

DESIGN OF MORPHOLOGICAL FILTERS USING GENETIC ALGORITHMS

Stephen Marshall

Dept of Electronic and Elect Eng,
University of Strathclyde
204 George Street, Glasgow, UK
Tel +44 141 548 2199
fax +44 870 163 9219
e-mail: s.marshall@eee.strath.ac.uk

Neal R Harvey

NIS-2, MS D-436
Los Alamos National Laboratory
Los Alamos, NM 87545, USA
Tel +1 505 667 9077
fax +1 505 665 4414
e-mail: harve@lanl.gov

David Greenhalgh

Dept of Statistics and Modelling Sci,
University of Strathclyde
26 Richmond Street, Glasgow, UK
Tel +44 141 548 3653
fax +44 141 552 2079
e-mail: david@stams.strath.ac.uk

ABSTRACT

The design of optimum morphological and other non linear filters based in set, rank and logic is a non trivial task. The search space of possible solutions grows very rapidly with the number of filter parameters such as window size and signal bit depth. One of the simplest methods to estimate optimum parameters is to use the iterative search technique known as genetic algorithms.

This paper will describe a new tighter bound for the convergence of genetic algorithms. It will also present an approach to the design of morphological filters using genetic algorithms. It uses iterative search techniques to probe the solution space in a way which models evolution in nature.

Both practical and theoretical results achieved in this area will be included:

1. A new improved upper bound, which has recently been derived by the authors, gives the number of GA generations required to guarantee (with a pre-specified certainty) that the optimal solution has been located. Unlike the previously accepted bound it reduces with increasing population size.
2. Results using soft morphological filters designed by genetic algorithms in film restoration will be shown.

The GA method is simple in its approach and bypasses the need for highly complex models of the process. As well as providing an interesting perspective to the non linear design debate, solutions found in this way may prove to be complementary to more analytical approaches by both confirming and even prompting new solutions by these routes.

1 INTRODUCTION

In this paper the convergence properties for genetic algorithms are first discussed. By looking at the effect of mutation on convergence we show that by running the genetic algorithm for a sufficiently long time we can guarantee convergence to a global optimum with any specified level of confidence. An upper bound is obtained

for the number of iterations required. This improves upon previous results. The upper bound decreases as the population size increases. It can also be shown that this upper bound is asymptotically optimal for large population sizes.

In the second part of the paper a genetic algorithm is used to design an optimum 4D greyscale morphological filter for use in film sequence restoration.

2 BACKGROUND

Genetic algorithms are robust search mechanisms based on underlying genetic biological principles. Having been established as a valid approach to problems requiring efficient and effective search, genetic algorithms are increasingly finding widespread application in business, scientific and engineering circles [1]. These algorithms are computationally simple yet powerful in their search for improvement. They work on a wide range of discrete search spaces and are very versatile as few assumptions are needed about these spaces. However, because of this versatility, relatively little mathematical theory is available to quantify the performance of these algorithms. Genetic algorithms (GAs) comprise three basic mechanisms: reproduction, crossover and mutation. Typically the discrete space to be searched is coded as a set of binary strings of length γ . These strings are analogous to chromosomes. An initial population of n strings is chosen. For simplicity n is taken to be even. The goal of the genetic algorithm is to find the maximum of some objective function f (called the fitness function) defined on the search space.

A simple genetic algorithm is described by Goldberg [1]. Starting with the original population a new population of n strings is selected in three stages. Firstly a new set of n strings (not necessarily distinct) is chosen from the original n strings by selecting each member of the population with probability proportional to its fitness. This is called reproduction. Secondly the new set of strings is mated (paired) at random. For each pair crossover occurs with probability χ . If crossover occurs then a position between 1 and $\gamma-1$ is chosen at random to divide the string. The first half of the first string in the pair is matched with the second half of the second string and the second half of the first string is matched with the first half of the second, to give a new pair of child strings.

Thirdly each bit of each string in the new population is independently flipped (changed from zero to one or vice-versa) with probability μ . This is called uniform mutation. Usually the mutation probability μ is small. A typical value for μ is less than 0.05. More complex genetic algorithms may have non-uniform crossover rates between iterations, or non-uniform mutation rates between iterations, or both. Other possibilities include more complex crossover schemes (for example multiple point crossover) and mutation schemes (for example mutation of blocks of digits altogether simultaneously).

The process of reproduction probabilistically ensures that individuals of high fitness are preferentially maintained in the population, whilst crossover and mutation are necessary to introduce the diversity needed to ensure that the entire sample space is reachable and to avoid becoming stuck at suboptimal solutions. In fact mutation on its own is sufficient to introduce such diversity. The idea is that after a large number of iterations the population should consist mainly of individuals very close to the optimal solution. It is possible, however, that the optimal string may have been present in an earlier population and then may have subsequently disappeared. To address this problem, a record of the string with the maximal fitness value seen so far is maintained from generation to generation. The optimal string may therefore be present in the final population or may have occurred in an earlier one.

The nature of the problem means that genetic algorithms are difficult to model. Attempts have been made to determine an upper bound on the number of iterations required to guarantee that the optimal string has been seen with some fixed probability. The existing bound was given by Aytug and Koehler [2] and is defined as

$$t_1(d) = \text{INT} \left[\frac{\ln(1-d)}{\ln[1 - \min\{(1-\mu)^{\gamma n}, \mu^n\}]} \right]. \quad (1)$$

Here $\text{INT}[x]$ = the smallest integer greater than or equal to x , γ =string length in bits, δ =certainty of having seen the optimum string, μ =mutation rate, n =population size.

Assuming that μ is small and so $\mu^{\gamma n}$ is very small, therefore $\ln(1 - \mu^{\gamma n}) \approx -\mu^{\gamma n}$ and $\text{INT}[y] \approx y$ if y is large gives,

$$t_1(d) \approx -\frac{\ln(1-d)}{\mu^{\gamma n}}. \quad (2)$$

The value of the bound is very large for any practical problem. Contrary to intuition this bound increases rapidly with population size.

3 NEW BOUND FOR GA CONVERGENCE

The assumptions used in deriving the bound in equation (1) lead to massive overkill. They are based on the premise that in order to have seen the optimal string with

probability, δ , it is necessary to see δ of all possible *states* of the population. In practice however it is only necessary to see δ of all possible *strings*. Based on this premise the authors have derived a new upper bound as,

$$\tilde{t}_1(d) = \text{INT} \left[\frac{\ln(1-d)}{n \ln[1 - \min\{(1-\mu)^{\gamma-1}, \mu, \mu^{\gamma}\}]} \right]. \quad (3)$$

Applying similar assumptions as in equation (2) this can be approximated as

$$\tilde{t}_1(d) \approx -\frac{\ln(1-d)}{n\mu^{\gamma}}. \quad (4)$$

This bound falls with increasing population size. It can be easily shown that the new bound is tighter than the existing one i.e.

$$\tilde{t}_1(d) \leq t_1(d). \quad (5)$$

with equality occurring when $n=1$. The approximate ratio of the bounds is given as,

$$\frac{t_1(d)}{\tilde{t}_1(d)} \approx n\mu^{-\gamma(n-1)}. \quad (6)$$

Inserting typical values an example with string length, $\gamma = 10$, mutation rate, $\mu = 0.2$, population size, $n=15$ and certainty $\delta = 0.9$. This gives the new bound as 1.5×10^6 iterations whilst the old bound is approximately 1.6×10^{105} iterations. If we double the population size to $n = 30$ the new bound reduces by half to 7.5×10^5 and the old bound almost squares to 1.1×10^{210} .

The bound may be extended to non binary alphabets and for an alphabet of cardinality K , is given as

$$\tilde{t}_1(d) = \text{INT} \left[\frac{\ln(1-d)}{n \ln \left[1 - \min \left\{ (1-\mu)^{\gamma-1} \left(\frac{\mu}{K-1} \right) \left(\frac{\mu}{K-1} \right)^{\gamma} \right\} \right]} \right]. \quad (7)$$

Further extensions to the theory, such as modifications involving non uniform mutation rates, have been developed and these may be found in [3].

Note that the bound is based on the analysis of mutation only. In practice the optimum would be reached much earlier due to the effects of crossover and the bias in reproduction towards the better performing strings.

4 USE OF GAs TO DESIGN SOFT MORPHOLOGICAL FILTERS FOR FILM RESTORATION

Soft morphological filters are powerful set based image operators which can remove noise from images whilst preserving structure. However their design methods can

be complex. In this paper we use a genetic algorithm to optimise a soft morphological filter which has been applied to an old film sequence in order to remove dirt. Film dirt consists of small particles which become trapped in the film transport mechanism damaging the film and causing loss of information. This damage manifests itself as “blotches” of random size, shape and intensity. These blotches are non-time correlated (temporally impulsive). An example of a single dirt corrupted frame is given in Figure 1.

A genetic algorithm is used to optimise a 4 dimensional soft morphological filter. The structuring system is 5x5x3 pixels. It consists of a hard centre and soft surround and is depicted in Figure 2. As a result of space limitations only a brief description can be given here but the interested reader is referred to [4,5,6,7] for more details.

Soft morphological filters combine the spatial properties of morphological filters with rank order properties. They do this by splitting their structuring element into two parts, a hard centre and a soft surround. Soft morphological filters are less harsh than standard morphological filters and are therefore less likely to damage image structure.

The design of a soft morphological filter requires the determination of four sets of parameters:

- Size and shape of structuring system's hard centre
- Size and shape of structuring system's soft boundary
- Repetition parameter
- Choice of soft morphological operations

The four parameter sets are represented in terms of a number of finite length words. These words are concatenated to form the strings which comprise the data set of the genetic algorithm. Each string carries sufficient information to define a soft morphological filter and all of its parameters. Every string has associated with it a ‘fitness’ value which is a metric describing how well the filter performs in the restoration task. Typically this metric would correspond to the Mean Absolute Error (MAE) which exists between the filtered version of a noisy image and its uncorrupted original.

A random population of strings is submitted to the genetic algorithm and the process of crossover and mutation proceeds. The GA is run for a few hundred generations until it converges and the best filter found up to that point is considered to be the optimal soft morphological filter.

This implies that the design stage has available to it an uncorrupted version of the image which it is attempting to restore i.e. a training set. Usually no ideal version of the original exists. In this case a training set is artificially created by selecting a small relatively noise-free area of the image to serve as the original. The noisy version is then created by pasting in typical noisy ‘blotches’

extracted from other parts of the sequence. Where the sequence to be restored contains rapid motion then it is important that the training set also has rapid motion otherwise the resulting filters will be unable to cope with it.

Alternative methods have been tested which avoid the use of a training set by maximising an image quality measure. These perform moderately well but the filters derived are not as effective as those arising from the training set approach.

5 PRACTICAL RESULTS

A soft morphological filter was trained using the training set shown in Figure 3. The hard centre consisted of only a single pixel at the origin with a soft surround. The entire film sequence was filtered using these parameters. Noise particles were removed from the sequence whilst fine detail and rapid motion were preserved. Figure 4 shows the frame of Figure 1 after filtering. To appreciate the significance of the results it is necessary to view a moving sequence at a larger scale. This may be carried out by going to <http://www.eee.strath.ac.uk/restoration>.

6 CONCLUSIONS

A new upper bound, which has recently been derived by the authors, gives the number of GA iterations required to guarantee (with a pre-specified certainty) that the optimal solution has been located. Unlike the previously accepted bound it reduces with increasing population size.

A method has been shown for the optimisation of grey-scale soft morphological filters for application in the specific area of film-dirt removal from archive film material. The technique is capable of finding grey-scale soft morphological filters which have excellent performance in the removal of film-dirt corruption, whilst retaining the necessary fine details within the original images. An example of applying the technique to real, corrupted image sequences have been given.

7 ACKNOWLEDGMENTS

The authors would like to thank the staff at BBC R & D Dept., Kingswood Warren, Surrey, England, for providing archive film material and to the EPSRC for funding part of this work.

8 REFERENCES

1. D.E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison Wesley, 1989.
2. H. Aytug, G.J Koehler, “Stopping criteria for finite length genetic algorithms”. *ORSA J. Computing*, **8**, 183-191, 1996.
3. D. Greenhalgh, S. Marshall, “Improved stopping criteria for Genetic Algorithms”, to appear *SIAM J. Computing*, early 2000, details not available at time of going to press.

4. N.R. Harvey, S. Marshall, "Restoration of Archive Film Material Using Multi-Dimensional Soft Morphological Filters", Proceedings of the IEEE Workshop on Nonlinear Signal and Image Processing, NSIP'99, Turkey, 811-814, June 1999.
5. N.R. Harvey, S. Marshall, "GA Optimisation of Spatio-Temporal Grey Scale Soft Morphological Filters with Applications in Archive Film Restoration", First European Workshop in Evolutionary Image Analysis, Signal Processing and Telecommunications, EvoIASP'99, Gothenburg, pp 31-45, May 1999.
6. N.R. Harvey, "New Techniques for the Design of Morphological Filters using Genetic Algorithms", Ph.D. Thesis, Dept. of Electronic and Electrical Engineering, University of Strathclyde, UK, 1997.
7. N.R. Harvey, S. Marshall, "Grey Scale Soft Morphological Filter Optimisation by Genetic Algorithms", in P. Maragos, P. Schafer, R. Butt (eds.), Mathematical Morphology and its Applications to Image and Signal Processing", Kluwer Academic Publishers, 179-186, 1996.

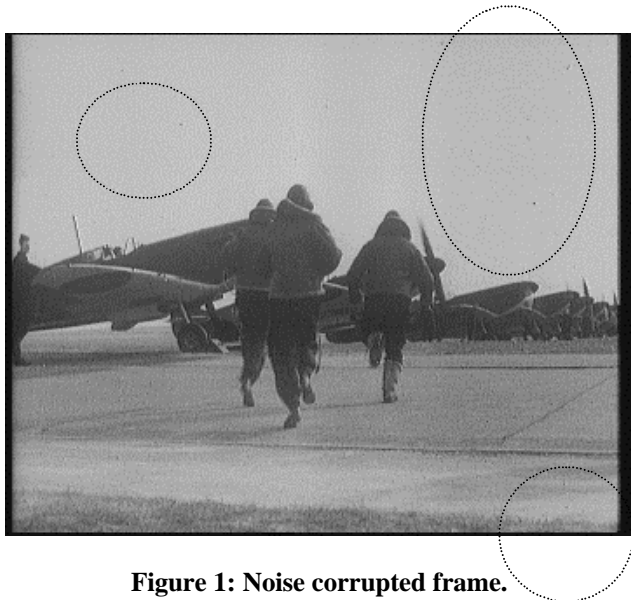


Figure 1: Noise corrupted frame.

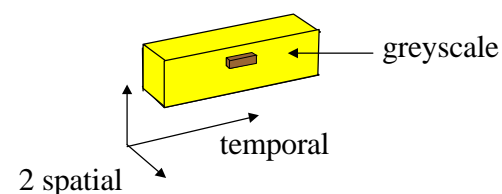
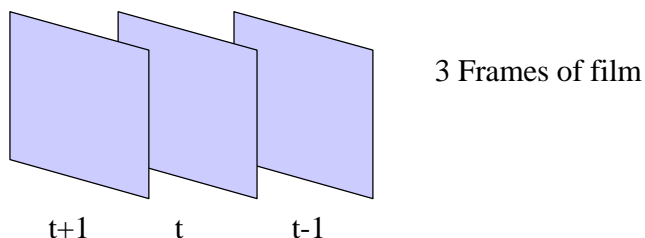


Figure 2: Four dimensional structuring system.



Figure 3(a):Ideal images of training set.



Figure 3(b):Corrupted images of training set.



Figure 4: Filtered frame.