Matching an Elastic Model of Chromosomal Shape to Features on a Self-Organising Map[1].

M. Turner[a], J. Austin[a], N. M. Allinson[b] and P. Thompson[c].
Departments of Computer Science[a], Electronics[b] and Psychology[c],
York University,
Heslington, York, YO1 5DD, UK.

Abstract

We describe a technique for matching a single, learned elastic model of the shape of normal chromosomes to chromosomal images. Our model has a hierarchical organisation, with increasingly coarse shape descriptions at higher levels. A problem of finding the model description most likely to have generated an image is reduced to one of matching the locations of model points to the locations of image features encoded on a Kohonen Self-Organising Map (SOM). During matching, coarse shape information is communicated between levels via fixed, viewpoint-independent transformations between object-based frames. After matching, frame parameters provide a compact, multi-scale description of important shape information. We propose using this information to train a neural network to recognise structurally damaged chromosomes.

1. Introduction.

The statistical evaluation of chromosome structural damage is well-established both as a method of radiation dosimetry, and as a way of assessing the potential harm of chemicals. Analysis requires the search for different types of chromosomal structural aberrations which may be present in cells, but only at very low frequencies. Figure 1 shows some examples of aberration types. Predominantly, analysis is performed by skilled cytotechnicians using conventional microscopes. However, microscope work is slow, laborious and subjective, and there is a great demand for faster, more consistent techniques.

Detecting aberrations is a difficult problem due to the variability in chromosome appearance. Chromosomes may bend, touch or overlap. In addition, objects appear fuzzy under a light microscope, and staining gradients and the existence of artifacts can be troublesome. A number of automated systems have been built over the past twenty years [1,2,3,4]. Classification has been based on the measurements of predefined image features. All have concentrated on recognising a limited number of important aberration types from the dozen or so that may be distinguished [5]. Typically, attempts are made to detect just the 'dicentric', a rather specific indicator of ionising radiation, and comparatively easy to recognise in most cases from density profile and boundary curvature information [4].

We have been interested in the design of a neural network system for application

1. Supported by SERC grant 06R00174

BMVC 1993 doi:10.5244/C.7.50

to aberration recognition. The major characteristic of our system is its ability to learn through experience. In particular, image features are learned rather than predefined. We have not restricted ourselves to detecting dicentrics. Rather, we are interesting in attempting to detect any one of a set of a dozen aberration types commonly sought in toxicology studies [5]. As part of this approach we have recently described: (a) a neural network technique for the initial location of scattered chromosomal objects within images of human blood cells and (b) a neural network centring and classification system applied to low resolution cell images [6]. The system was centred around the use of a Kohonen Self-Organising Map (SOM) [7] to extract image features.

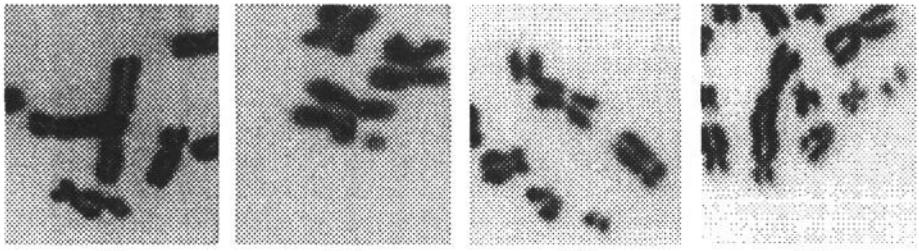

Figure 1. Sections from human blood cells showing examples of structural aberrations. Aberrations are (probably) (a) chromatid deletion (b) minute (c) deletion (d) dicentric and fragment.

A 'post-labeled' SOM was also used in the classification stage. A number of problems can be identified with this approach to classification: (1) The (feedforward) network possesses no prior knowledge of chromosome shape. All its information must be extracted from training data. (2) When the network performs a classification it does not give any explanation of its decision, nor does it make any shape information explicit. (3) The network has difficulty dealing with bent chromosomes. In order to overcome these problems we have taken a model-based approach. We use a single deformable elastic model of the normal chromosomal shape, which is matched with low-level image features. These features are extracted from chromosomal images by an SOM as in [6]. Once matched, information made explicit by the model may be used to classify the object. Classification has not yet been implemented, and is not described in this paper. A schematic diagram of the current system is shown in figure 2.

In this paper feature extraction is described in section 2, and our elastic model in section 3. The model-image matching technique is detailed in section 4. Some results and a brief discussion appear in section 5.

2. Feature Extraction.

Kohonen's SOM is a biologically-plausible neural network which has recently become popular, particularly in speech [8] and vision tasks, due to its ability to learn, without supervision, non-linear, topological mappings from high dimensional input spaces onto low dimensional coordinate systems. Typically, a map consists of a 2D array of units, each with its own adjustable template. During a period of training on example inputs, the templates adapt towards the salient features of a multi-dimensional input space. After training, the dimensions of the map correspond to the two 'most important' feature dimensions of the input space [9]. A further property of the map is that the density of the units' templates

approximates the density of the original input space. Consequently, each unit is responsive to roughly the same proportion of input signals.

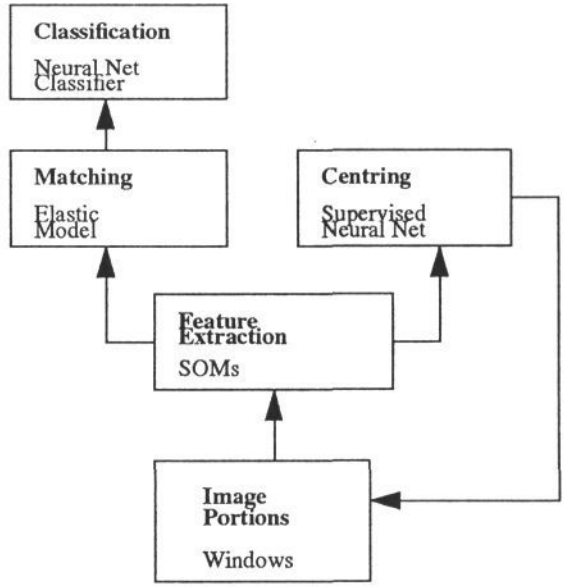

Figure 2. Schematic diagram of centring and classification system.

Our SOM consisted of a 2D array of 10 x 10 units. Input vectors were the set of image portions grabbed from windows placed in the vicinity of chromosomal objects. Windows were 8 x 8 pixels in size. (We rotated image portions before passing them as input to the network by a random amount so as not to bias the mapping towards any one orientation). Details of the training algorithm used are as given in [6], with the single exception that the initial 'training neighbourhood' was 5 x 5 units. An example of the templates learned by the network is shown in figure 3.

An inspection of figure 3 suggests that the SOM features are portions from a roughly circular blob. Furthermore, the map dimensions correspond, approximately linearly, to the x and y coordinates of the centre of the blob relative to an image window.

Once trained, the SOM in figure 3 was used to extract features in the vicinity of objects, located within cell images by the neural network technique described in [6]. A set of small, overlapping windows was placed on each located object. There were 19 x 19 windows in the set, each 8 x 8 pixels in size, and overlapping by 50%. Effectively, the SOM was duplicated for each window, so that there were 19 x 19 identical SOMs, which we arranged as a set of tiles. Each SOM extracted a single image feature from its window, encoded by the location of its most responsive unit. These locations were passed, initially to the centring neural network in [6], and then, once a stable position for the window set had been found, to the elastic matching subsystem. Note that 'homogeneous' image portions, as defined by a simple statistical test, were not encoded by the SOMs.

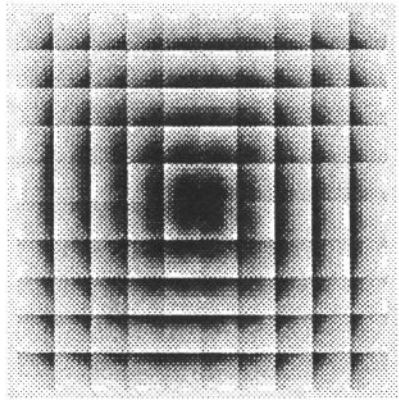

Figure 3. Templates in the SOM for high resolution images.

3. Elastic Model of a Normal Chromosome.

Elastic models [10,11,12] are just one of a class of physically-based deformable models, prevalent in computer graphics, in which deformation is governed by the laws of mechanics, in this case elasticity theory. Their selection was influenced by their successful application to a number of difficult matching problems, notably the Travelling Salesman Problem [13] and hand-written character recognition [14]. Unlike Hinton et al, who employed an elastic model for each shape of interest, we use just a single model of the shape of a normal chromosome. That is, we do not have models for the shapes of damaged chromosomes. There are two reasons for this. Firstly, elastic models are computationally expensive. Matching many models to an image would be prohibitively slow. Secondly, we consider each chromosomal object to be composed of a normal chromosome plus possible damage, and the main task of matching is to make explicit information about the damage *relative to* the normal chromosome. It is this information which we propose using to classify objects. Our model has a hierarchical organisation, with shape descriptions of increasing coarseness at higher levels in the hierarchy. Shape information is passed between these levels during matching. For the moment, however, it is easiest to consider the shape description in a single hierarchy level.

3.1 A Single Model Description.

Typically, an elastic model consists of a set of points at given spatial locations. These locations define an ideal shape, and elastic connections between points define shape deformations from the ideal. However, the points in our model map onto locations on SOMs, which define a set of model features. Therefore, the 'nature' of our model points depends upon the feature dimensions learned by the SOMs, and need not be spatial. In fact, the current model points are spatial, since our SOM encodes the locations of blobs. However, we are carrying out experiments in which multiple SOMs are trained, and model points map, not only to the spatial locations of blobs, but also to non-spatial features such as the orientation and extent of a bar.

As stated, SOM features are portions from a blob, and the locations on an SOM can be thought of as linearly encoding the blob's location relative to the image window. However, we assume that it is spatial locations relative to the image, that is, relative to the space defined by the set of image windows, that is important for matching. We create a new space, above the SOMs, to encode such locations. This is the model point space, and it is proposed that model-image matching should take place in this space. We map from a location $\mathbf{x}_i$, on the *ith* SOM at location $\mathbf{m}_i$ in the tiled set, onto a location $\mathbf{y}$ in model point space, such that

$$\boldsymbol{y} = a\boldsymbol{m}_i + b\boldsymbol{x}_i \tag{3.1}$$

where a and b are constants. The first term on the right hand side of equation 3.1 can be thought of as relating to the location of the *ith* window relative to the image, and the second term relates to the location of a blob relative to this window. The constants a and b are found experimentally, by requiring the locations of SOM features that correspond to portions from the same blob in the image domain, to map onto the same location in model point space. The mapping is reversible. A model point location maps onto those SOM locations which encode portions from the same blob in the image domain.

We make use of 3.1 to encode an image, via the SOMs, as a set of locations in model point space (see section 4.1). The reverse mapping is useful when interpreting a model description. Each model point in a description maps onto SOM locations which in turn map onto a blob in the image domain. Similarly, the length of an elastic connection between a pair of model points maps onto a distance between blobs. Furthermore, the strength of the elastic maps to the strength of a constraint that the blobs should remain at a constant distance apart.

'Object-based' frames are established in a model description in order to segment the model into 'stable' shape regions, i.e., regions that may change position, orientation or scale, but are subject to few shape distortions. Each frame is specified by the parameters [x,y,c,s] where (x,y) is the frame's position, and c=rcosα, s=rsinα, where r is the frame's scale and α its orientation. The importance of the frames rests with their ability to provide a suitable framework in which to express shape information, and we will discuss them in more detail in subsection 3.2.

A single model level description consists of elastically-connected model points, segmented into regions by, and defined relative to, object-based frames. We deform a model by changing the length of the elastic between pairs of model points. This corresponds to changing the distance between blobs in the image domain. The deformed model is called a model *instance*. We assume that the probability of an instance is the probability of generating the elastic lengths from the points in each segmented region independently, and that elastic lengths have a Gaussian distribution about their resting lengths in the undeformed model. The probability of a model instance $\mathbf{M}$ given a model $\mathbf{M}^0$ is

$$P(\boldsymbol{M}|\boldsymbol{M}^0) \propto \prod_{r,\, i \in r,\, j} exp\left(-K_{ij}(l^r_{ij} - l^{r0}_{ij})^2\right) \tag{3.2}$$

where r labels a segmented region and K_{ij} is the strength of the elastic connection between model points i and j. l^r_{ij} is the distance, relative to the frame for region r, between model points in the model instance, and l^{r0}_{ij} is the distance between the *home* locations of the model points. The cost associated with a deformation, i.e., its elastic energy, is proportional to the negative log of this probability.

As a model description is matched with the image (see section 4), model point locations change. We need to update object-based frames in order to compensate for these changes, so that the frame parameters continue to provide reasonable estimates of the position, orientation and scale of a region. We update frame parameters for each region's frame by finding the changes in position, rotation and scale that minimise the squared Euclidean distance between the home locations and current locations of points in the region.

3.2 A Hierarchically-Organised Model.

Our model has a modular, hierarchical organisation of the kind suggested by Marr for object recognition [15]. Each level in the hierarchy contains a shape description of the type detailed in section 3.1. As we move up the hierarchy, the descriptions become increasingly 'coarse'. At the top level, the description consists of just a few, elastically-connected model points, defined relative to a single object-based frame. The parameters of this frame specify the global position, orientation and size of a chromosome. At lower levels, there may be many frames, each encompassing a stable region of the model for that scale of description.

Within the hierarchy there are weighted connections between 'parent' frames in one level and 'constituent' frames in the level below. These connections encode fixed, viewpoint-independent shape transformations between model regions in different descriptions. Transformations ensure that, as the parameter values of a parent frame change during matching at its level, its constituent frames maintain the same relative position, orientation and scale (see Zemel [16]). Constituent frames are themselves parents to frames in the next level down, so that shape information feeds all the way down to the lowest hierarchy level. Since frames are established in stable shape regions, they tend to provide a good (and compact) descriptions of the shape at the currently matched level.

In addition to providing a simple, compact way of communicating shape information during matching, the object-based frames also enable important shape information to be made explicit after matching. This information may subsequently be used in classification. In particular, the changes in a frame's parameter values relative to its parent(s) can easily be found. Such changes provide information, in the 'appropriate contexts' [15], about how the estimated shape of the image differs from that of a normal chromosomal, as encoded by the model. The contexts are appropriate because, when matching with a normal chromosome, constituent frames tend to remain stable with respect to their parents. For example, the disposition of a chromosome's tip is most stable when specified relative to the arm that contains it. Likewise, the disposition of an arm is most stable when specified relative to the main body of a chromosome. Instability is significant, and may indicate structural damage. For example, large changes in the relative position of a tip may indicate tip breakage, and large changes in relative orientation suggests severe bending.

Each description of the normal chromosome shape at each hierarchy level is initially set by hand. This is easily achieved by specifying model point locations and object-based frames, and by setting all elastic connection strengths to be equal. However, connection strengths and transformations between frames are subsequently updated during a period of training on example images of normal chromo-

somes. For each example a match of the full model hierarchy is performed using the current model (see section 4). The state of the model after matching is thought of as a 'true' description of the image. Connection strengths in each level are then adjusted in order to minimise the elastic energy at that level, subject to the constraint that the strengths sum to 1. As a consequence, strengths between model points whose variance is large during training decreases. Weights encoding the viewpoint-independent transformations in the model are updated so that transformations are closer to those consistent with the frame parameters in the model.

4. Model-Image Matching.

The approach taken to matching an image is to find, for each hierarchy level in turn, the model instance most likely to have generated it. Shape information about the match at one level serves to update model descriptions in levels below.

4.1 Matching a single model description.

At this point it helps to consider how we might generate a set of image portions from a model description. Generation involves three stages:
(1) Create a model instance by deforming the model. The probability of picking an instance is $P(\mathbf{M}|\mathbf{M}^0)$ as in equation (3.2).
(2) Map from each model point onto the locations of 'feature generators' on the SOMs, as desribed in section 3.1. Place a probability distribution on each generator of picking a feature encoded at some distance from it on the map.
(3) Either (a) select a generator, pick a map location from its distribution, and map the template chosen into the appropriate image window or (b) select a unit at random with a probability P_{noise} and map its template into its image window.
(4) Repeat (3) many times.

Now consider the probability of generating a set of image portions from a model in the manner described above. Given $\mathbf{M}^0=\{\mathbf{M}^0_k\}$ to be the set of home locations of model points relative to model point space, we generate a model instance by deforming the model. Using Bayes' theorem, and just considering the best-fitting model instance, we can approximate the probability of a generating the set of image portions $\mathbf{I}=\{\mathbf{I}_i\}$ by

$$P(\boldsymbol{I}|\boldsymbol{M}^0) = P(\boldsymbol{I}|\boldsymbol{M})\,P(\boldsymbol{M}|\boldsymbol{M}^0) \qquad (4.1)$$

where $\mathbf{M}=\{\mathbf{M}_k\}$ is the best fitting model instance.

The probability of generating a single image portion from a model instance is the sum of the probabilities of all possible ways of generating it from the model points (via the SOMs), or from noise. We now make two simplifying assumption, based upon characteristics of an SOM. Firstly, we assume good vector quantisation by the SOM. That is, the templates of units that encode image portions are 'close' to those portions. Consequently, the probability of generating an image portion from a feature generator is proportional to the probability of generating the best-matching SOM feature template for that image portion. (A test on 1000 image portions not present during network training had the scalar product between portions and templates 0.97+-0.01). Secondly, we assume that the dimensions of our SOM correspond to two new random variables with a uniform distribution. In such circumstances it is reasonable to give our feature generators a Gaussian probability

distribution. Under these two assumptions it is now possible to write the probability of generating a single image portion as the sum of the probabilities of all possible ways of generating the location that encodes it on the SOM from the location of the feature generators on the SOM, or from noise. Furthermore, we can map this probability onto model point space, and write,

$$P(\boldsymbol{I}_i|\boldsymbol{M}) = p_{noise} + (1 - p_{noise}) \sum_k \lambda_k exp(-\beta|\boldsymbol{Y}_i - \boldsymbol{M}_k|^2) \qquad (4.2)$$

where $\mathbf{Y}_i$ the location of an 'image point', namely the location on the SOM encoding the *ith* image portion *mapped onto model point space*. λ_k are normalisation constants, and β defines a 'range of attraction' for model points.

The cost of a fit is proportional to the negative log probability of generating the image portions from the model and we update the location of model points in order to reduce this energy by gradient descent. The result is

$$\Delta \boldsymbol{M}_k = -\alpha \sum_n w_{kn} (\boldsymbol{Y}_n - \boldsymbol{M}_k) + \gamma \sum_{r,\, m \in r} K_{mk} \boldsymbol{l}^r_{mk} \frac{\Delta l^r_{mk}}{l^r_{mk}} \qquad (4.3)$$

where

$$w_{kn} = \frac{(1 - p_{noise}) P(I_n|M_k)}{P(I_n|M)} \qquad \text{and } \mathbf{l}_{mk} = \mathbf{M}^r_k - \mathbf{M}^r_m$$

with $\mathbf{M}^r_k$ being the current location of the *kth* model point relative to the frame for the *rth* model region. An interpretation of equation (4.3) is as follows. Matching occurs in model point space. Each model point has a radial Gaussian probability distribution of generating an image point at some distance from it in the space. Model points are attracted towards each image point by an amount depending on their relative probability of generating the image point compared with the probability of generating the point from the model as a whole. Deformation is subject to constraints imposed by the elastic connections. The relative importance of matching the image points compared with minimising deformations in controlled by the parameters α and γ.

4.2 Matching a Hierarchically-Organised Model.

The previous section described how a single model description is matched to the image. The matching procedure for the hierarchical model consists of the following procedures:
(1) Selecting the top level in the hierarchy.
(2) Performing an elastic match for that level to maximise the probability of generating the image from the model description.
(3) Updating each frame at the current level to compensate for the match.
(4) Updating the frames in levels below via the fixed viewpoint independent transformations.
(5) Repeating (2)-(4) a few times.
(6) Moving down a level in the hierarchy and going to (2) (unless at the bottom level in the hierarchy).

The variance on model points, relating to the radii of blobs in the image domain, is

gradually decreased at lower levels in the model hierarchy. This decreases the range of attraction of the model points. Similarly, the elastic coefficient, γ, is gradually decreased, thereby reducing the relative importance of maintaining elastic lengths, compared with matching the image points. This strategy is similar to that employed in [13,14].

At higher levels in the model hierarchy large variances serve to smooth the energy function, so that it contains just a few, possibly one, minima. The hope is that the matching technique finds the global minimum at each level. Furthermore, we hope that the shape information communicated to lower levels provides a good starting point for matching at these levels.

5. Results and Discussion.

A comprehensive test of the matching technique is currently underway. Initial results do suggest that the model can capture shape variability and deform to match isolated images of normal and damaged chromosomes. In the initial experiments three levels of description were employed in the model hierarchy. The parameters $[\alpha,\beta,\gamma]$ were set at [2.22,0.09,0.60] at level 1, [4.00,0.05,0.30] at level 2 and [13.33,0.015,0.05] at level 3, and P_{noise} was 0.10. Three iterations of the matching schedule were performed at each level. Figure 4 shows some results of matching with images of isolated chromosomes.

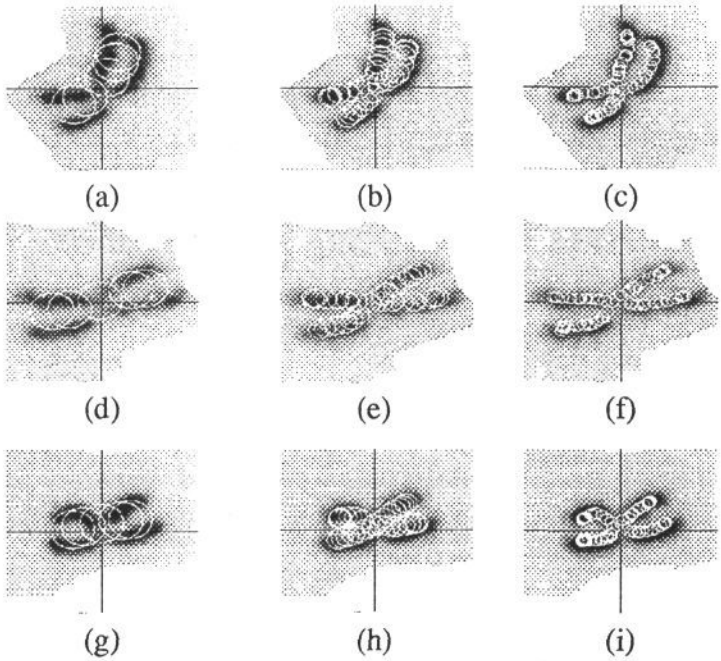

Figure 4. Examples of elastic matching in the model hierarchy for the model of a normal chromosome shape. Model points have been mapped down onto the image space. The centres of circles relate to the position of model points, their radii to the standard deviations on points during matching. (a), (b) and (c) show the results after matching with an image at each of the three levels in the model hierarchy. Similarly for (d), (e), (f) and (g), (h), (i).

Chromosomal images frequently contain clutter or overlapping objects, and it is important that an elastic model does not get stuck in a local minimum when matching such images. We are currently investigating the behaviour of our model in cluttered images. At present, our SOM only encodes the spatial location of blobs. No information about, say, local orientation, or the position of tips or centromeres is available *during matching*, although such information may be extracted from the model descriptions after matching. Our model flexes to match spatial locations only, but in doing so may stray onto nearby clutter. We intend making richer descriptions of local model and image regions available during matching, by incorporating multiple SOMs into our system. Multiple SOMs fit naturally into the current framework. We are also looking to train a neural network to classify chromosomal objects based upon the shape information made explicit by frames in the model descriptions.

References.

[1] M.A. Bender, M.A. Kastenbaum, C.S. Lever and D.R. Pelster (1972). Combination Bayesian and logical approach to analysis of normal and abnormal chromosome spreads. *Computers in Biology and Medicine*, V2, 151-166. Pergamon Press, UK.

[2] N. Wald, C.C. Li, J.M. Davis and S.R. Fatora (1976). Automated analysis of chromosome damage. *Automation in Cytogenetics*. M.L. Mendelsohn (editor), National Technical Information Service, U.S. Dept. of Commerce, 39-45.

[3] T. Lorch, C. Wittler, G. Stephen and J.Bille (1989). An automated chromosome aberration scoring system. *Automation of Cytogenetics*, C. Lundsteen and J. Piper (editors), Springer, Heidelberg, 19-30.

[4] R. Bayley, A. Carothers, X. Chen, S. Farrow, J. Gorden, L. Ji, J. Piper, D. Rutovitz, M. Stark and N. Wald (1991). Radiation dosimetry by automatic image analysis of dicentric chromosomes. *Mutation Research*, V253, 223-235.

[5] J. R. K. Savage (1975). Classification and relationships of induced chromosomal structural changes. *Journal of Medical Genetics*, V12, 103-122.

[6] M. Turner, J. Austin, N.M. Allinson and P. Thompson (1993). Chromosome location and feature extraction using a neural network system. *Image and Vision Computing: BMVC 92 Special Edition*, V11, N4, 235-239. Butterworth-Heinemann Ltd.

[7] T.Kohonen (1984). *Self-organisation and associative memory*. Springer-Verlag, Berlin.

[8] T. Kohonen (1989). Speech recognition based on topology-preserving neural maps. *Neural Computing Architectures*, 26-40. I. Aleksander (editor). Noth Oxford Academic Publishers Ltd., London.

[9] T.Kohonen (1990). The self-organising map. *Proceedings of the IEEE*, V78, N9, 1464-1480.

[10] D.J. Burr (1981). Elastic Matching of line drawings. *IEEE PAMI*, V3, N6, 708-713.

[11] D. Terzopoulos and K. Fleischer (1988). Deformable models. *The Visual Computer*, V4, 306-331. Springer-Verlag.

[12] D. Terzopoulos, J. Platt, A. Barr and K. Fleischer (1987). Elastically deformable models. *Computer Graphics*, V21, N4, 205-214.

[13] R. Durbin and D. Willshaw (1987). An analogue approach to the travelling salesman problem. *Nature*, V326, 689-691.

[14] G.E. Hinton, C.K.I Williams and M.D. Revow (1992). Combining two methods of recognising hand-printed digits. *Artificial Neural Networks*, V2, 53-60. I. Aleksander and J. Taylor (editors). Elsevier Science Publishers.

[15] D. Marr (1982). *Vision*. W.H. Freeman and Company.

[16] R.S. Zemel, M.C. Mozer and G.E. Hinton (1990). TRAFFIC: Recognising objects using hierarchical reference frame transformations. *Advances in Neural Information Processing Systems*, V2, 266-273, D.S. Touretzky. Morgan Kaufmann Inc., San Mateco, CA.