ELSEVIER

Fisheries Research 51 (2001) 11–15

FISHERIES RESEARCH

www.elsevier.com/locate/fishres

Fish species recognition using computer vision and a neural network

Frank Storbeck[a,*], Berent Daan[b]

[a]*Department of Biology and Ecology, Netherlands Institute for Fishery Research, P.O. Box 68, 1970 AB IJmuiden, The Netherlands*
[b]*Lost Boys interactive, Herengracht 410, 1017 BX Amsterdam, The Netherlands*

Received 21 March 2000; received in revised form 10 July 2000; accepted 19 July 2000

Abstract

A system is described to recognize fish species by computer vision and a neural network program. The vision system measures a number of features of fish as seen by a camera perpendicular to a conveyor belt. The features used here are the widths and heights at various locations along the fish. First the measured values are used as input values to a neural network, together with the information on the species. The network is trained to recognize the species from these input data. To decrease the time to train the network, a learning rate, a momentum factor and the elimination of non-contributing connections and nodes were introduced. Testing of the network showed that more than 95% of the fish could be classified correctly.

Keywords: Computer vision; Neural nets; Pattern recognition; Process control; Recognition of fish

1. Introduction

As part of the EU-project *Integrated quality assurance of chilled food fish at sea* a vision system has been developed to determine the weight of flatfish on a conveyor belt passing a camera (Storbeck and Daan, 1991). Due to the constraints of the project, the weight estimation procedure was limited to plaice (*Pleuronectes platessa* L.), although the method was applicable to other species. However, observed volume–weight relationships were species dependent (Fig. 1). Therefore, for fish not sorted to species, a method capable to determine the fish species is a necessity.

Because the same fish fed into the system may produce different numbers, a solution should be robust enough to tackle this problem. Strachan et al. (1990) describe various methods used to determine species. Shape analysis appears to be the most promising one, classifying a correct fraction of 90% of the fish seems possible. However, building a database with shape descriptors is a complex job, while matching an observed pattern with all those present in the database takes up a lot of computing time.

People are much better than machines at classifying objects but they are not quicker (McLelland et al., 1986). This implies that an approach in which the machine mimics the way people classify objects they observe might do a better job than conventional ones based on analytical methods. However, simulating human complexity of the brains is not feasible. This would mean constructing a network that mimics

* Corresponding author. Tel.: +31-255-564-790; fax: +31-255-564-644.
E-mail addresses: franks@rivo.wag-ur.nl (F. Storbeck), daan@lostboys.nl (B. Daan).

0165-7836/01/$ – see front matter
PII: S0165-7836(00)00254-X

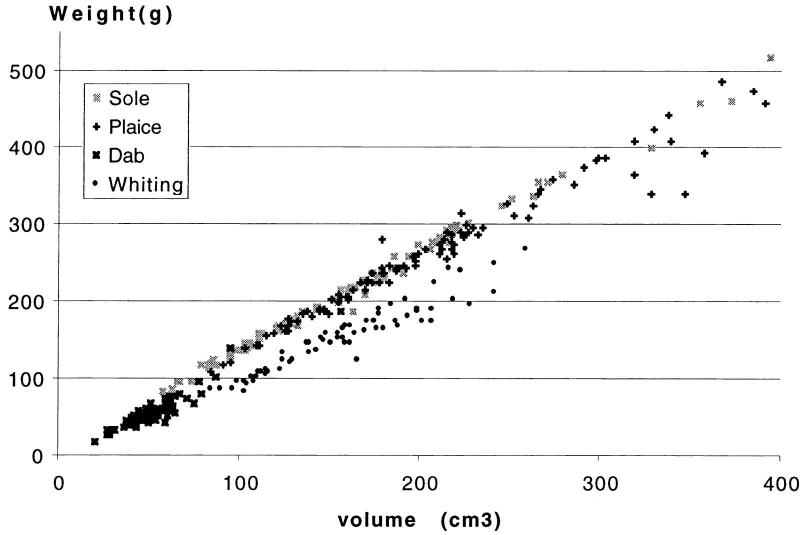

Fig. 1. Observed relationships between the measured volume by the Fish–Vol system and the weight of fish for plaice, sole, dab and whiting (from Storbeck and Daan, 1991).

10^{10} neurons each having connections to 10^3–10^5 fellow neurons (Rumelhart and McLelland, 1986). Mynsky and Papert (1969) pointed out that for most problems a network must at least have three layers to be able to map a number of input values to a specified output pattern. The first layer is used as input, the second is a so-called hidden layer and the third one is the output layer.

To circumvent the construction of a complex database with shapes and a matching algorithm, a flexible system with use of a neural network is presented.

2. Method

The configuration used is described by Storbeck and Daan (1991). By viewing a projected line under an angle from a different direction, a camera "sees" a curved line, representing the shape of the object. (Fig. 2).

The system incorporates a small conveyor belt with a length of 1.7 m and a width of 0.4 m. The speed of the belt is 0.21 m/s. On top of this belt a construction was made to hold a video camera (SONY, XC-77CE) with a 8–16 mm zoom lens. The axis of the camera is orientated perpendicularly above the belt in such a way that the scanlines of the image are parallel to the direction of the movement of the belt.

A helium–neon laser (LIMAB, 2 mW, class 3B) with a wavelength of 632.8 nm is placed at an angle of 45° to the plane of the belt. By putting a cylindrical lens in the laser bundle the latter spreads out in one direction and a bright light line perpendicular to the direction of movement of the conveyor belt is created (Storbeck and Daan, 1991).

For the image processing, three VME-bus based printed circuit boards from DATACUBE Inc. were used: MAXSCAN frame grabber, ROISTORE video memory and MAXGRAPH display unit. The latter is

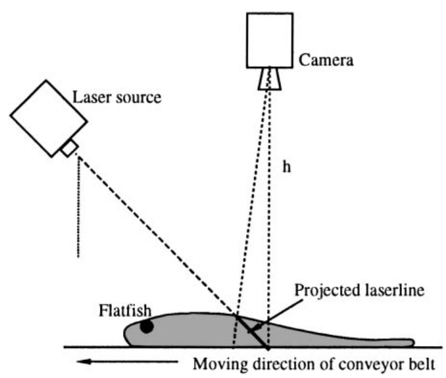

Fig. 2. Concept of the vision system that computes the volume of a fish on a conveyor belt (from Storbeck and Daan, 1991).

not essential but is handy during the development phase of the software to see what is happening on a separately connected video monitor (PHILIPS, CM8833). For developing the software, the VME-system is connected to a workstation (SUN 4/330, SUNOS 4.0.1). This is the platform we used to generate and train the neural network. The computer program is based on a sample program from Parker (1989), modified to perform four tasks: (1) generating a network, (2) training the network with a learning set, (3) testing it and (4) exporting the network. This export file can be imported by the computer program that is used to measure the volume of the fish. It uses the trained network to guess the fish species.

Storbeck and Daan (1991) describe how the start and the endpoint of the curvature of the distorted line determines the width of the fish at various locations along its length. Fig. 3 shows how the width and the observed distortion of the projected laser line were encoded into a pseudo image. Length of the fish was scaled to the width of the pseudo image. At the bottom of the pseudo image, the width of the fish is represented by proportional columns with pixel values of +1 (Rosenfeld and Kak, 1982). From the top of the pseudo image similar columns are created that are proportional to the maximum distortion of the laser line along the length of the fish. The rest of the pixels got values of -1. All the pixel values were then used as the input values for the input nodes of the neural network.

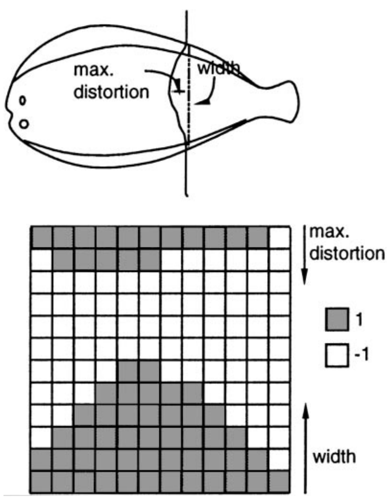

Fig. 3. Coding of the measured width and maximal distortion along the movement direction of the fish into a pseudo image, which is the input layer to the neural network. (here a 12×12 raster is displayed for clarity, in reality a 20×20 raster was used).

The concept of the network is based on conventional backpropagation (Rumelhart et al., 1986). Improvements were made by using a sigmoidal trigger function

$$G(x) = \frac{1}{1 + e^{\sigma(\tau - x)}}$$

defining the output level of a node, where τ is a threshold value that influences the level at which the output value changes from a low value to a high one and vice versa, and σ a parameter that changes the steepness of the trigger function, i.e. the sensitivity of the output value to the input values. To regulate the speed of learning, a mechanism is used to change the weights of a connection in the following way:

$$\Delta w_i(t) = \rho(1 - \mu)x_i\delta + \mu\Delta w_i(t - 1)$$

in which $\Delta w_i(t)$ is the change in weight after learning cycle t, ρ the factor that influences the learning rate, μ the momentum factor and δ the backpropagated error:

$$\delta = G'(x)\sum_j w_j\delta_j$$

$G'(x)$ is the first derivative of the sigmoidal trigger function $G(x)$. The index i runs over the domain of all connections that provide input to the node and the index j covers all connections to which the node provides its output. The momentum parameter determines how history of changes influence a new weight change. The effects of introducing the learning rate and the momentum factor were tested on a traditional XOR network.

In the neural net there can be connections that have only a small weight factor. They limit the influence of the underlying node to the higher one. An algorithm was introduced to remove such connections from the network and if this resulted in a situation where a hidden node had no connection to another layer, this node was also removed, including all its remaining connections.

During sea trials fresh caught fish was used to train and test the network. A sample with sole (*Solea solea*), plaice, whiting (*Merlangius merlangus*), dab (*Limanda limanda*), cod (*Gadus morhua*) and lemon

Table 1
Number of fishes used for training and tests

Species	Training set	Test set
Sole	50	50
Plaice	48	48
Whiting	42	43
Dab	46	46
Cod	16	16
Lemon sole	49	48

sole (*Microstomus kitt*) was split randomly in two, a training set and a test set. Table 1 shows the number of fishes actually used for separate training and testing.

3. Results

The effect of introducing a momentum factor (μ) is shown in Table 2. With a factor 0.9 the learning speed was a factor of 10 higher. A learning rate (ρ) of 4.0 gave also a faster convergence than a rate of 0.4.

The output layer of the network consists of eight nodes. The network was trained in such a way that node κ ($\kappa = 0, \ldots, 7$) had to be one when a fish of species with identification number κ had passed the vision system, while all the other output nodes remain zero. The idea behind this is that the node value can be used to operate on a valve that causes the fish to be moved into a bin with number κ.

Table 2
Speed improvement by introducing a momentum factor for a simple three-layer XOR network

Momentum	Number of iterations needed
0	400
0.9	38

The first network had an input layer of 40 nodes, a hidden layer with 20 nodes and an output layer with eight nodes, of which only the first six were used in this study. After training this network, only 60% of the fishes of different species were classified correctly. Results improved considerably when two hidden layers were implemented. The final network consisted of 400 input nodes, two hidden layers with 30 nodes each and one output layer with eight nodes. The network was fully connected, giving a total of 13080 connections. Computations were very slow. Therefore, a strategy was adopted to eliminate those connections whose weight were below some threshold. This reduction of the network had two advantages. Computation became less time consuming and the convergence within each set of learning iterations showed a steep increase. After 600 learning cycles (each cycle using about 250 fishes for training) and after elimination of all connections with a weight less than 0.6 after every 50 learning cycles, a network remained with 447 nodes and 676 connections between them. With this strategy, training of the network took about 15 min and classification was done in 7.6 ms.

Table 3 shows how the test set with fishes was classified. Out of 251 fishes, five were classified incorrectly.

4. Discussion

The idea that synapses with a relative small weight do not contribute significantly to the output of a node worked quite well. The network underwent a

Table 3
Results of classification by the network

Species	Classified by the network as						Error (%)
	Sole	Plaice	Whiting	Dab	Cod	Lemon sole	
Sole	50	0	0	0	0	0	0
Plaice	0	47	0	1	0	0	2
Whiting	0	0	41	0	2	0	5
Dab	0	0	0	46	0	0	0
Cod	0	0	1	0	15	0	6
Lemon sole	1	0	0	0	0	47	2

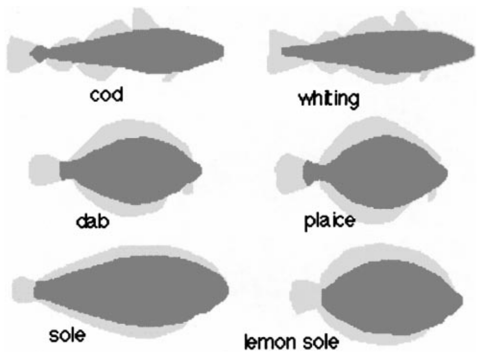

Fig. 4. Contours of the six fish species used (after Knijn et al., 1993).

considerable reduction and this resulted in a faster convergence. Also the introduction of a momentum factor (μ) gave good results. Using a learning rate (ρ) gave improvement but of less significance.

The idea to use a pseudo image as input for the neural net was a lucky one. Before this idea emerged, the convergence of the networks was slow and the results were not very encouraging. After using a pseudo image, most of the work was tuning the number of input nodes and the number of hidden layers. Information collected for weight estimation should also reveal the species of the fish. Deciding which information to use was more an art than a science. The final choice made is by no means the 'one and only' solution for solving this problem. It just did the trick.

The equipment was used to compare the selectivity of two fishing gears and came up with tables of different species caught by the gears, together with the lengths and weights. The fact that the system isn't influenced by physical forces makes it possible to use it on a rolling and pitching ship. It also demonstrates that it is robust enough to be used in a production environment. Freshly caught fishes were directly processed without any preparation.

Because the condition of a fish changes with the season of the year, its geometry changes also. As the network can be trained within a reasonable amount of time, before each trip a new recent learning set could be offered to the system, after which it can be used for production.

That the trained network classifies a lemon sole as being a sole can be expected, because of their similarity in contours (see Fig. 4). The same holds for dab and plaice. Whiting and cod also show similar outlines. With this in mind, the network did its job quite well. Nearly 98% was classified correctly (Table 3).

Acknowledgements

This project was partially funded by the European Union (contract UP-1-67).

References

Knijn, R.J., Boon, T.W., Heessen, H.J.L., Hislop, J.R.G., 1993. Atlas of North Sea Fishes. ICES Cooperative Research Report no. 194, Copenhagen, p. 268.

McLelland, J.L., Rumelhart, D.E., Hinton, G.E., 1986. The appeal of parallel distributed processing. In: Rumelhart, D.E., David, E. (Eds.), Parallel Distributed Processing, Vol. 1. MIT Press, Cambridge, MA, pp. 3–44.

Mynsky, M., Papert, S., 1969. Perceptrons. MIT Press, Cambridge, MA.

Parker, D., 1989. Backpropagation algorithm. Dr. Dobbs J. 89–11, November.

Rosenfeld, A., Kak, C.A., 1982. Digital Picture Processing. Academic Press, Orlando, FL.

Rumelhart, D.E., McLelland, J.L., 1986. PDP models and general issues in cognitive science. In: Rumelhart, D.E, David, E. (Eds.), Parallel Distributed Processing, Vol. 1. MIT Press, Cambridge, MA, p. 131.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning internal representations by error propagation. In: Rumelhart, D.E, David, E. (Eds.), Parallel Distributed Processing, Vol. 1. MIT Press, Cambridge, MA, pp. 318–360.

Storbeck, D.E., Daan, B., 1991. Weight estimation of flatfish by means of structured light and image analysis. Fish. Res. 11, 99–108.

Strachan, N.J.C., Nesvadba, P., Allan, A.R., 1990. Fish species recognition by shape analysis of images. Pattern Recognition 23 (5), 539–544.