, 7, 8 and 9 are 0.104, 0.095, 0.063 and 0.045 respectively. The sudden increase in performance by incorporating prior knowledge concerning not only the pattern itself, but also the amount of information within the test region is understandable. The classifier can perform a test based not only on the pattern that is embedded inside the LUT, but also on other criteria that are available in the test image. Therefore, performance of the classifier should be expected to increase accordingly.

From all the experimental results, it has been shown that the performance of the n-tuple classifier varies according to, how the training set is chosen, number of training examples used, level of noise present in the training image and amount of prior knowledge that is available to the classifier.

The results from these experiments suggest that the n-tuple classifier offers a simple, efficient but accurate algorithm for locating fish position in an underwater image. Therefore the use of an n-tuple classifier should allow the PDM fitting algorithm to be initiated automatically with high accuracy. Hence, the process of the extracting lateral length information from the fish in the stereo images using could be done automatically.

7 Conclusion

This paper described a system that will allow fish farmers to inspect their valuable salmon stocks without the stress introduced by current examination practices. A segmentation and pattern recognition technique, based on WISARD, is described in detail including how the technique can be used to aid the remote salmon inspection process.

Previous results in earlier publication [3] showed the potential of the WISARD algorithm in processing underwater images for this particular application. In this paper a more detailed experiment were described in order to assess the performance and usefulness of the n-tuple classifier in the application in question.

It has been found that the WISARD classifier performance will be affected by factors that are associated with the training of the LUT. These factors included:

- Training examples used in the training set.
  - The classifier will give better performance if the training set data are collected over a range of different fish from different image sequences (Graphs 1 and 2).
- Number of examples used in the training set.
  - The bigger the training set, the better the classification performance. However it should be noticed that the LUT could be over-trained, therefore poor performance from the classifier as a result. (Graphs 2 and 3).
- Amount of noise included in the training images.
  - The less noise the training patterns contained, the better the classification performance (Graphs 3, 4 and 5).
- Amount of prior knowledge available to the classifier concerning the pattern and the test image.

- More prior knowledge available to the classifier, the better the classification performance (Graphs 6,7,8 and 9).

The results from these experiments suggested that the n-tuple classifier is a suitable tool to be used for fish recognition in underwater images, hence it is a suitable tool to be used for monitoring fish underwater.

References

[1] Beddow, TA et al. Predicting biomass of Alantic salmon form morphometric lateral measurements, Journal of Fish Biology, 1996.
[2] Chan, D et al. Image processing for underwater measurement of salmon biomass, Digest of IEE Colloquium on Underwater Application of Image Processing, London, UK, 25 March 1998.
[3] Chan, D et al. A trainable n-tuple pattern classifier and its application for monitoring fish underwater, to be presented at the seventh International Conference on Image Processing and its Applications, University of Manchester, UK, 12-15 July 1999.
[4] Hockaday, S et al. Using stereo image pairs to measure mass in strains of Atlantic salmon, Salmo salar L. Proc. Sensor and their Application VIII Conf., UK, 7-10 September 1997.
[5] McFarlane, NJB et al. Fitting 3D point distribution models of fish to stereo images. Proc. British Machine Vision Conf., Essex, UK, 8-11 September 1997.
[6] Rohwer, R et al. The Theoretical and Experimental Status of the n-tuple Classifier. Neural Networks, Vol 11, No. 1, pp 1-14, 1998.
[7] Ruff, BP et al. FishSizing and Monitoring Using a Stereo Image Analysis System Applied to Fish Farm. Aquacultural Engineering, 155-173, 1995.
[8] Stonham, IJ. Practical face recognition and verification with WISARD. Aspects of face processing, Nartinus Nighoff, 1986.

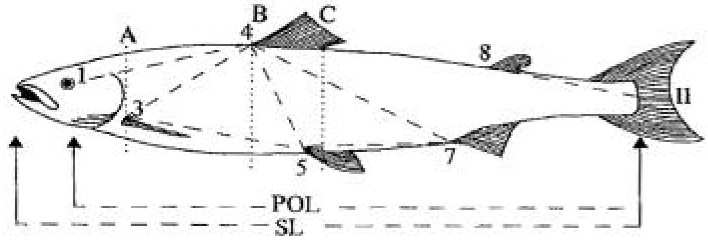


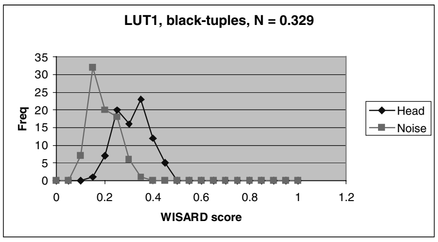
Figure 1 , Salmon truss network

Figure 2, Underwater stereo image pair

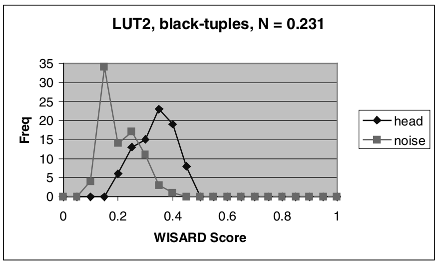
Figure 3, Examples of training images

Figure 4, Pre-processed image

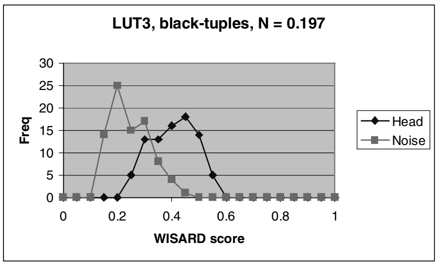
Figure 5, WISARD in action



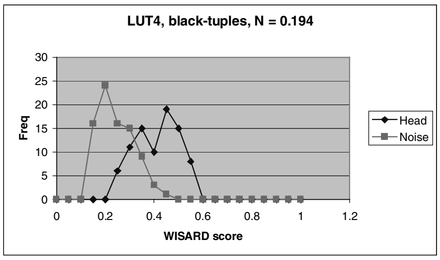
Graph 1, WISARD scores with LUT1



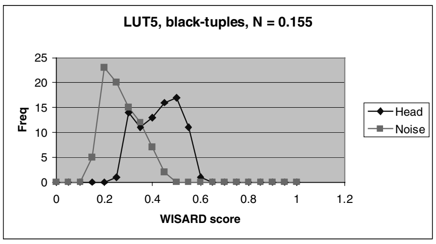
Graph 2, WISARD scores with LUT2



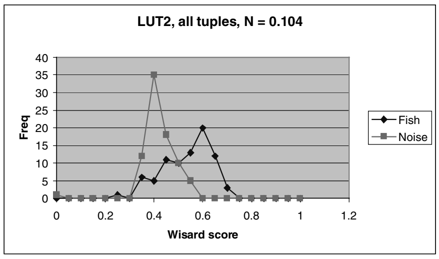
Graph 3, WISARD scores with LUT3



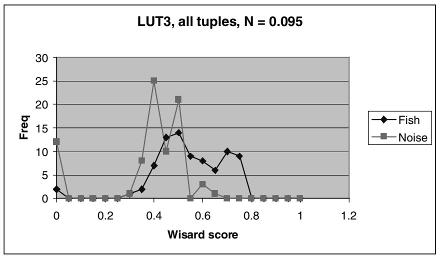
Graph 4, WISARD scores with LUT4



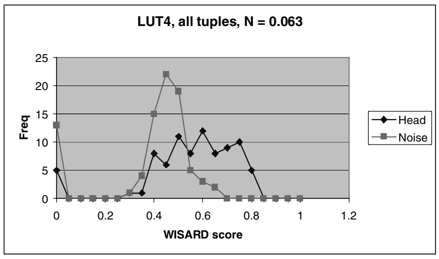
Graph 5, WISARD scores with LUT5



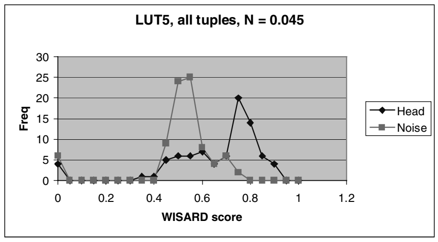
Graph 6, WISARD scores with LUT2



Graph 7, WISARD score with LUT3



Graph 8, WISARD score with LUT4



Graph 9, WISARD score with LUT5