# IMAGE THRESHOLDING FOR LANDSLIDE DETECTION BY GENETIC PROGRAMMING

Paul L. Rosin
*Department of Computer Science*
*Cardiff University*
*UK*
*E-mail:* `Paul.Rosin@cs.cf.ac.uk`

Javier Hervás
*Institute for Systems, Informatics, and Safety*
*EC Joint Research Centre*
*Ispra, VA, Italy*
*E-mail:* `javier.hervas@jrc.it`

This paper describes an approach to image thresholding that combines various multiscale and morphological features, including texture, shape and edge filtering, by using genetic programming, to detect the presence of landslides and their active sectors in change detected multitemporal aerial images.

## 1 Introduction

Detecting landslides and monitoring their activity is of great relevance for natural hazard assessment and disaster prevention in hilly areas. Very high resolution optical satellite data is now becoming available, and to evaluate the potential application of these new images to detect ground surface changes as a result of landsliding, in previous work we developed and applied change detection and thresholding methods on digital aerial photographs over the Tessina landslide in Veneto, Italy[1]. This is a nearly 3-km long complex landslide consisting of rotational slides in its head and a mudflow in the remaining, largest part of the body. The landslide has developed in Eocene flysch materials partly covered by colluvial and glacial till sediments. The landslide was first triggered in 1960 and has since undergone a number of reactivations. The physical setting of the area and landslide dynamics are further described in[2,1].

The landslide currently affects an area including a number of small villages and mixed woodland-grassland landcover. It generally appears as a distinct bright feature on visible-wavelength panchromatic images because of outcropping soil under disrupted vegetation (figure 1a&b).

This paper focuses on applying genetic programming techniques to detect and monitor landslides from optical remote sensing data. To this end, we have used radiometrically mormalised and orthorectified multitemporal aerial photographs at 1m ground resolution over the Tessina landslide area[1].

## 2 Genetic Programming

Genetic programming (GP) is an optimisation technique based on the concepts of Darwinian evolution [3]. A population of individuals is created, each representing a potential solution to the problem. The solution offered by each individual is assigned a fitness value which indicates how well that solution performs. Over time the fitter individuals live and reproduce while the others tend to die without reproducing, simulating the process of survival of the fittest. With each generation the overall population fitness improves, thus performing optimisation.

Whereas genetic algorithms usually code solutions by fixed-length strings which results in their solution space being constrained, GP uses a tree structure. The leaves of the tree (called terminals) represent input variables or numerical constants. Internal nodes (called functions) represent arithmetic, mathematical, boolean, etc. functions. Individuals are evaluated by traversing through the tree, applying function nodes to their children, and so can be considered to represent programs. The search space of GP is the set of all the possible compositions of the functions in the union of the terminal and function sets.

The initial population is formed by constructing random trees. Thereafter, individuals are chosen from the population according to the fitness of their solutions. In a similar manner to genetic algorithms, genetic operators are applied to these trees. Many variations of the operators exist, but the two main ones are as follows. In *mutation* an individual receives a random alteration. Either the value or variable at a terminal is replaced, or an entire subtree is replaced. In *crossover* two random nodes are selected from within each of two parent individuals and then the resultant subtrees are swapped, generating two new child individuals which typically replace less fit members of the population.

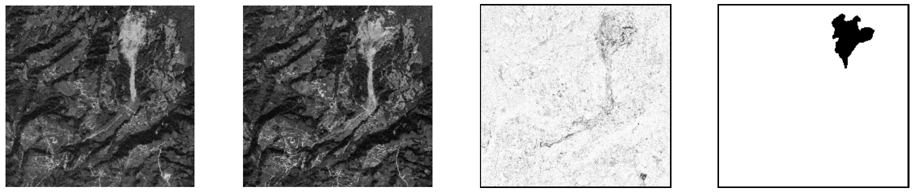
## 3 Automatic Thresholding

Our previous work on change detection considered a variety of algorithms, based on quite diverse principles, for determining appropriate thresholds for difference images [4]. Although some approaches were better than others, none were entirely satisfactory on their own as they tended to pick up spurious detail arising from residual effects of the radiometric normalisation. In an attempt to overcome this, a post-processing step was included to reduce the noise using shape properties of the thresholded regions. Area, width, perimeter, and rectangularity filters were applied and were shown to provide significant improvements.

In this paper an alternative approach is taken. Rather than manually design thresholding algorithms, one will be learnt using genetic programming. In this case the desired program is a binary classifier (although GP is more general purpose, and can also be applied to generate many other types of programs). That is, we wish to distinguish the “change” and “no-change” classes in the scene due to landsliding. Various schemes are possible. For instance, previously we have used GPs to generate classifiers made up from individual compositions of functions, one tree per class. These are then run in parallel and a winner takes all strategy used to select the class[5]. Here however, since only two classes are present a simpler strategy is sufficient and was found to work better. Programs are generated that take for each pixel a set of property values and compute a single real value. Pixels are assigned to the appropriate class according to whether their evaluation by the GP is a positive or negative value.

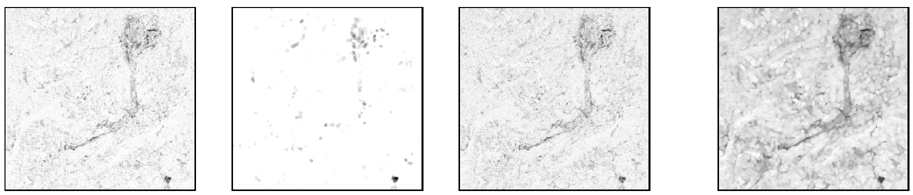
The function set was made up of a small set of standard operators: the arithmetic ones: *, +, -, /; as well as *absolute*, *sigmoid*, *minimum*, and *maximum*. The terminal set is more crucial and needs to be chosen with some care. In addition to random ephemeral constants and the difference image further input is required to enable a good classifier to be constructed. Whereas our previous work applied post-processing to the thresholded image it is desirable to retain the intensity information as long as possible before binarising. This is analogous to fuzzy reasoning where the data is only “crispened” at the end of processing chain. As already noted, much of the spurious detail was at a large scale, which suggests that providing coarser scale information could be helpful. To this end morphological opening and closing were applied over a range of scales to the difference image to remove the bright and dark details respectively. (See Stringa[6] for the use of mathematical morphology in change detection). Another technique for incorporating scale is to simply smooth the image, which again was applied over a range to provide a multiscale image representation. The spurious detail might show a distinct texture, and so some texture maps were included to enable their identification. Laws’[7] texture energy approach involving the convolution of masks for detecting spots, edges, lines, V’s, etc. was used.

Finally, a modified distance transform was applied to the inverted difference image. The standard distance transform is applied to a binary image and efficiently computes the distance at each pixel to the nearest feature pixel[8]. While straightforward for binary data, it is more difficult for grayscale data which generally needs to be thresholded, bringing us back to our thresholding problem. One approach has been suggested for incorporating intensity information for edge-like features[9]. An alternative was developed here for arbitrary grayscale images in which the image is thresholded at all possible levels. The distance transform is applied to this image set and the results combined by adding them together and rescaling. The effect is to incorporate some coarser spatial information while retaining graylevel information. This is done without arbitrarily thresholding the image, and also without distorting the image by blurring. Low values are produced in the distance transformed image as a function both of the intensity of the input image (i.e. the darker the inverted difference image, the darker the transformed image) and also of the proximity to dark



(a) 1988 (b) 1994 (c) difference image (d) ground truth

Figure 1: Images of Tessina before and after the landslide reactivation.



(a) $3 \times 3$ opening (b) $31 \times 31$ opening (c) $3 \times 3$ closing (d) $31 \times 31$ closing

Figure 2: Mathematical morphology operations applied to difference image

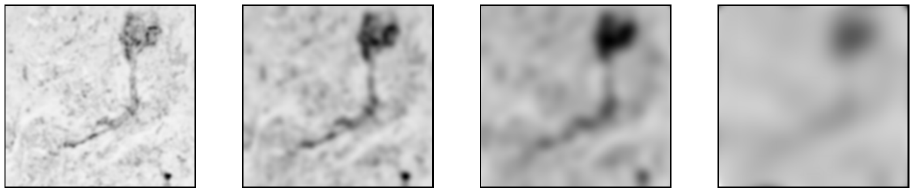
pixels (i.e. the closer they are, the darker the transformed image).

## 4 Results

Figure 1a&b shows orthorectified and radiometrically normalised images of the Tessina area covering the period before and after the most recent major landslide movement in 1992. The landslide runs from near the top right corner to the centre bottom left of the images; the thinner sector represents the mudflow. After some cropping their size is $2703 \times 2590$. The difference image (figure 1c; black indicates high change, white low change) is calculated by taking the absolute difference of the pixel intensity values over the two dates.

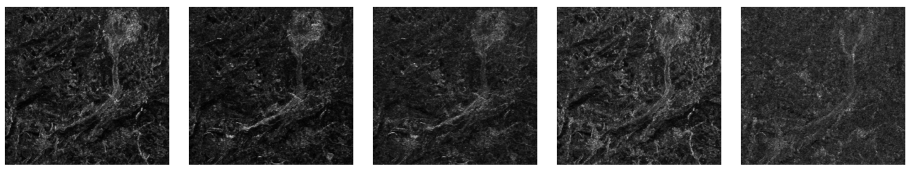
Various categories of change can be specified. We have taken the head scarp and upper accumulation area of the landslide as ground truth (figure 1d) as it gives a reasonably substantial set of training data for both classes (change and no-change), but future work will look at the smaller reactivation area as defined in[2,1]. The selected area of change makes up about 3% of the image, while the reactivation area is only 0.3%. Figures 2–5 show the various features extracted from the difference image. For reference, the result that we previously obtained using the corner thresholding method[4] is given in figure 6.

As only one image is available it has to be used both for training and testing.



(a) $\sigma = 5$ (b) $\sigma = 10$ (c) $\sigma = 20$ (d) $\sigma = 40$

Figure 3: Smoothed version of the difference image



(a) horiz. edge (b) vert. edge (c) horiz. line (d) vert. line (e) spot

Figure 4: Texture energies of the difference image

About 2700 training pixels were sampled from the image (i.e. about 4% of the data) for training the classifier which is then tested by applying it to the full image. For some classifiers, applying an even sampling of the image such that the number of class samples is proportional to the size of the class builds into the classifier expectations of the class likelihoods. This is the case for the classifier in this paper as the fitness function is the percentage of correct classification. While it is desirable to use these expectations it was found to sometimes lead to problems due both to the larger no-change class dominating the optimisation criterion, and also because of the small number of change class training pixels.

The GP was run with a population of 20,000 trees which were allowed to grow to a maximum depth of 7. For each experiment 10 runs with different random initialisations were made, with each population undergoing about 200-300 generations. Substantial computing power is required for such development as each run takes about 4 days to complete on a Sun Ultra-5 workstation. The following figures (7-12) show the results using either representational sampling (left image) or quasi-equal sampling (right image).

The problem with a large variation in class sizes is demonstrated when the classifier is trained with just the difference image (figure 7). When representational sampling is used it is found that the best classification is obtained by simply blanking the image. Although the change class is almost completely wrongly classified the

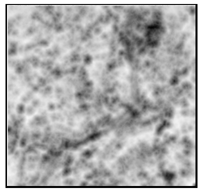


Figure 5: Summed distance transform over all thresholds

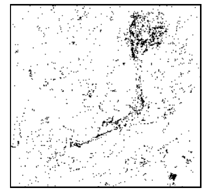


Figure 6: Thresholding the difference image using the corner method

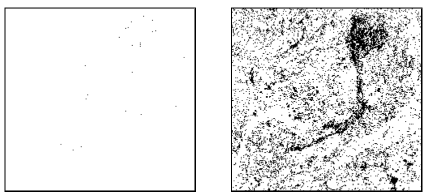


Figure 7: Training using the difference image alone

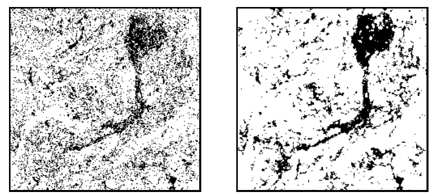


Figure 8: Training using the difference image plus opening and closing

remaining 97% of the image is correct, and so the GP solution obtained a high fitness score. Taking equal (or at least roughly equal) numbers of samples avoids this trivial solution, although there is not enough information available to the GP system to generate a solution better than a standard thresholding method.

Incorporating the data from the filtered images produces a noticeable improvement. Combining the difference image with its opened and closed versions enables the coarse structure of the landslide to be identified even though much residual noise is still picked up (figure 8). Adding textural information to the difference image gave extremely poor results (figure 9), and its addition to the mathematical morphology filtering also provides no benefit (figure 10).

The smoothed images provide much coarser scale spatial information than the opened and closed images (figure 11). This is evident when they augment the difference image as only a few large-scale blobs are extracted, and the majority of noise has been eliminated. Including opening and closing and the distance transformed image provides some further improvement so that the result in figure 12 is getting close in size and shape to the ground truth.

The fittest individual found with representational sampling that was executed to produce figure 12a is shown below (written as a prefix expression) where the terminals `s5`, etc. are the difference image smoothed by $\sigma = 5$ etc., `o15` is the image opened with a $15 \times 15$ mask, and `dt` is the distance transformed image.

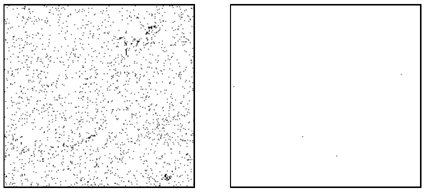


Figure 9: Training using the difference image plus texture

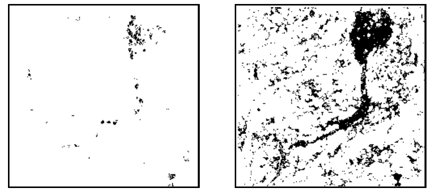


Figure 10: Training using the difference image plus opening, closing, and texture

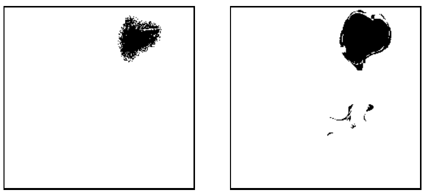


Figure 11: Training using the difference image plus smoothing

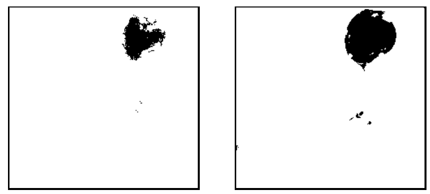


Figure 12: Training using the difference image plus opening, closing, and smoothing, and distance transform

```
(+ (minimum (minimum (- (- s10
                   (* difference
                      (+ s20 s20))) dt)
             (maximum (maximum dt
                       (- s40 dt))
                  (* (- s10 dt) s40)))
        (minimum (* (- o15 s10)
                (sigmoid (* difference
                        (+ s20 s20))))
             (maximum (minimum (+ -0.74079 s20)
                       (- (absolute difference)
                          (* s40 s10)))
                  (- (* s10 o15) s20))))
   (- (* (+ s20 s10)
         (* s40 s10))
      (maximum (- (* (absolute difference)
                 (* (+ s20 s10)
                    (* s40 s10)))
              (maximum (* (* s10 o15)
                      (- s40 dt))
                   (- s40 s10)))
           (maximum (* (- s10 dt) s40)
                (maximum (- (absolute difference)
                        (* s40 s10))
                     (- s40 dt))))))
```

## 5 Conclusions

Our experiments showed that simple per pixel thresholding was not entirely satisfactory on its own in providing a good segmentation of a difference image (obtained from time-separated acquisitions) for detecting landslide occurrence and/or reactivation. Utilising coarser scale information as well as texture, shape, and edge filtering provided a set of additional features. These were combined using genetic programming, and formed a new classifier that provided better identification of active sectors within a landslide body.


## 6 Acknowledgements

Program development was done using the lil-gp genetic programming system developed at Michigan University[10].


## References


1. J. Hervás, J.I. Barredo, P.L. Rosin, A. Pasuto, F. Mantovani, and S. Silvano. Monitoring landslides from optical remotely sensed imagery: the case history of Tessina landslide, Italy. *Geomorphology*, in press.
2. F. Mantovani, A. Pasuto, S. Silvano, and A. Zannoni. Collecting data to define future hazard scenarios of the tessina landslide. *Int. J. of Applied Earth Observation and Geoinformation*, 2(1):33–40, 2000.
3. J.R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.
4. P.L. Rosin, J. Hervás, and J.I. Barredo. Remote sensing image thresholding for landslide motion detection. In *1st Int. Workshop on Pattern Recognition Techniques in Remote Sensing*, pages 10–17, 2000.
5. P.L. Rosin and H.O. Nyongesa. Combining evolutionary, connectionist, and fuzzy classification algorithms for shape analysis. In S. Cagnoni *et al.*, editor, *Real-World Applications of Evolutionary Computing*, pages 87–96. Springer, 2000.
6. E. Stringa. Morphological change detection algorithms for surveillance applications. In *British Machine Vision Conf.*, pages 402–411, 2000.
7. K.I. Laws. Rapid texture identification. In *Proc. SPIE Image Processing for Missile Guidance*, pages 376–380, 1980.
8. G. Borgefors. Distance transformations in digital images. *Computer Vision, Graphics and Image Processing*, 34(3):344–371, 1986.
9. P.L. Rosin and G.A.W. West. Salience distance transforms. *CVGIP: Graphical Models and Image Processing*, 57:483–521, 1995.
10. D. Zongker and B. Punch. *http://isl.cps.msu.edu/GA/software/lil-gp*. 1996.