# Genetic algorithms for automatic algorithm and parameter selection in ATR applications

P. G. Ducksbury, M. J. Varga, P. J. Kent, S. Foulkes, D. M. Booth

Defence Evaluation & Research Agency, St Andrews Road, Malvern, Worcs, WR14 3PS, UK.

## ABSTRACT

One of the difficulties that has been apparent in applying image processing algorithms not just for automatic target recognition but also for associated tasks in image processing and understanding is that of the optimal choice of parameters and algorithms. Firstly we must select an algorithm to use and secondly the actual parameters that are required by that algorithm. It is also the case that using a chosen algorithm on a different image class yields results of a totally different quality, here we consider three image classes, namely infra-red linescan, dd5 - Russian satellite and SPOT imagery. We are now exploring the use of genetic algorithms for the purpose of parameter and algorithm selection and will show how the approach can successfully obtain results which in the past have tended to be obtained somewhat heuristically.

**Keywords :** Genetic Algorithms, Algorithm Selection, Parameter Selection, Image Processing, ATR

## 1. INTRODUCTION

One of the difficulties in the application of image processing algorithms for automatic target recognition is that of the optimal choice of firstly algorithms for automatic feature detection and recognition and secondly the actual parameters for the chosen algorithm. Our previous research has concentrated [1,2,3] on using Bayesian methods (*Pearl Bayes Networks*) for combining information from feature detectors to obtain more reliable and consistent results. An alternative to actually combining the results of individual detectors is to explore the possibility of actually optimising the choice of those detectors and their required associated parameters. We are now exploring the use of genetic algorithms for this purpose. Our research will show how the approach can successfully obtain results which in the past have tended to be obtained somewhat heuristically. For example, it is often the case that with inherited software and applications that statements are made like 'algorithm W is good on infrared (IR) imagery with parameter set X', but 'algorithm Y is better on SPOT imagery with parameter set Z'. We will show how a genetic algorithm is used to optimise both the parameter set for a given feature detector and the feature detector itself. One of the difficulties in a problem like this can be the size of the search space. Here however the population used for the genetic algorithm is reduced considerably by considering only a constrained space, ie by only examining feasibly members, for example a given parameter is likely only have a certain valid range of values (*a low hysteresis threshold should be less than the high one*). The technique has also been explored for optimising a Kohonen network for ship detection as well as determining parameters for multi-modal image registration.

## 2. FEATURE DETECTION

The feature detectors we have concentrated on have covered the detection of points, lines and regions. In particular, point detection [4], edge detection [4,5], ridge detection [6], region detectors such as Fractal and Wavelet [7] and Pearl Bayes (*belief*) Networks [1]. Most of our previous effort has been concentrated on the region detection as a means of locating urban (*textured*) regions as a cueing aid to assist either an image interpreter or for more detailed processing such as road and building detection. Here we concentrate on the detection of roads and urban regions and obtaining suitable parameters automatically. The road detection has previously been used as a means of registering imagery and digital maps together [8] as a precursor to image distortion correction.

---

Further author information -
Correspondence : P.G.Ducksbury, Email : ducksbury@signal.dera.gov.uk

Most edge detectors that have been developed are based around the assumption of detecting step edges and have been optimised with this goal in mind. The work by Petrou [6] presents a theory for the development of optimal convolution filters for the identification of wide linear features, such as, roads, canals, hedges and rivers. If we consider a cross section taken from of a typical road in an airborne image then it is not just two step edges back to back, it has a characteristic structure. Undoubtedly conventional edge detectors that have been optimised for step edge detection will in fact yield some response upon encountering wide linear structures but this is unlikely to be as good as from a purposely designed filter. Upon using this filter it became apparent that it was very good as a means of road detection on infra-red linescan imagery but not quite so good on the other two image classes, needing different choices of parameters.

We also use an alternative detector which is a more general purpose line finder where the actual ridge detection is based on a Marr-Hildreth [12]. This being followed by a number of post-processing stages such as non-maximal suppression (*including an additional stage for detection and removal of step edges*), skeletonisation, hysteresis, morphology (*for closing up small gaps*) and finally junction filtering (*for short road removal, eg minor side roads, isolated features etc*.) were lines are traced using a decision making algorithm that employs a dictionary of valid pixel patterns.

The detection of textured regions is achieved using three approaches, Belief Network [1], Fractal and Wavelet [7]. With the belief network the idea is to essentially combine a set of statistical evidence measures taken at multiple scales into a confidence (*or belief*) of some event occurring. Prior knowledge is required and this forms a set of multi-dimensional conditional probabilities which relate the inputs (*causal information*) to the outputs of a given node within the network. A set of rules are used which are based on the assumption that casual information that is tightly clustered should naturally generate a higher belief than that which is more dispersed. This is a reasonable assumption since if we consider segmentation of gray level imagery, then neighboring pixels from different regions which have intensity values lying at either end of the gray level spectrum have low probability and for those that belong to homogenous regions, high probability.

The work on fractal analysis concentrated on texture discrimination and assumes that texture can be modeled as fractals (*a reasonable assumption for the type of imagery and application domain of interest*). A fractal has several properties, namely irregular structures and scale invariance (*or self similarity of the structures over scale space*). Examples of fractal sets could be snow flakes, ferns, coast lines etc. For a self-similar fractal a measure is used called the fractal dimension. This does however only measure the topological properties of a fractal and research has shown that two textures can have very different appearances even if they have the same fractal dimension. Further discrimination can be achieved by using a measure called the D-dimensional measure which is equivalent to length, area or volume for 1, 2 or 3 dimensions respectively, Falconer[22]. This measure being independent of dimension. The work developed by Oxford consists of a multiscale analysis algorithm for texture discrimination.

The wavelet analysis uses a scheme which employs the two dimensional local energy of the wavelet transform for texture segmentation. The approach basically consists of four stages such as. Convolution of the image with a set of dilated wavelets to generate a set of wavelet detail images at given orientations and scale. Generation of a set of local energy images (*which removes phase dependency from the detail images*). Scale space fusion which includes inter-scale clustering (*to minimise the spatial localisation problem*) and inter orientation fusion. Texture feature clustering and boundary detection.

This now provides us with two sets of feature detectors and their parameters to optimise. The next required stage prior to the GA is to produce some measure of confidence (cost function) as to how good the feature detectors are, this can often be the most difficult stage as this is often a subjective task.

### 2.1 Confidence Estimation - Roads

The confidence measure that is used is relatively straightforward. For a given curve we trace out its pixel chain and then recursively subdivide this into a set of line segments. Figure 1 illustrates this, on the left we take the pair of points (x1,x3) and fit a straight line segment between them. Then we locate the point (x2) which has the largest perpendicular distance (d) between the line segment and the curve. If this distance is greater than some agreed tolerance then the curve is subdivided into two line segments as shown in the middle, about the point (x2). If the distance is less than the tolerance then the line segment is accepted and the process moves on to remaining sections of the curve. Thus process is recursively applied until the curve has been decomposed into a set of line segments.

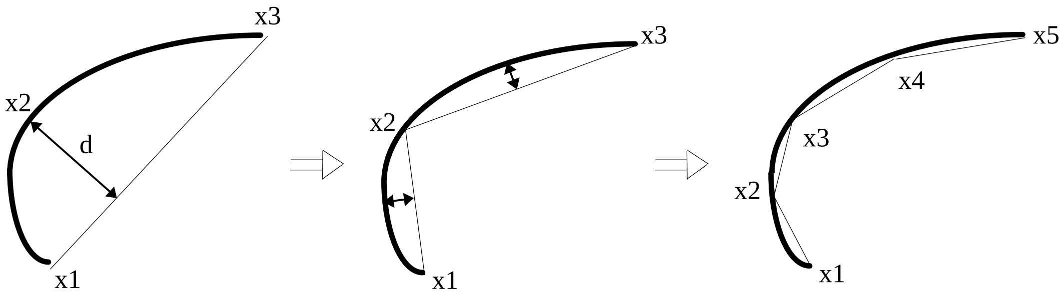

Figure 1; Line confidence estimation.

The ratio of the number of fitted line segments to the number of pixels in the curve provides a very good (*almost a fractal type of*) discriminent for the confidence in straight man made types of object.

### 2.2 Confidence Estimation - Regions

In the case of region detection algorithms the boundary is taken and the skewness $\frac{1}{N}\sum_{j=1}^{N}\left[\frac{x_j - \mu}{\sigma}\right]^3$ is computed for a line placed perpendicular to each boundary point. The skewness characterising the degree of asymmetry of a distribution around its mean. If a boundary was placed in the middle of a field as opposed to the edge of a region then we would naturally expect a different distribution and skewness measure. The sum of differences between the generated confidence outline and the similar for digital urban area outline data is taken as the confidence measure for a particular instance of an algorithm. Other methods have been proposed for confidence measures such as Palmer et. al. [19] who studied the requirements and proposed a performance measure. Alternatively a region either side of a segment of the boundary could have been analysed.

## 3. GENETIC ALGORITHM

### 3.1 Search Strategies

Of the search and optimisation strategies that are available such as calculus based methods, enumerative search and random search we consider the genetic algorithm approach to be the more suitable albeit for this application.

As Goldberg [9] points out the calculus based methods search for local optima either indirectly by solving a set of equations which result from setting the gradient of an objective function to zero or directly by using the function and moving in a direction towards a local gradient referred to as either hill-climbing or steepest descent. The enumerative search methods essentially consider every point in the search space and evaluate the objective function at those points. They have the disadvantage of the actual size of the search space. For the same reason we also discounted the use of random search approaches, although these techniques do have their uses. The random search techniques have been used previously Ducksbury [10] by means of a parallel implementation of Price's [11] controlled random search algorithm for global optimisation of non-linear functions.

### 3.2 Constraints

The problem we are addressing is essentially a constrained optimisation one. Whenever a constraint is violated the solution would be infeasible and should therefore have no fitness value. In some highly constrained domains obtaining feasible points may prove to be a difficult task. Penalty type methods try to obtain information from infeasible solutions by the degrading of the fitness in relation to the relative degradation in the constraint violation. This approach is somewhat difficult in our domain as the objective function is actually an image processing or image understanding program (*or even a suite of programs*) and we may not be able to allow an infeasible member in the first place as the decoded set of parameters may have no sensible meaning to the image processing algorithm. This could form an interesting item for future work to incorporate sensitivity analysis in an attempt to gain information from infeasible points.

### 3.3 Parameter Set Encoding

For encoding the parameter set for these problems we use a simple intuitive and direct mapping of the parameters into substrings within a string (*the analogy here being the genes within the chromosome*). The parameter set for both of the problems is therefore encoded into a bit string with distinct substrings representing particular algorithms and parameters. The bits essentially represent the parameterisation of the problem, for example currently parameter p0 represents the choice between the algorithms.

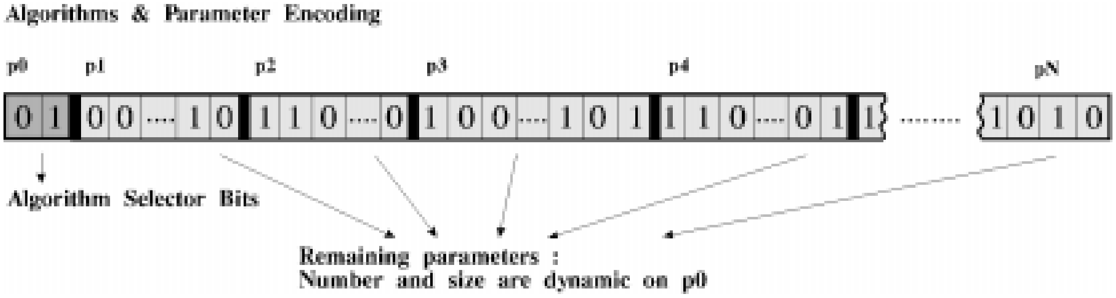


Figure 2; Parameter encoding.

Different algorithms may result in differing string lengths, thus zero padding is included to the size of the maximum string.

For the road detection the various parameters for the algorithms represent items such as filter size, gaussian constant, width of feature, low pass characteristics, low hysteresis threshold, high hysteresis threshold and minimum acceptable junction size. For the region detection algorithms the parameters represent sampling window size, thresholds, filtersizes and the number of scales. Each of these parameters having a predefined dynamic range.

Given this encoding an initial population can be built by randomly generating bit strings. The constraints that are applied at this stage cause the removal of members (*chromosomes*) which have parameters (*genes*) outside the constrained search space as well as the removal of duplicate members. This results in a possibly highly constrained but feasible population. Given this initial population we can now perform the operations of crossover and mutation as described below.

### 3.4 Crossover

In performing crossover we are essentially playing a game of mix and match with members of the population that are highly correlated with previous success. When it comes to the operations of crossover and mutation a variety of different approach's exist, for example Goldberg [9] and the on-line tutorial in [16]. Our approach consists of a combination of the elitist strategy with a tournament selection, all members being chosen using a roulette wheel based selection strategy.

The elitist approach copies the 'best' solutions from one generation to the next thereby guaranteeing that the GA never loses its best solutions. A variety of thresholds being experimented for this with 0.05 being the chosen. The tournament selection chooses one member at random from the population as a parent and then chooses two others, these then essentially compete for the right to breed with the first chosen parent based on which of the two has the highest confidence measure. The roulette wheel selection is used whenever a member of the population is required to be chosen at random. Individual slots within the wheel are weighted in proportion to the members confidence measure (*or fitness value*). This being implemented along the lines of computing the cumulative histogram of the fitness values. A random point then indicates the location at which we take the first member for which the cumulative sum is greater.

The actual crossover of the selected parents is done by a simple exchange of substrings between the two, the location within the parents being randomly chosen. If the new members are either infeasible or duplicates, then they are rejected on the grounds that they would waste valuable population space and reduce the diversity. This crossover process continues to iterate until a new population has been generated. The initial population size is chosen according to a rule of thumb which chooses a population that is at least an order of magnitude greater than the number of parameters.

### 3.5 Mutation

The mutation operation again consists of choosing members at random using the roulette wheel selection and then mutating just a single bit within the members string chosen at random. The relative amount of mutation was set at 0.05 although different ratios where tried, larger amount having a slight detrimental effect on the overall population quality. Again new infeasible and duplicate members are rejected.

## 4. OTHER APPLICATIONS

### 4.1 Ship Detection

In this section we examine the application of a Kohonen neural network to ship detection in ERS SAR imagery. Kohonen networks model the probability density function of feature vectors; objects sharing similar characteristics will cluster together in feature space.

Kohonen neural networks are self-organising, feed-forward networks consisting of an input layer and in this case, a two-dimensional output or clustering layer. Kohonen's algorithm for training a vector quantizer (VQ) network is an iterative process [21]. The initial neuron positions are chosen randomly and each neuron is associated with a local neighbourhood which is reduced in size during training. For each input vector from the training set, the winning neuron is the one whose centre is the least distance from the input vector. The weights of the winning neuron and its neighbours are adjusted to reduce their distance from the input vector. To prevent a single neuron from dominating during training, the Frequency Sensitive Competitive Learning (FSCL) algorithm is often used [21]. Rather then using a local neighbourhood to prevent a single unit from dominating, the winner is penalised so that it has to work harder to win any subsequent input vectors. In this way, a dominant neuron is steadily weakened until other neurons can win some of the input training vectors. The weights of the winning neuron are updated in the same way as the Kohonen weights update procedure.

In order to perform the training the sea background structure was suppressed by convolving the image with a second order differential of a Gaussian digital filter. This filter was designed to give maximum response for ship like objects, while large features, such as sand banks and current fronts were suppressed. Raw pixels were input to the Kohonen network. Performance was measured on differently sized input windows and network architectures. As ships occupy a very small proportion of the sea area it was not possible for the network to learn the ship versus sea mapping because other surface structures dominate. So, although Kohonen networks do not require labelled training pixels, equal numbers of ship and background samples were supplied to avoid dissipating the influence of the ships.

After the Kohonen network has organised itself, the detection algorithm enters a further learning phase during which a validation image was presented to the network, thus enabling those neurons which correspond with ships to be identified automatically. A single ship may cause several different neurons to fire as it passes through the sampling window: one neuron might respond to a rising edge, one to a peak, and one to a falling edge. The resulting network is then applied to a third image in order to evaluate the performance of the network. This involves computing a number of attributes for the network including, the number of true targets detection (TT) and the number of false alarms (FF). The figure of merit for the network was then calculated using: TT / (FF+DT) where DT is the total number of true targets.

### 4.2 Image Registration

Considerable work has been done on multi-modal image registration and it is not the subject of this paper, however the Genetic Algorithm technique used here has been applied with success in this application. Kent [20] has worked on a technique based on mutual information using a probabilistic method of registration which employs intensity values thus avoiding the information loss due to conventional segmentation based techniques. The approach is based on a principal of a causality relationship existing between intensity values in different modality images. Some hand registered SAR and Optical (*as well as optical and IR*) images where used into which translational effects had been introduced. The registration being recovered to sub-pixel accuracy (this is probably greater than the accuracy initially achieved by the hand alignment). Use of the full affine transform is now being considered but using an evolutionary algorithm which is more suitable to the continuous case.

## 5. RESULTS

Figure 4 shows two different types of imagery, the first being a section of a dd5 Russian satellite image of Plymouth whilst the other two are both infra-red linescan images over Bedfordshire taken at night at approximately 3000 ft. Figure 5 shows result for the optimisation of both the algorithms and their parameters for road detection, for the dd5 image the performance of the linefinder filter was purposely degraded thus resulting in the selection of the Petrou filter. As the general purpose linefinder is so highly optimised in terms of postprocessing that without forcibly degrading its performance it would automatically be chosen as the best (*there is currently no postprocessing in the Petrou filter to remove the small line segments*). Figure 6 is an example of the best result for the infra-red linescan image taken from the first generation, showing a far less desirable result. The worst member of the population at the first generation actually resulted in an output with zero road detection. Figure 7, shows the results for the region optimisation using three region finding algorithms. This problem was actually treated as a minimisation of the sum of differences of the outline with that of an urban area digital outline dataset. The left image being the selection of the Belief Network algorithm (*this algorithm was implemented from a cueing perspective and was not intended to produce a detailed outline*). The right image being the worst result obtained, in this case from the fractal algorithm with a poor choice of parameters in particular the thresholding parameter that is applied to the feature image in order to generate the outline.

Figure 8, shows a set of graphs illustrating various characteristics of the genetic algorithm, the first three being for the single-algorithm version. Top left is the algorithm convergence characteristics over a large population. Top right is the convergence with varying population sizes, for populations which are too small we can see that convergence (*certainly to a global optimum*) has failed whereas with the larger populations we move to the vacinity of a good solution very quickly, certainly within a few generations. Bottom left shows the reducing effect on the entire population of increasing the mutation rate, here a straight line equation has been fitted to the data points. Bottom right shows the multi-algorithm version of the GA with different elitist probabilities, we observe improved results with the lower values of this parameter (*although for higher values convergence does eventually occur but requires more generations*). Goldberg [9] pointed out that De Jong [17] had observed similar effects on some functions and suggested that the elitism improves local search at the expense of a global perspective. Figure 3 shows a visual representation of the evolutionary process albeit just the first and last and for the case of optimising two line finder algorithms. The left half is the initial population of chromosomes for 50 members together with a decoded gray level representation of them and the cost function. On the right half of the figure is the result of the 50$^{th}$ generation showing the order that has been introduced, the randomness remaining to the right of the chromosomes corresponds to differences in the hysteresis parameters. As there are only two algorithms the first bit in the chromosome corresponds in this case to the algorithm selector and as can be seen the second algorithm has been removed from the population, this occurring within the first few generations.

Figure 9, shows an ERS image of the English channel near Dover. It can be seen that the sea region has many features, including sand banks and current flows. Superimposed on the image are the results of passing this image through the neural network structure that achieved the best performance. True ship detections are marked with white squares and false alarms are marked with black squares. There is a small number of missed targets, these being very small ships or fishing boats.

## 6. CONCLUSIONS

We have shown that in principle a genetic algorithm is a very suitable search mechanism for optimising both the choice of parameters and algorithms for image processing operations such as road, region and ship detection. In fact any image processing/Understanding algorithm is suitable provided some confidence judgement can be made as to its effectiveness. It is not proposed to use this genetic approach every time some feature detection is required but that it would be used in situations where either a new class of imagery or a new algorithm is encountered and we need to know the most suitable strategy to adopt. Currently no attempt has been made to optimise the performance of the genetic algorithm although parallel processing architectures would offer significant advantages. An alternative to its use here may be in situations like the general purpose linefinder ie, an algorithm that is composed of several sub-algorithms such as filtering, non-maximal suppression, skeletonisation, hysteresis, morphology and junction filtering. The sub-algorithms themselves could be optimised more systematically. Another area that might be of interest for future work would be to examine the posibility of incorporating sensitivity analysis in an attempt to gain some information from infeasible points.

Another important issue is that image processing algorithms must return some sort of a confidence measure with their results. The is a far more widespread issue that effects the whole of image processing and is neccessary not just for our genetic approach but also for the use of higher level reasoning processes within image understanding. For example if we apply an edge operator to an image we should not just receive a binary image as output but idealy some sort of a graph based representation of all the different line segments with confidences.

## 7. ACKNOWLEDGMENTS

Some of this work has resulted from collaboration with the University of Oxford. In particular we wish to thank Zi-Yan Xie at Oxford for software which implements the Fractal and Wavelet research.



## 8. REFERENCES

1. P. G. Ducksbury, 'Parallel Texture Region Segmentation using a Pearl Bayes Network', *British Machine Vision Conference* - BMVC-93, University of Surrey, Sept 1993.
2. P. G. Ducksbury, D. M. Booth, C. J. Radford, 'Vehicle Detection in Infrared Linescan Imagery', *5th IEE Int. Conf. on Image Processing & its Applications*, Herriot Watt University, Edinburgh, UK, July 1995.
3. P. G. Ducksbury, M. J. Varga, 'Region Based Image Content Descriptors and Representation', *6th IEE Int. Conf. on Image Processing & its Applications*, Trinity College, Dublin, July 1997.
4. S. M. Smith, J. M. Brady, 'A new approach to low level image processing', Unpublished report, DERA Chertsey, 1994.
5. J. Canny, 'A Computational Approach to Edge Detection', *IEEE Trans PAMI*, Vol. 8, No. 6, Nov. 1986.
6. M. Petrou, 'Optimal Convolution Filters and an Algorithm for the Detection of Linear Features', Technical Report VSSP-TR-21/90, Department of Electronic and Electrical Engineering, University of Surrey, 1990.
7. Zhi-Yan Xie, 'Multi-scale Analysis and Texture Segmentation', PhD Thesis, Dept of Engineering Science, University of Oxford, 1994.
8. P. J. Kent, D. M. Booth, C. J. Radford, 'Imagery fusion for the AGS - Registration of infra-red linescan imagery with ordnance survey maps', Unpublished report, DERA Malvern, July 1994.
9. D. E. Goldberg, 'Genetic Algorithms in Search, Optimisation and Machine Learning', *Addison Wesley*, 1989.
10. P. G. Ducksbury, 'Parallel Array Processing', *Ellis Horwood*, Chichester, 1986.
11. W. L. Price. 'A new version of the controlled random search procedure for global optimisation', Engineering Department, University of Leicester, 1981.
12. D. Marr, E. Hildreth, 'Theory of Edge Detection', *Proc. R. Soc.*, London, B 207:187-217, 1980.
13. J. Pearl, 'Fusion, Propagation and structuring in Belief Networks', *Artificial Intelligence* 29, pp 241-288, 1986.
14. D. J. Nettleton, R. Garigliano, 'Large ratios of mutation to crossover; the example of traveling salesman problem', *SPIE* vol 1962, Adaptive Learning Sys. II (1993).
15. J. E. Shamblin, G. T. Stevens, 'Operations Research - A fundamental approach', *McGraw-Hill*, 1974.
16. -, 'The Genetic Programming Tutorial Notebook', http://www.geneticprogramming.com/
17. K. A. De Jong, 'An analysis of the behaviour of a class of genetic adaptive systems', Doctoral dissertation, University of Michigan, 1975.
18. S. Hickey, 'Genetic Imaging', Image Processing, October 97, pp 20-23, *European Technical Publishing Ltd*.,
19. P. L. Palmer, H. Dabis, J. Kittler, 'A performance measure for boundary detection algorithms', *Computer Vision and Image Understanding*, Vol 63, No 3, pp 476-494, 1996.
20. P. J. Kent, 'Evaluation of Probabilistic Methods for multi-modal Image Registration', Unpublished report, DERA Malvern, March 1996.
21. S. C. Ahalt, K. Krishnamuthy, P. Chen, D. E. Melton, 'Competitive learning algorithms for vector quantisation', *Neural networks*, 3, 1990.
22. K. Falconer, 'Fractal Geometry', *John Wiley & Sons*, 1990.

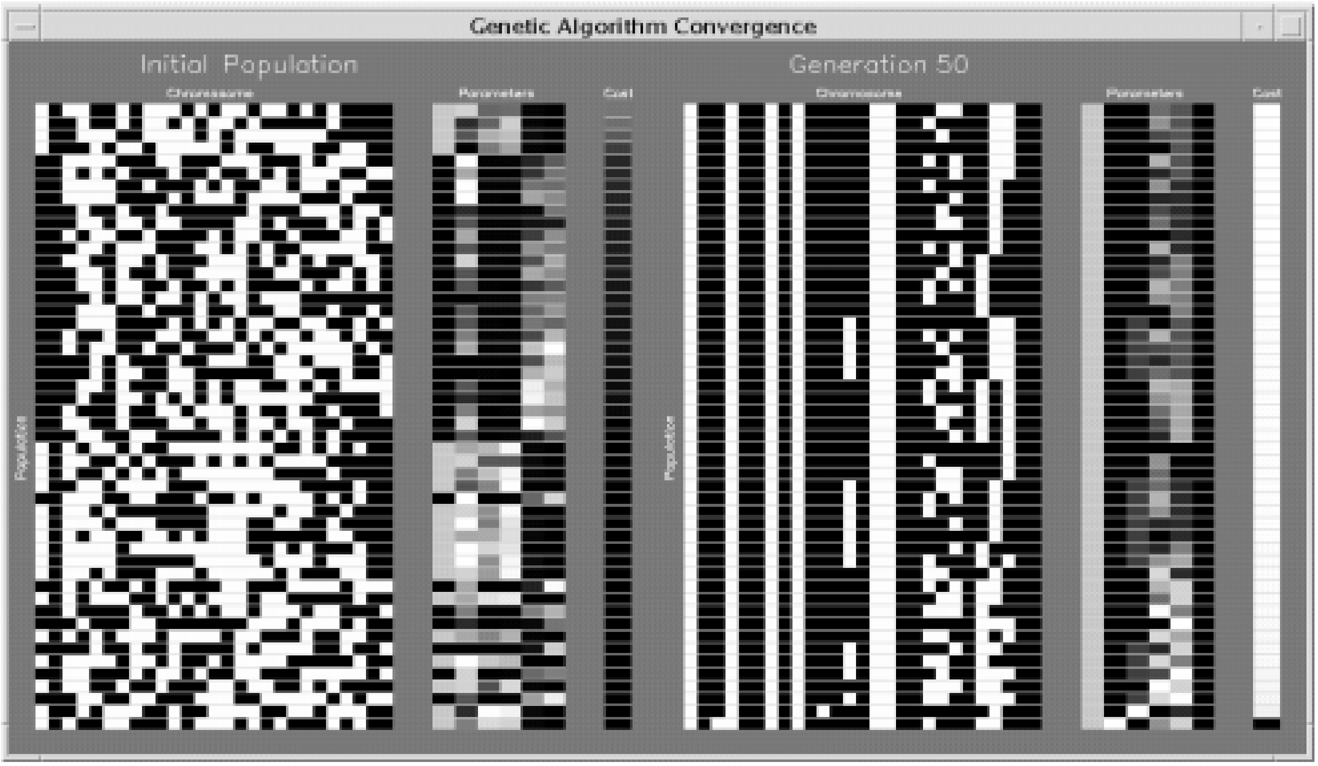


Figure 3; Genetic Algorithm Convergence - II

Figure 4; Top Left - dd5 image, Plymouth; Top Right & Bottom - Infra-red Linescan images.

Figure 5;  Final converged result, generation 50, (confidence images).

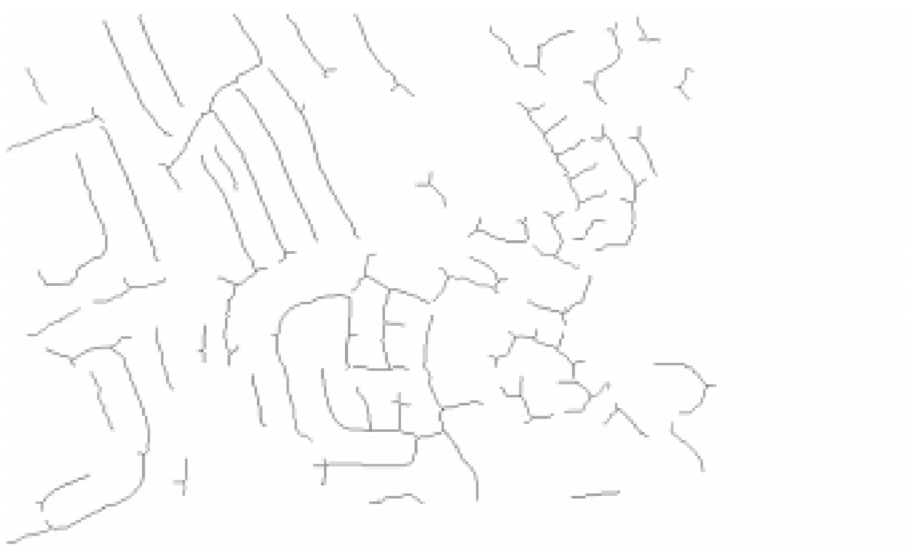

Figure 6; Best result at generation 1

Figure 7; Urban region optimisation, left - best result, right - worst result.

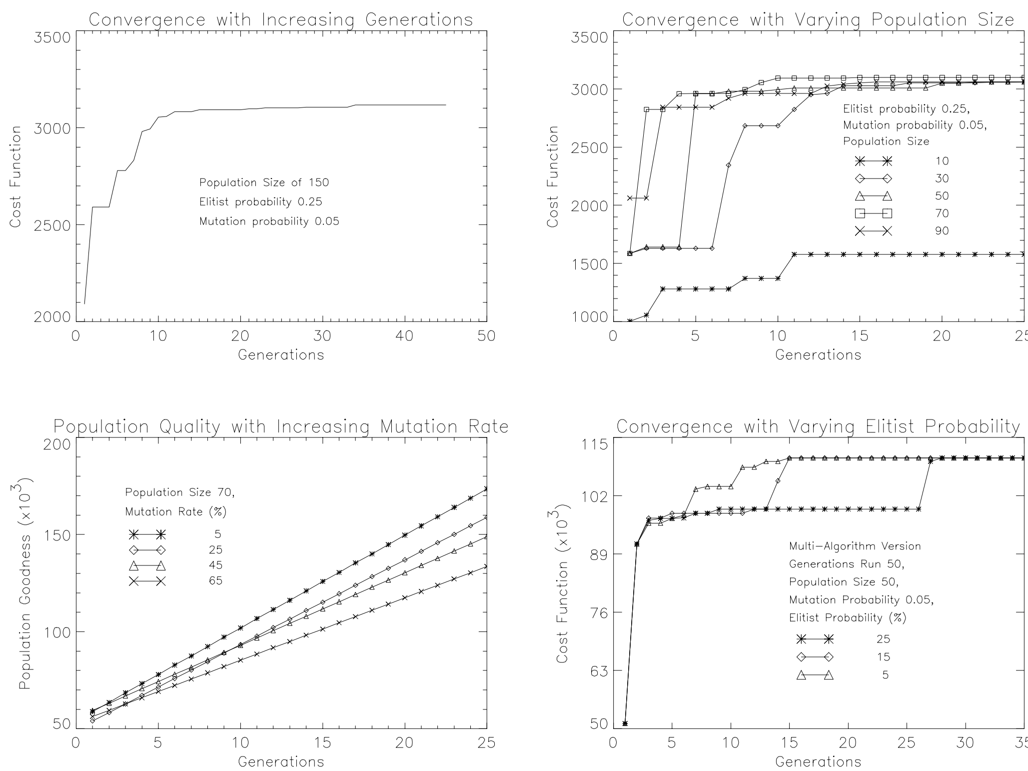


Figure 8; Genetic Algorithm Convergence - I

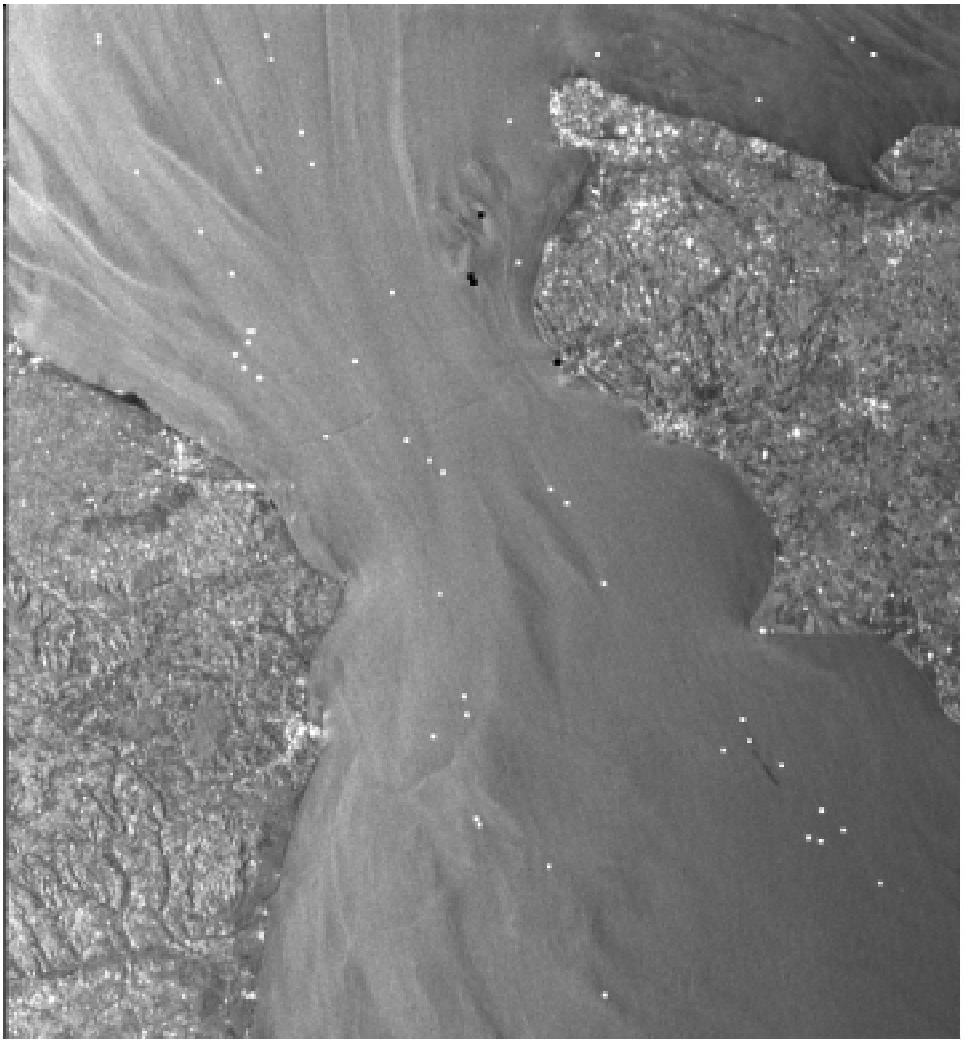



Figure 9; Output from the final neural network configuration.  True ship detections are marked with white squares and false alarms are marked with black squares.