

Optimal histogram partitioning using a simulated annealing technique.

R. Brunelli

Istituto per la Ricerca Scientifica e Tecnologica
I-38050 Povo, Trento, ITALY

Abstract—A method for histogram partitioning based on the maximization of a criterium function is proposed. The choice of the criterium can be made using a priori information to mimic human performance as far as possible.

Keywords. Image processing, simulated annealing, automatic threshold selection, image segmentation, histogram analysis.

1. INTRODUCTION

The analysis of one dimensional data is a common task in image processing. Image segmentation through slicing of the histogram of intensity level frequencies or through the analysis of projections is a familiar situation to everyone working in computer vision. The qualitative analysis of this type of data is only deceptively simple, and reproduction of human performance proves to be hard. The key point seems to be the ability to pick up a good partition out of a very large number of possible candidates. The number of possible partitions is usually exponential in the number of characteristic points (local extrema in the data). The criterium function (a formal definition of *goodness*) to be maximized is defined on a discrete space (all possible partitions identified by the characteristic points) and the problem becomes one of combinatorial optimization. Smoothing techniques can be used to simplify data analysis by reducing the search space but even if tunable (see for example [7] for adaptive smoothing) they are difficult to control in order to reproduce human performance.

A different strategy can be adopted when the modality of data is known. The distribution of values can then be fitted by the superposition of an appropriate number of functions, which are usually taken to be gaussian ([8]). The same approach is used in the more general field of functional approximation, where the use of radial functions - of which gaussians are but a single example - is widespread ([10]). Snyder et al. ([9]) have successfully applied tree annealing, a generalization of simulated annealing, to the least-square fit of multimodal histograms by a linear combination of gaussian functions.

The approach proposed in this note, while also based on simulated annealing, moves from a different strategy. The main point is the introduction of a *goodness* criterion characterizing the *shape* of a histogram slice (the data between any two local minima). The histogram is then partitioned so as to maximize the sum of its slice scores. No commitment is made on the number of slices, even if a-priori knowledge can be easily incorporated, as will be shown.

After a brief review of simulated annealing techniques, the segmentation task is formally described with the introduction of a criterium function and the algorithm for simulated annealing is presented. Some results are reported.

2. SIMULATED ANNEALING

A large class of problems require the minimization (maximization) of a function. More formally, let $F(\bar{x})$ be a function where \bar{x} represents a set of variables (parameters). We seek the

value \bar{x}_0 that minimizes $F(\bar{x})$:

$$F(\bar{x}_0) \leq F(\bar{x}), \forall \bar{x} \in X \quad (1)$$

where X represents the space of possible values for \bar{x} (configurations). Sometimes \bar{x} varies continuously: analytic techniques, such as gradient descent, can be employed (but see also [9] where the technique of tree annealing is applied to continuous minimization). Often \bar{x} can assume only discrete values: the problem becomes one of combinatorial optimization. An example is given by the Ising model: a lattice of magnetic dipoles with only two possible (anti-parallel) orientations. The state of the system is represented by a vector whose dimension equals the number of dipoles and whose components only assume the values $-1, 1$. The most used heuristics in the solution of optimization problems is that of *iterative refinement*: a possible solution is initially guessed and the space of neighbor configurations is explored looking for configuration at lower energy. This strategy (like gradient descent) is severely afflicted by the problem of local minima. Simulated annealing ([1], [3], [4]) owes its name to the process of annealing, by which a solid can be driven into a low energy state (usually a crystal lattice). The solid is initially heated to a temperature where many atomic rearrangements are possible. A cooling schedule is then activated, progressively reducing the possibility of atomic rearrangements: the solid freezes into a low energy state. Cooling must be very slow, allowing the system to attain thermal equilibrium at each temperature. An analogy can be established through the low energy state of a configuration of atoms and the minimum cost configuration \bar{x}_0 . If we can introduce a temperature, the analogy can be transformed into a procedure for combinatorial optimization. The main difference between simulated annealing and iterative refinement is that the former allows the system to visit configurations whose energy is higher than that of the actual configuration. The transition from configuration \bar{x} of energy E_0 to a configuration \bar{x}' with energy E_1 is always accepted if $E_1 \leq E_0$. If $E_1 > E_0$, a random number η , uniformly distributed in the interval $[0, 1]$, is generated and the new transition is accepted if $\eta \leq e^{-(E_1 - E_0)/T}$. To apply simulated annealing we must define:

- the form of the energy function;
- the configuration space of the system;
- the transformations between configurations (generation mechanism);
- the analogue of temperature;
- a cooling schedule.

At each temperature the system tries a given number of configurations and accepts a transition using the Boltzmann rule. The algorithm stops when the system is frozen: the temperature is so low that the system cannot visit any higher energy state and there are no lower energy states to jump to with the allowed transformations (a review of necessary and sufficient conditions for asymptotic convergence can be found in [1], [2]).

The histogram of an intensity image is defined as the distribution of the frequencies of the intensity values ([6]). To such a frequency distribution we can associate a list of nodes, each node representing a local minimum (valley) and the right adjacent local maximum (peak). Each node carries some information such as:

- histogram value at the valley and peak,
- value of the cumulative distribution at both points,
- active/inactive flag.

The configuration space we consider is given by all possible configurations of the flags. The number N of possible configurations is given by

$$N = 2^n \quad (2)$$

where n represents the number of valleys in the given histogram. The number of configurations is then exponential in the number of input data. A generation mechanism defines a neighborhood δ for each configuration c , consisting of all configurations which can be reached from c in one transition. In the reported implementation a new configuration is generated by flipping a random number of node flags with equal probability: the flag value is changed from active to inactive or viceversa, thereby deactivating (activating) the corresponding valley as a slice limit. Let δ_n denote the neighborhood obtained by flipping η node flags with η an integer random variable uniformly distributed in $[1, n]$. For each configuration we need an energy function $E(c)$ which represents the cost we want to minimize. This corresponds to the maximization of $-E(c)$ which can be interpreted as the *goodness* of the configuration. The energy function used in the reported experiments has the following form:

$$E(c) = - \sum_s \frac{\min(|p_s - l_s|, |p_s - r_s|)}{p_s} \left(1 - \frac{A_s}{p_s w_s} \right) \quad (3)$$

where the sum is extended over the slices s of the current configuration c , A_s represents the area of the slice, w_s its width, p_s the maximum histogram value in the slice, and l_s and r_s the values at the left and right delimiting valleys respectively (for a similar function see [5]). The first factor of each term in the sum reflects how deep the uppermost of the two delimiting valleys is, relative to the peak height; the second term takes into account how deep the other valley is (see Fig. 1). Other functions can be written depending on the particular slicing needs.

Suppose the number of peaks, say m , is known. This *a-priori* knowledge can be easily incorporated into the definition of the energy function:

$$E'(c) = \frac{E(c)}{\alpha |n_s - m| + 1} \quad (4)$$

where n_s is the number of slices in the given configuration and α is used to weight the contribution of this new factor. The new energy function E' reduces to E for those configuration having $n_s = m$. As will be shown in the next section, the use of *a-priori* knowledge (such as the modality of data) proves very effective in improving the performance of the algorithm.

4. RESULTS

The algorithm has been tested on several real images. As an example, the performance is discussed here with reference to the image shown in Fig. 2. It is well known that the cooling schedule and the generation mechanism may noticeably affect the performance of simulated annealing (see [1]). In order to assess the potential of the algorithm, two different cooling schedules

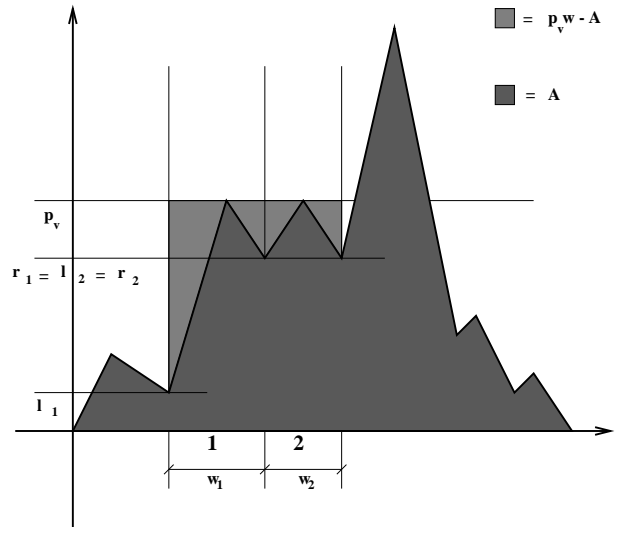


Fig. 1. Histogram values used in the score function.

and two different generation mechanisms have been compared. The initial temperature T_0 is chosen so that 95% of the generated transitions are accepted. The first cooling schedule $C1$ is a fixed (fractional) decrement annealing ([3], [4]):

$$T_{n+1} = 0.9T_n \quad (5)$$

where T_n is the temperature at the n -th annealing step. The alternative cooling schedule $C2$ is defined by:

$$T_{n+1} = T_n e^{\left(-\frac{T_n}{2\sigma_n}\right)} \quad (6)$$

where σ_n represent the standard deviation of energy at temperature T_n ([11]). The number of configurations generated at each temperature was held fixed for both cooling schedules and set equal to 10000. The annealing stops when the systems freezes (i.e. no accepted transitions are generated). The evolution of energy as a function of temperature for the fixed fractional decrement annealing can be seen in Fig. 4. Schedule $C2$ proves to be more efficient, exhibiting the same evolution of the average energy as a function of temperature but exploring less configurations using a faster cooling as can be seen from Fig. 5.

The performance of the algorithm using different generation mechanisms was also tested. The annealing of the system for two different neighborhoods δ_2 and δ_{10} (using the $C2$ cooling schedule) is reported in Fig. 5. The use of a greater neighborhood allows the system to cool faster at high temperatures thereby reducing the number of tested configurations. Finally the annealing obtained using the proposed energy functions E and E' , with $\alpha = 1.0, m = 4$, is compared (see Fig. 6). The use of the biased energy E' rapidly *focus* the system on the configurations with the required number of slices (see Fig. 7) which are near optimal.

The original histogram has 31 local minima (see Fig.3), giving a quite large configuration space (2^{31} possible configurations). The final histogram configuration (all of the experiments ended up with the same partition) has only four slices as can be seen in Fig. 8. The resulting segmentation is shown in Fig.9. The algorithm (using the $C2$ cooling schedule and the δ_{10} neighborhood) explored only $7.0 \times 10^{-5}\%$ of the total configuration space



Fig. 2. Sample image.

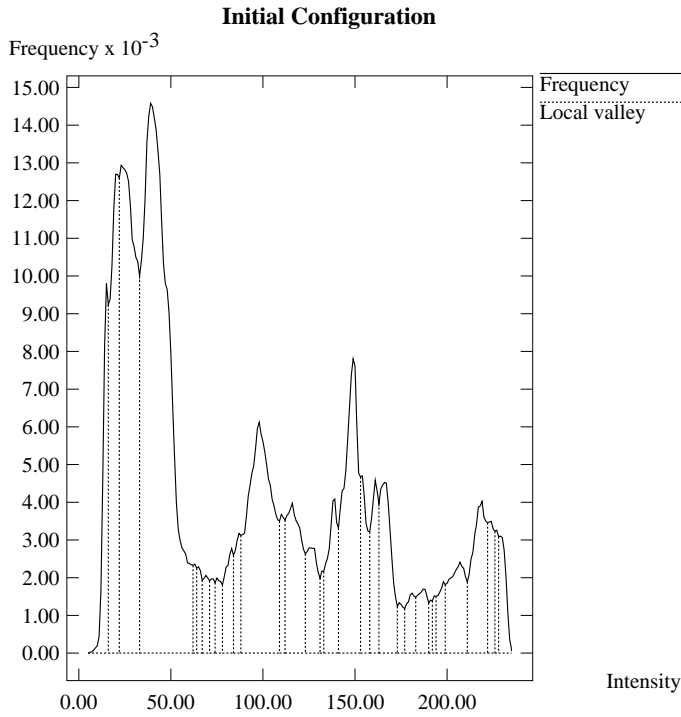


Fig. 3. Intensity histogram and starting configuration.

using energy function E (unconstrained number of slices) and $2.3 \times 10^{-5}\%$ using the biased energy function E' (with $m = 4$ and $\alpha = 1$).

5. CONCLUSIONS

Our experiments suggest that the use of simulated annealing techniques can prove useful in the task of image segmentation through histogram slicing, where an exhaustive search of the configuration space is very demanding except for the simplest problems. The possibility of defining energy functions reflecting the apriori knowledge of the problem is useful to obtain results that match human performance in the given task and to speed up the annealing cycle.

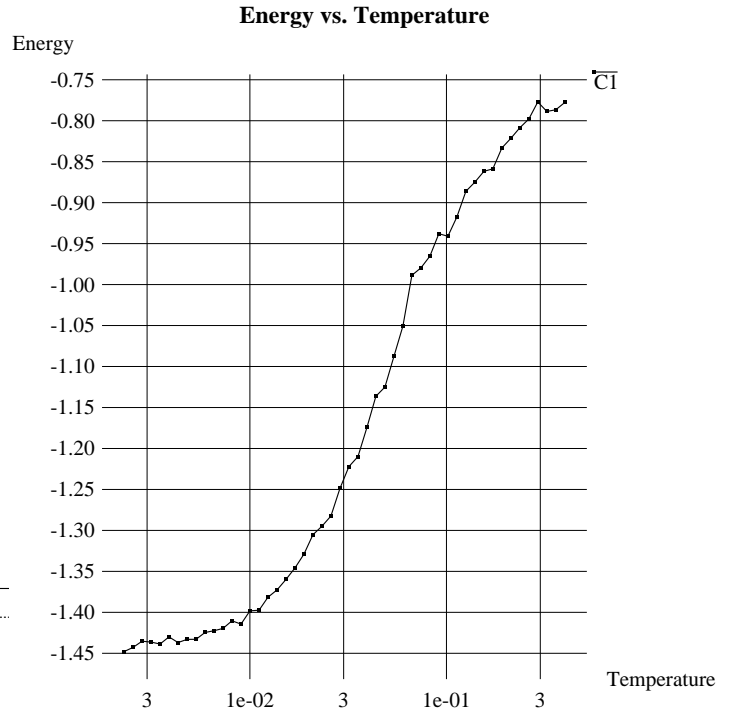


Fig. 4. Average configuration energy as a function of temperature using the $C1$ schedule and the δ_2 neighborhood.

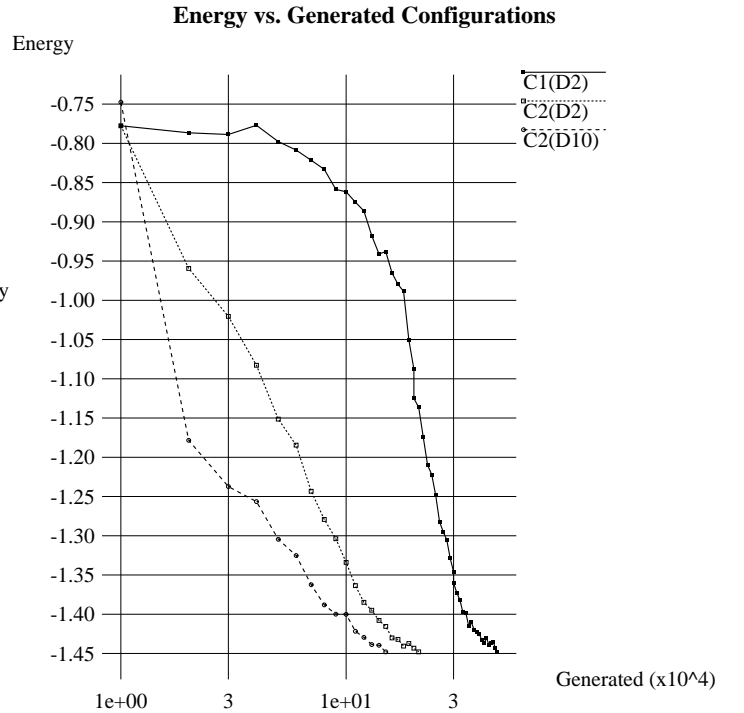


Fig. 5. Energy as a function of the number of generated configurations for different cooling schedules ($C1$ and $C2$) and for different neighborhood sizes ($D2$ and $D10$ correspond to neighborhoods δ_2 and δ_{10}).

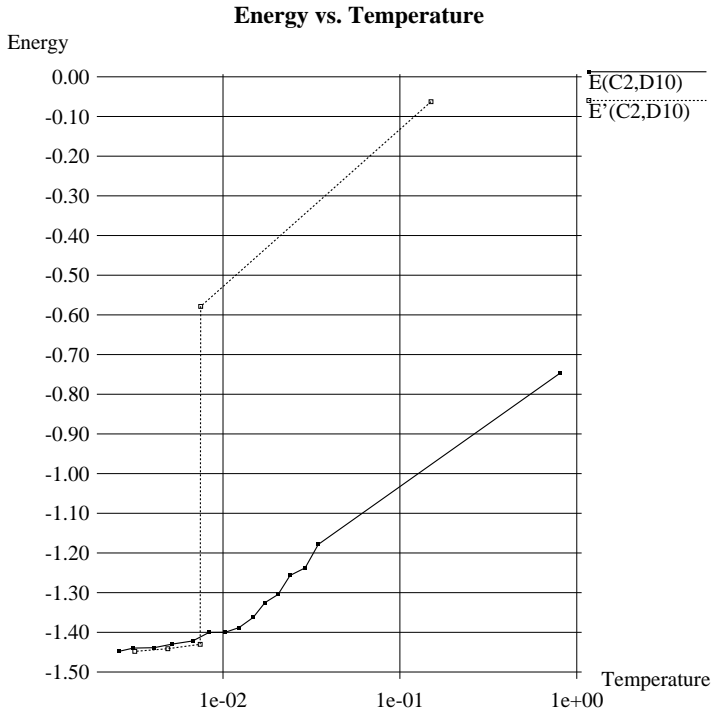


Fig. 6. Average energy as a function of temperature for the two energy functions E and E' (with $m = 4$ and $\alpha = 1$) using the C2 cooling scheme and the δ_{10} neighborhood.

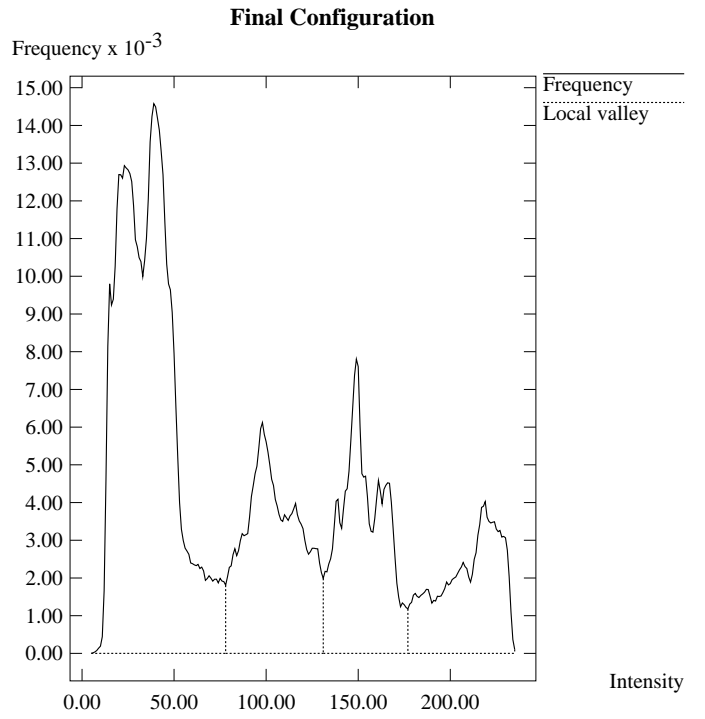


Fig. 8. Final configuration obtained by all of the annealing strategies.

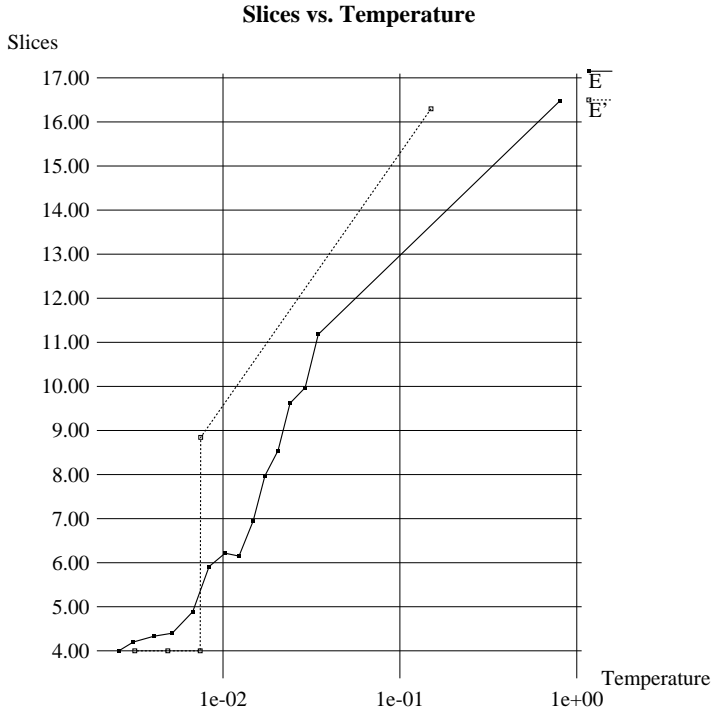


Fig. 7. Average number of slices for the (un)biased energy (using the C2 cooling scheme and the δ_{10} neighborhood).



Fig. 9. Histogram based segmentation.

ACKNOWLEDGEMENTS

The author thanks Dr. C. M. Modena, Dr. G. P. Tecchiolli and Prof. T. Poggio for many helpful suggestions and stimulating discussions.

REFERENCES

- [1] E. H. L. Aarts and P. J. M. van Laarhoven. *Simulated annealing: Theory and Applications*. Reidel, 1987.
- [2] E. H. L. Aarts and J. Korst. *Simulated annealing and Boltzmann Machines*. John Wiley & Sons, 1989.
- [3] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.
- [4] R. A. Rutenbar. Simulated annealing algorithms: An overview. *IEEE Circuits and Devices Magazine*, 5(1):19–26, 1989.
- [5] Y. Ohta. *Knowledge-based Interpretation of Outdoor Natural Color Scenes*. Pitman Publ., London, 1985.
- [6] D. H. Ballard and C. M. Brown. *Computer Vision*. Prentice Hall, Englewood Cliffs, NJ, 1982.

- [7] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *PAMI*, pages 629–639, July 1990.
- [8] R. C. Gonzalez and P. Wintz. *Digital Image Processing*. Addison-Wesley, 1987.
- [9] W. Snyder, G. Bilbro, A. Logenthiran, and S. Rajala. Optimal thresholding - a new approach. *PRL*, 11:803–810, 1990.
- [10] T. Poggio and F. Girosi. A theory of networks for approximation and learning. A.I. Memo No. 1140, Massachusetts Institute of Technology, 1989.
- [11] M. D. Huang, F. Romeo, and A. L. Sangiovanni-Vincentelli. An Efficient General Cooling Schedule for Simulated Annealing. In *Proc. IEEE Int. Conference on Computer-Aided Design*, pages 381–384, 1986.