to object decomposition and the feeding of separate vectors to the BPNN structure allows the recognition accuracy to be increased. By using two-band coding of the input vector, we reduce the total number of input elements of each BPNN. Thus, the required memory resources are reduced, as well as the time needed for computation. The identical structures of the BPNNs provide an opportunity to use the same running weight matrices and to perform parallel computations. The filtering BPNNs, and also BPNN, for general classification combining the components of the object, increase the recognition rate. The presented system does not require any preliminary knowledge; because it is a self-teaching system. Because it is possible to adjust the method for different kinds of objects, it can be applied in different fields of learning, in all computer vision and real time working systems.

References

Maze-tracing algorithm applied to eye-fundus blood vessels


A new maze or blood vessel tracing algorithm is proposed, inspired by the behaviour of flying bats which are looking for their way in complex caves, emitting ultrasonic waves in all directions and gathering the reflections to find obstacles. This idea is applied to trace eye-fundus blood vessels.

Introduction: Most existing methods for tracing mazes or blood vessel networks employ the following steps: (i) given a starting (base or centre) point and initial searching direction, (ii) proceed some distance in this direction, (iii) find the right and left boundaries (walls) on the perpendicular line to the given direction, (iv) check if any branch exists, (v) obtain the (base or centre) point between the left and the right boundaries, and (vi) consider the base point as a new starting point and obtain the new direction from previous to new base points [1 - 3]. In maze or blood vessel tracing, the most important information to obtain is that regarding where the boundaries are and in what direction the tracing should proceed. To achieve this, most methods check only the right and left data on the line perpendicular to the current tracing direction. It is probably preferable to get data from more directions, or from all directions as a bat does.

Methods: Fig. 1a shows the proposed tracing method, where the tracing so far has proceeded from the previous base point $P_i$ to the new base point $P_j$. The dotted lines are the boundaries detected so far. $R$ and $L_i$ are the last detected points on the right and left boundaries, respectively. Boundaries (zero-cross points) [1] for the present study, other sources may be used) are to be detected within the area of a circle of radius $r$. From $P_j$, the centre of the circle, the search beams are emitted with radius $r$ in all directions (angles) from $P_j-C$. The thick lines on the right and left are the boundaries which have just been detected. The beam hits the boundary at $R_i$ as in Fig. 1a. The directions in the hatched area have no detected boundaries, and the new directions are determined to be the centre directions between $P_j-R_i$ and $P_j-L_i$. The next base point should be on the line $P_j-P_i$ within radius $r$. With the new base point $P_i$, the same procedure is repeated as before.

Fig. 1 Tracing method
a No branch
b Two-way branch
c Three-way branch
d Dead end

The branches and the dead-end of the maze are classified as shown by Figs. 1a-d. Fig. 1a is for no branches, Fig. 1b is for the two-way branch, Fig. 1c is for the three-way branch, and Fig. 1d is for a dead-end. The dead end is also the case when the path becomes very narrow. At branches, one branch is processed and others are memorised on a queue to trace later. In the case of blood vessel tracing, the three-way branch is considered to be a crossing of different blood vessels [1] and the midway is taken to be crossing over the other vessel (Fig. 1c). To avoid retracing, the paths traced are marked so as not to be re-entered [3].

Fig. 2 Traced treelike image

Results: The proposed algorithm was applied to the output file of a second order derivative of Gaussian filters [1] which was applied to original monochrome images of 8-bit/pixel. The zero-crosses were used for boundary recognition. The algorithm is first applied to the human drawn treelike maze image to see how it works. Fig. 2 is the result. The trace line is successfully drawn throughout. At some points, however, the trace line is too close to the boundary.
The second image (Fig. 3) tested is the eye-fundus blood vessels image (very few research reports about tracing eye-fundus blood vessels have appeared in the literature). Fig. 4 is the result of this tracing. Three blood vessels were tested, of which the trace starting points are labelled as A, B and C. When Figs. 3 and 4 are compared, the traced lines look fairly good. The crosses near D and G were processed successfully, but the branches near E and F were not detected. Other branches were successfully detected. By checking the Gaussian filter output, we find that zero-crosses were observed at those branches near E and F, implying that those branches were somehow cut by the filter (probably owing to noise), suggesting that zero-cross data only is not enough for boundary information.

![Fig. 3 Original fundus image](image)

![Fig. 4 Traced blood vessels of fundus image](image)

Conclusions: A very simple algorithm is proposed to trace a maze or a blood vessel network. The idea comes from the navigational method of a bat. The experiments showed fairly good results. Since a human often uses a similar method when feeling his way in the dark, the method should be considered general. The advantage of this method is that information is obtained from many directions, rather than just from the right or the left. A more sophisticated algorithm could be developed using the idea, possibly combining some existing methods.

© IEE 1998

Electronics Letters Online no: 19980689

T. Kurokawa, M. Kondoh, T. Oda and S. Kito (Department of Industrial Engineering, Aichi Institute of Technology, 1247 Yagazucho, Toyota, 470-0356, Japan)

E-mail: kurokawa@iee.aitech.ac.jp

S.C. Lee (School of Electrical Engineering, The University of Oklahoma, 202 West Boyd, Room 219, Norman, Oklahoma 73019-0631, USA)

References


Median-rational hybrid filters for image restoration

L. Khriji and M. Gabbouj

A new class of nonlinear filters is introduced, called median-rational hybrid filters (MRHF's), based on rational functions (RFs). The output is the result of rational operation taking into account three sub-functions. It is shown that every sub-function will preserve details within its sub-windows. The proposed MRHF filters have the inherent property that on smooth areas they provide good noise attenuation whereas on changing areas the noise attenuation is traded for good response to change. It is shown that a consistent reduction in the objectively measured mean absolute error and mean square error is obtained.

Introduction: The rational filter is one of the most recent and major classes of nonlinear filters. As the name indicates, it consists of a ratio of two polynomials. It is well-known that a rational function has several properties (it is a universal approximator and good extrapolator, can be trained using a linear algorithm, and requires lower degree terms than Volterra expansions) which can make it very effective in many signal processing tasks.

Rational function filters were used by Leang and Haykin [1] based on the work of Walsh [2] for signal detection and estimation, and were later applied by Ramponi in [3, 4] for image filtering and enhancement. This was later extended to deal with multidimensional data, in [5], for color image interpolation.

Specifically, in the image restoration problem, the rational filter developed in [3] performs well for relatively high SNR Gaussian contaminated environments. To derive a rational filter to deal with various kinds of noise such as Gaussian noise, impulsive noise and mixed (Gaussian/impulsive) noise, we propose a new class of nonlinear rational type hybrid filter for signal and image processing, median-rational hybrid filters (MRHF's). The MRH filter is based on three sub-operators in which the central sub-operator is a centre weighted median CWM filter acting on a plus-shaped mask.

One-dimensional median-rational hybrid filters: Consider the input vector \( \mathbf{x}(n) = [x(n-N), x(n-N+1), \ldots, x(n-1), x(n), x(n+1), \ldots, x(n+N)] \), which contains \((2N+1)\) observation samples at each location \( n \). The output variable \( y(n) \) is the result of a rational function using three input sub-functions which form an input vector \( \mathbf{Φ} = [\mathbf{Φ}_1, \mathbf{Φ}_2, \mathbf{Φ}_3]' \), where the 'central' sub-function \( \Phi_2 \) is fixed as a centre weighted median filter. The proposed MRHF output is defined by

\[
y(n) = \Phi_2(n) + \frac{\sum_{i=1}^3 \alpha_i \Phi_i(n)}{h + k(\Phi_1(n) - \Phi_2(n))^2}
\]

where \( \alpha = [\alpha_1, \alpha_2, \alpha_3]' \) characterises the constant vector coefficients of the input subfunctions and satisfies the condition: \( \sum_{i=1}^3 \alpha_i = 0 \). In our study, \( \alpha = [1, -2, 1]' \). \( h \) and \( k \) are positive constants. The parameter \( k \) is used to control the size of the nonlinear effect.

The sub-filters \( \Phi_1 \) and \( \Phi_3 \) are chosen so that an acceptable compromise between noise reduction and edge preservation is obtained. It is easy to observe that this median-rational hybrid filter differs from a linear low-pass filter mainly in the scaling, which is introduced on \( \Phi_1 \) and \( \Phi_3 \) terms. Indeed, such terms are divided by a factor proportional to the output of an edge-sensing term characterised by \( \Phi_0(n) - \Phi_2(n) \). The weight of the median-operation output term is modified accordingly, in order to keep the gain