

# Reducing False Alarms using Genetic Programming in Object Detection

Andrew Innes  
School of Aerospace, Mechanical  
and Manufacturing Engineering  
RMIT University  
Melbourne, Australia

John Mamutil  
Braces Pty Ltd  
404 Windsor Road  
NSW 2153, Australia

Vic Ciesielski  
School of Computer Science  
and Information Technology  
RMIT University  
Melbourne, Australia

Sabu John  
School of Aerospace, Mechanical  
and Manufacturing Engineering  
RMIT University  
Melbourne, Australia

**Abstract:** *This paper describes a refinement of an approach to locating objects in complex images with the objective of reducing the false alarm rate. The method uses genetic programming to evolve a detector using a threshold for defining an unclassified region. It is envisaged that incorporating a threshold during training will encourage programs to produce a high output when the object has been located. The objects required to be located are two difficult cephalometric landmarks that are either ambiguous in nature or located in a cluttered background. Results suggest that while increasing the threshold reduces the false alarm rate, it is to the detriment of the detection rate.*

**Keywords:** genetic programming, object detection, automatic cephalometric analysis.

## 1 Introduction

Object detection is a subset of the computer vision domain where a detector has to correctly classify and locate the position of an object. Correctly classifying objects is domain dependent where the cost of false positives and false negatives varies with application. For example, the criteria of classifying objects in medical applications is different to target recognition in military applications where undesirable false alarms result in the selection of wrong targets. The overall goal of our problem is to automatically locate

cephalometric landmarks in digital X-rays and minimise the number of false alarms. Ideally if the landmark can not be located then the landmark can be manually located by an 'expert'.

Our previous work [1, 2] of locating cephalometric landmarks on digital X-rays has used genetic programming to evolve programs using pixel level features. The pixel level features were calculated using mean and standard deviation of grayscale intensities for a set of shapes from within an input window. The region is searched by traversing an input window across the image and applying a detection program, in moving window fashion, across the X-ray and the output of the program is calculated. After the traversal, a decision is made about the location of the landmark based on the recorded position of the highest output. It is anticipated the feature set and function set will be able to evolve a detection program that can locate a specific landmark in the training set. This work has produced excellent results locating some of these landmarks, although harder landmarks have been less successful.

This is a result of landmarks located in either cluttered regions or being presented with ambiguous information. An example of ambiguity is finding the position of the upper lip that can also encompass the lower lip as shown in Figure 1. The z-axis of the wire mesh superimposed on the image in Figure 1 illustrates the output of a program at each pixel location. Using this approach

of detecting the upper lip, the image in Figure 1 would have incorrectly located the landmark in favor of the lower lip. The reason for this occurring is the lower lip exhibits similar characteristics to the upper lip. This is better depicted in View A of Figure 1 that shows a marginally higher output at the lower lip in comparison with the upper lip.

Part of the work presented by Teredesai et al. [3] focused on reducing false alarms by specifying either side of a decision boundary of two classes as uncertain (refer to Figure 2b). Results were not presented to determine how effective the indecisive region was for reducing false alarms. A difference between our detection strategy is that we have adopted highest output for locating the position of the landmark. So the aim of this paper is to determine if we can adopt a similar ideology of reducing false alarms for difficult landmarks. A similar method to Teredesai et al. for reducing false alarms is to not classify a landmark if the computed output at other pixel positions are similar to the highest output. This paper is to determine how false alarm rate is affected and whether it will significantly reduce detection rate. It is expected landmarks not classified will then be manually located by an expert.

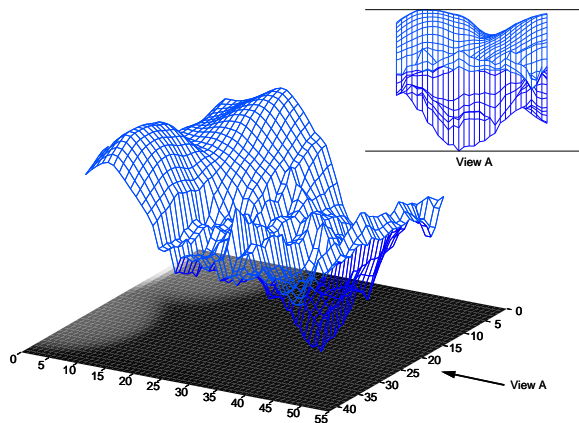


Figure 1: An ambiguous image that contains the desired object, upper lip, but also contains the lower lip that is similar in appearance. The surface plot is the output from a detector applied to the image. The highest output determines the position of the landmark. The image shows the highest output is located at the lower lip at position (11,31) and the region containing the second highest output is located at the upper lip at position (20,7). This is better shown in the side profile of View A.

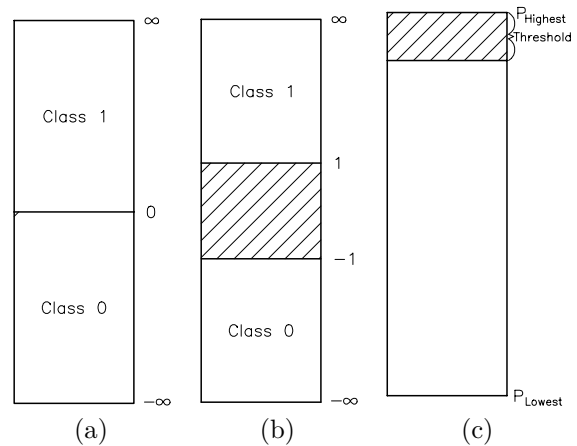


Figure 2: Schematic of decision boundaries for (a) binary classifier, (b) binary classifier incorporating an undefined region, and (c) landmark detector using the highest output incorporating an undefined region. The cross hatched region denotes the undefined region.

## 1.1 Previous work

A popular methodology for object detection using genetic programming (GP) has been to formulate the problem as object and non-object. This is the most simplistic classification model as the output of the program decides the class and often zero is chosen as the decision boundary. An example of this is shown in Figure 2a. An exception to this is work presented by Teredesai et al. [3] who proposed a method for training a detector for each digit of an on-line number recognition and an undefined region of  $[-1,1]$  indicating a detector was unable to make either a positive or negative decision (refer to Figure 2b).

In our problem we have used prior knowledge to extract a sub-image from a cephalometric X-ray that contains the relevant landmark. We also know that each sub-image will contain only one landmark. Rather than using the common methodology of choosing zero as the decision boundary, we use domain knowledge when passing the program over the sub-image and determine the location from the position at the highest output. Previous work has shown the highest output for locating a landmark significantly reduces the number of false alarms when compared to classifying as a two class problem.

Ideally what we want to achieve as a result of training is to produce a high output at the position of the landmark compared with the surrounding background. In reality if no pressure

is placed on the output by the fitness function, then this is unlikely to occur. The strategy developed will hopefully not classify a landmark when a position has produced a similar output to the predicted position.

## 1.2 Fitness

A problem of genetic programming is achieving a fitness function that is able to reflect the goals of the problem. This could be as simple as differentiating between class 1 and class 2. A fitness function applies pressure to a program to achieve specific criteria and rewards the performance of a program accordingly. A program is rewarded using a fitness measure and the population of programs are ranked based on fitness to determine the most successful programs. As part of the evolutionary process the fittest or most successful programs are copied into the next generation of individuals.

In the context of vision and image related applications, fitness is generally based on classifying objects and non-objects correctly. An objective of the fitness function maybe to differentiate between object/background, edge/no-edge, etc. If an object or non-object is incorrectly classified then this is a false negative and false positive respectively.

So why is specificity/sensitivity important? The importance of specificity/sensitivity is evident when more class 1 examples exist than class 2 during training. When the number of class 1 greatly outweighs class 2, as is often the case in object detection, it becomes highly likely that class 2 will be misclassified and the object will not be detected. Therefore when the background greatly outweighs the number of objects, a trade-off is required that will reward correct detection but at the same time sufficiently penalise false alarms.

A common fitness function dealing with the specificity/sensitivity is,  $A \times Fr + B \times (1 - Dr)$ , where the first and second term deals with false alarm rate and detection rate respectively. The constants  $A$  and  $B$  represent sensitivity/specificity and reducing the first term rewards detection rate or reducing the second term rewards false alarm rate. The task is to minimise fitness and should ideally reach zero, but this is not often the case when using real data from large datasets. So therefore a trade-off is required between sensitivity and specificity.

## 2 Methodology

The methodology outlined in this section is to determine if false alarm rate can be decreased by increasing threshold during training. For further information regarding general information to genetic programming and object detection refer to papers by [2, 1]

### 2.1 Terminal Set

In the context of object detection using genetic programming, terminals correspond to image features as outlined in Figure 3. In addition to these features, a random number in the range [0,255] was included.

### 2.2 Function Set

The function set (+, -, \*, /) consists of four common functions used in vision and image related applications. The +, - and \* have their usual meanings, while '/' is protected. A protected division has its usual meaning except a divide by zero produces INT\_MAX. Previous research has shown using other operators in the function set does not improve detection rate. Using a function set containing irrelevant functions can also lead to slower convergence.

### 2.3 The fitness function

A program used for the purpose of locating a landmark is evaluated at each pixel in the sub-image. The output of the program is a floating point number that determines the likelihood the evaluated position is the location of interest based on the highest output. The position recorded at the highest output is compared with the known location and the landmark is recorded as a true positive (correct), false alarm (incorrect) or unclassified. The program is subsequently applied to a training set of images and the fitness is calculated using Equation 2. This determines how effective the program was at locating a landmark for the training set where the objective is to minimise the fitness function. A description detailing each step for evaluating program fitness is given as follows:

1. A program is traversed across the sub-image at each pixel location and the program output,  $P_{ij}$ , at position  $(i, j)$  is evaluated. The predicted position of the detected landmark

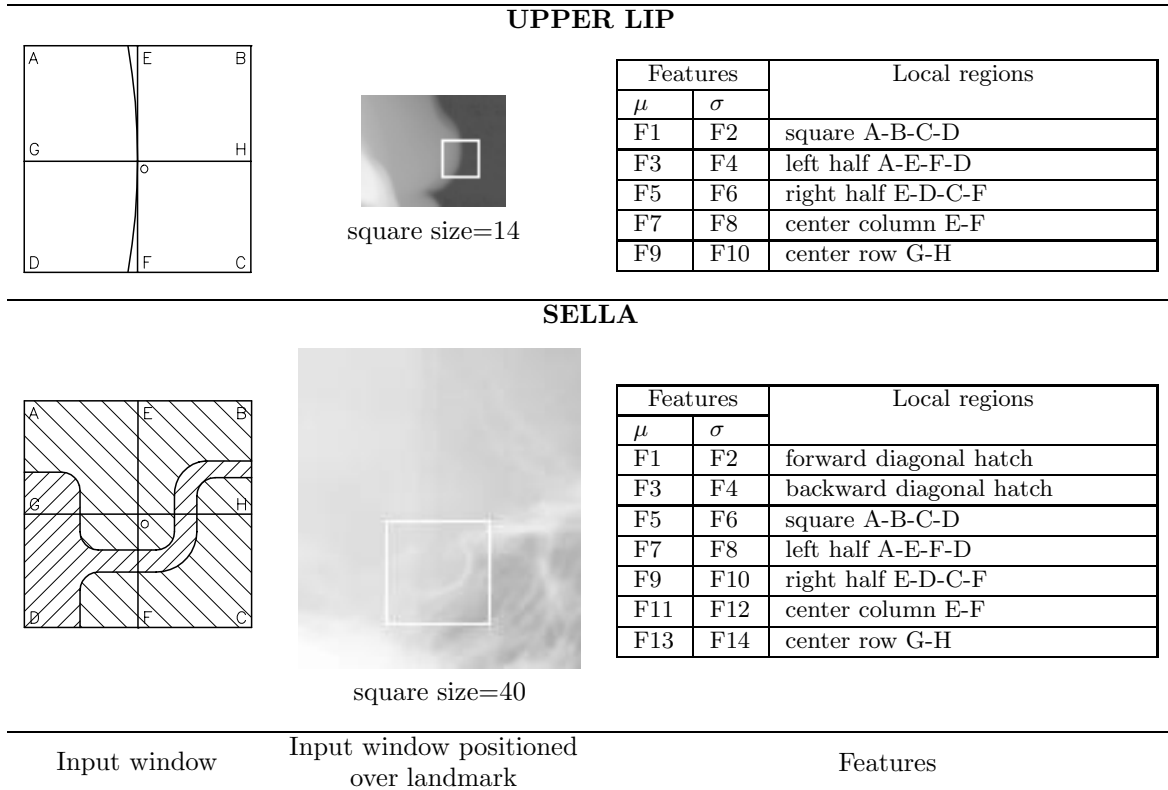


Figure 3: The diagrams in the left column depicts the input window used to extract the features for the upper lip and sella landmark. The features consist of the mean and standard deviation calculated for each shape from grey level intensities. The corresponding picture in the middle column depict the size of the input window (shown as the white square) relative to the sub-image. Note: The upper lip is contrast enhanced for clarity purposes only.

is recorded as the location corresponding to the highest output ( $P_{Highest}$ ).

2. A second traversal of the image verifies the output at each location in the image,  $P_{ij}$ , is not similar to the predicted position ( $P_{Highest}$ ). Each position is verified using Equation 1 and determines if a position has produced an output in the shaded region of Figure 2c. The size of the shaded region is pre-determined prior to training and represents the threshold that is a percentage between the lowest ( $P_{Lowest}$ ) and highest output ( $P_{Highest}$ ). For example a threshold of 10 requires the output at every position other than the predicted position to be at least 10% smaller than the highest output. If the criteria of Equation 1 has been fulfilled then the landmark for that image is recorded as being unclassified. Pixels located within a euclidean distance of 5 pixels (2mm) of the predicted position are not verified because they are within an acceptable error toler-

ance. It is also expected that an output at these pixel locations will most likely produce a similar output to the highest output.

$$\frac{P_{Highest} - P_{ij}}{P_{Highest} - P_{Lowest}} \times 100\% \leq Threshold \quad (1)$$

3. If the landmark in the image has not been unclassified, then a comparison is made with the predicted position and the known true location of the landmark. A match (true positive) occurs when the comparison is within a tolerance of 5 pixels or 2mm. If the comparison is not within the tolerance then the landmark for the respective image is recorded as a false alarm. The tolerance is an upper error limit that is deemed acceptable for a cephalometric analysis. The error is defined as euclidean distance between the position found by the 'system' and an 'expert'.
4. At the conclusion of evaluating the program for each image in the training set the detec-

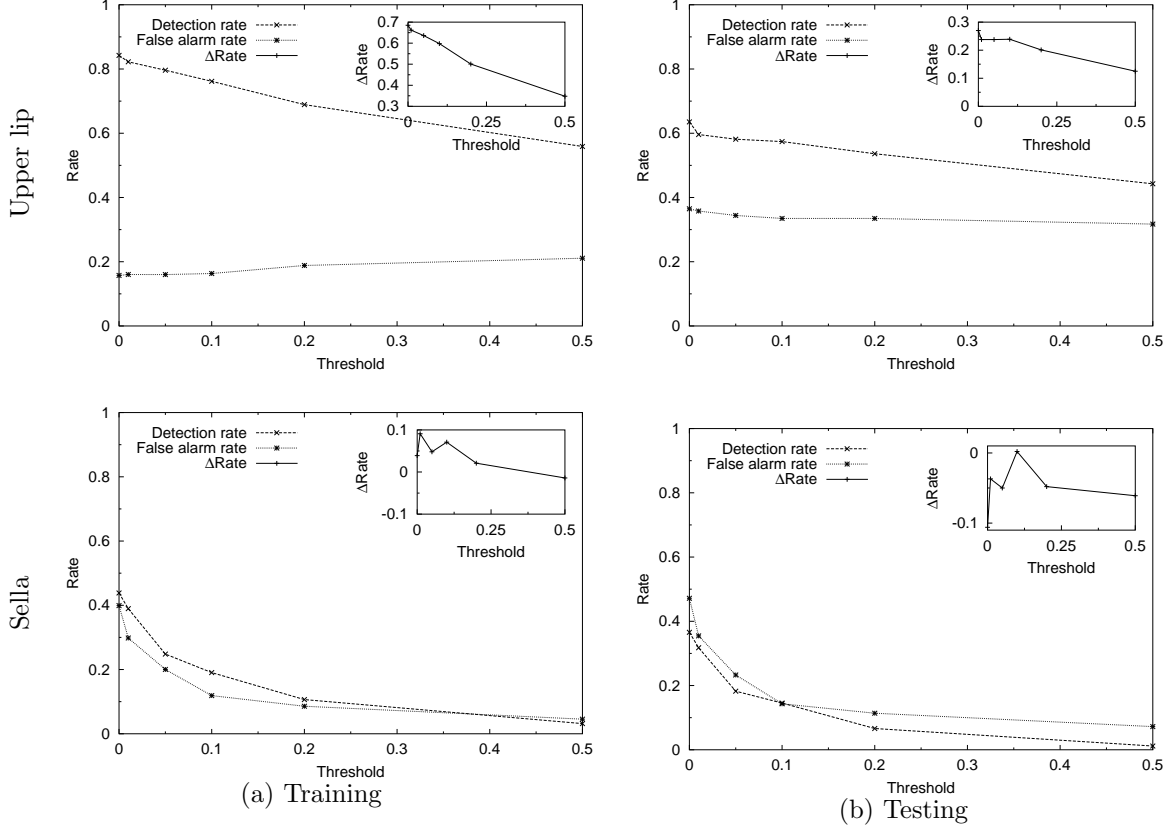


Figure 4: The graphs show how ‘threshold’ influences false alarm rate and detection rate. The top and bottom graphs are results for the upper lip and sella landmark respectively ( $N=80$ ). The graph inset is the difference between detection rate and false alarm rate,  $\Delta Rate$ , for each threshold.

tion rate,  $Dr$ , and false alarm rate,  $Fr$ , are calculated.

5. The fitness is computed as per Equation 2.

$$fitness = A \times Fr + B(1 - Dr) \quad (2)$$

where  $A$  and  $B$  are constants reflecting the relative importance of the false alarm rate and detection rate respectively.  $A$  and  $B$  are 200 and 1000 respectively or a ratio of 1:5. The fitness function defined in Equation 2 is constructed so as detection rate increases the fitness function approaches zero.

### 3 Results

To determine how threshold has effected false alarm rate, the method is tested on the upper lip and sella landmark. The upper lip landmark is a medium level of detectability based on the ambiguity between upper and lower lip. The sella

landmark is a hard landmark as it is located in a cluttered background. The data set consists of 110 images of which 83 images are randomly selected for training and the remaining images withheld for testing. Each parameter setting was conducted on 80 evolutionary runs.

$$\Delta rate = Dr - Fr \quad (3)$$

The graphs in Figure 4 show that incorporating a threshold during training results in programs having little effect on false alarm rate for detecting the upper lip, while the results for the sella landmarks show false alarm rate is significantly reduced and beginning to plateau at a threshold value of 0.1. However the smaller graph inset of each graph shows an undesirable trend that detection rate reduces at a faster rate than false alarm rate. This is described as  $\Delta rate$  and is defined in Equation 3. This indicates that while false alarm rate is reduced it is to the detriment of detection rate. In other words detection rate reduces at a

faster rate than false alarm rate.

## 4 Conclusion

A method was proposed for landmark detection using a genetic search strategy to encourage programs to produce a high output at the location of interest compared to background pixels. The emphasis was to reduce false alarms for difficult landmarks without significantly effecting detection rate. The work showed that increasing the threshold, or the difference between the highest output and surrounding outputs for a program, reduced false alarm rate but to the detriment of detection rate. So the method did not produce an outcome that was acceptable for our problem. Further work may possibly look at evolving a classifier to reduce background pixels prior to evolving a detector.

## Acknowledgments

This work is funded by the Australian Research Council (ARC) SPIRT grant scheme in partnership with Braces Pty Ltd from grant no. C00107119 and supported by grant EPPNRM054 from the Victorian Partnership for Advanced Computing.

## References

- [1] A. Innes, V. Ciesielski, J. Mamutil, S. John, and A. Harvey, "Landmark detection for cephalometric radiology images using genetic programming," in *Proceedings of the 6th Australia-Japan Joint Workshop on Intelligenet and Evolutionary Systems*, Ruhul Sarker, Bob McKay, Mitsuo Gen, and Akira Namatame, Eds., Canberra, Nov. 2002, pp. 125–132, <http://www.cs.rmit.edu.au/~vc/papers/aust-jap-ec02.pdf>.
- [2] Andrew Innes, Vic Ciesielski, John Mamutil, and Sabu John, "Landmark detection for cephalometric radiology images using pulse coupled neural networks," in *Proceedings of the 2002 International Conference on Artificial Intelligence and Workshop on Computational Models of Scientific Reasoning and Applications*, 2002, pp. 390–395, <http://www.cs.rmit.edu.au/~vc/papers/isai02.doc>.
- [3] A. Teredesai, E. Ratzlaff, J. Subrahmonia, and V. Govindaraju, "On-line digit recognition using off-line features," in *Indian Conference on Computer Vision, Graphics and Image Processing ICVGIP 2002*, 2002.