CHROMOSOME COUNTING VIA DIGITAL IMAGE ANALYSIS

Victor Gajendran and Jeffrey J. Rodríguez

Dept. of Electrical & Computer Engineering
The University of Arizona, Tucson, AZ 85721 U.S.A., rodriguez@ece.arizona.edu

ABSTRACT

Currently, cytogenetic analysis is done using semi-automatic methods, involving substantial human intervention, which is slow and expensive. Chromosome counting is of significant economical importance, for it accelerates the karyotyping process, thereby facilitating quicker and less expensive cytogenetic analysis. In this paper, we present an automated counting algorithm based on digital image analysis. The algorithm was tested on a metaphase image database, with a 6% error rate.

1. INTRODUCTION

Fig. 1 contains a digital image of normal female human chromosomes in the metaphase stage. A normal human diploid cell has 46 chromosomes (44 autosomes and two sex chromosomes). If there is even a small deviation from these numbers, then physical abnormalities will result. In genetic testing, manual chromosome counting and karyotyping (chromosome classification) are very slow processes. Therefore, hospitals often must employ additional technicians to manually analyze the metaphase images simultaneously to meet the incoming requests, making the process very expensive.

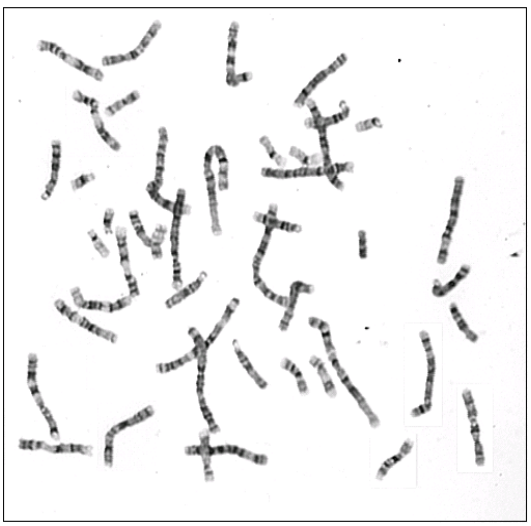

Fig. 1: Metaphase image

Current computer-based systems for chromosome analysis are mostly interactive and require human intervention to separate touching or overlapping chromosomes [1,5,8]. Since occlusion of chromosomes occurs in almost every metaphase image, the solution of this problem is vital. Typical approaches reported in the literature work well for simple cases, but tend to fail in more complicated situations where there is ambiguity or touching or overlapping chromosomes [1,2,4,5,7-9]. Other methods [2,10] are based on analysis of chromosome shape, but they are not well suited to the non-rigidity of chromosomes. Geometric separation of partially overlapping chromosomes is a technique that works by identifying the contours of the chromosomes via polygon approximation [1]. But this method also performs poorly in the case of complex clusters with more than two chromosomes. Nevertheless, the geometric separation method is the best of the prior techniques, with a global error of about 18%.

2. AUTOMATIC CHROMOSOME COUNTING SYSTEM

In this work, an automatic chromosome counting system was developed, based on digital image analysis. Specifically, the new system combines Otsu's threshold selection method, hysteresis thresholding, median filtering and thinning along with newly developed methods for average width calculation, chromosome separation and chromosome counting. The automatic chromosome counting system comprises two main stages, namely pre-processing and counting.

2.1. Pre-Processing

The pre-processing stage involves hysteresis thresholding, median filtering, thinning, separation and cleaning. Hysteresis thresholding segments the chromosomes from the background. The thresholds are 10% above and below Otsu's threshold [6]. Median filtering [3] removes the salt and pepper noise and fills the small holes in the body of the chromosomes. This stage also smoothes the chromosome contours so that thinning does not produce small, unwanted branches. Morphological opening and closing were investigated as

0-7803-8554-3/04/$20.00 ©2004 IEEE.

an alternative to median filtering, but were not found to perform as well for this application.

The binary image obtained from the median filtering is thinned [3] to obtain single-pixel-wide skeletons representing the clusters of chromosomes.

Based on examining a vast set of images, it is evident that the width of chromosomes is almost a constant for a given image, but varies from image to image, based on the image resolution. In order to automate the process to work for any metaphase image, the average width is computed for a given image using the median-filtered image in combination with the corresponding skeletons. The width of a chromosome at a given point is defined as the number of pixels in the line perpendicular to the skeleton at that point. The average width of the chromosomes in an image is later used for eliminating blobs of noise, identifying slight connections, separating chromosomes, and for solving some implementation issues.

Having determined the average width of the chromosomes in an image, the next step is to identify the region where there are slight connections and then eliminate them. To identify the slight connections, the same technique used for computing the width is applied – i.e., the width at all points in the skeleton is calculated. Those points where the width of the chromosome is less than half the average width are marked as slight connections. This threshold – half the average width – was selected to make sure that a single chromosome with a small bend in its contour is not cut into two.

Following chromosome separation, the skeletons are recomputed, as shown in Fig. 2. At this point, the skeleton image is made ready for final noise removal and chromosome counting.

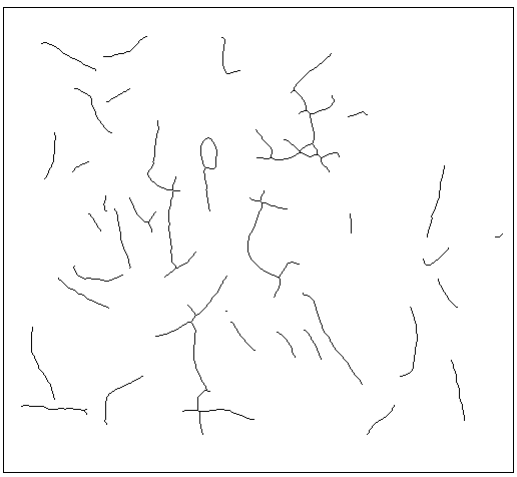

Fig. 2: Metaphase skeleton obtained after median filtering, thinning, and separation.

The pre-processed image at this stage might still have some noisy blobs remaining, due to the presence of cytoplasmic background components. The skeletons of such components will, however, be very small compared to those of the chromosomes. Nevertheless, these unwanted components would lead to an incorrect chromosome count, so the noisy blobs are eliminated by removing any skeleton components whose length is less than the average width of the chromosome.

2.2. Chromosome Counting

The counting stage takes a noise-filtered, skeletonized image as input. To facilitate the description of the counting algorithm, let the following terms be defined:

End Point: A point in the skeleton that has only one neighboring skeleton pixel.

Crossover: A point in the skeleton that has more than two neighboring skeleton pixels.

Trace: A path through the skeleton, beginning at an end point and ends at another end point. Generally, a trace is counted as one chromosome, except during special trace-sequence cases.

The newly developed algorithm that counts the number of chromosomes in the skeleton image is described as follows:

1. Label the skeletons in the input image.
2. For each component,
   a. Identify the end points.
   b. Identify the crossovers.
   c. Start from the end point that appears first in raster-scan order and keep tracing the skeleton until another end point is reached. At a crossover, take the path that appears first in raster-scan order.
   d. As the skeleton is being traced, delete all the pixels encountered, except for the crossover pixels.
   e. Increment the count of chromosomes by one for that component.
   f. Repeat steps a to e until all end points in the components are deleted.

The sequence of operations is illustrated in Fig. 3. The arrows indicate the traces, and the labeled circles indicate the chromosome count. The image segment at the top left of Fig. 3 represents the skeleton of a cluster consisting of three chromosomes. The next image segment indicates the trace using a thick curved arrow starting from the pixel that appears first in raster-scan order, running until an endpoint is reached. The next image segment shows the trace deleted from the component skeleton. This procedure is repeated until all the skeleton pixels are deleted from the connected component. This algorithm works for arbitrary

chromosome orientations. Some of the complex components that were successfully evaluated by the counting algorithm are shown in Fig. 4.

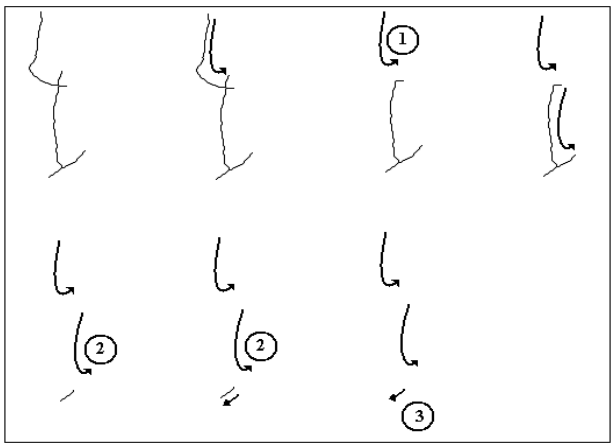

Fig. 3: Sequence of images explaining the chromosome counting algorithm.

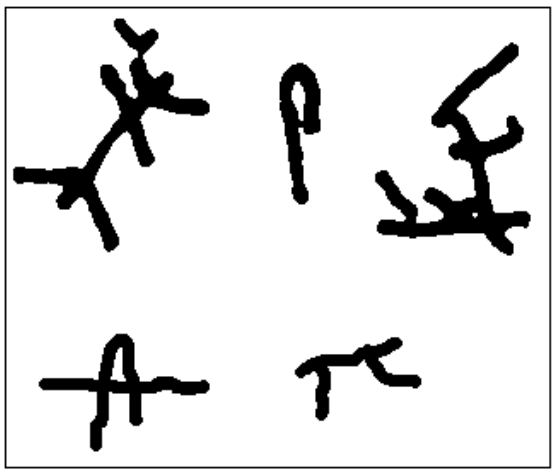

Fig. 4: Successfully resolved chromosome clusters.

3. EXPERIMENTAL RESULTS

3.1. Experimental Data

The metaphase images used for testing the system performance were obtained using the DSS Imagetech Cytovision software in the Cytogenetics laboratory in the Department of Pathology at the University of Arizona. The image acquisition setup involved a 100x microscope connected to a computer with the DSS Imagetech Cytovision software installed. Three different types of image databases were collected. The first database consists of chromosome images obtained by directly capturing the metaphases. This database has background noise as well as several connections and overlaps between chromosomes. It is called the Metaphase Image Database in this report.

The second database consists of metaphase images whose background noise was manually removed by a cytogenetics technician. This database has several overlaps and connections between chromosomes but no background noise, and so it is called the Background-Noise-Free Image Database. The third database consists of the karyotypes of the metaphase, manually created by an expert. Each of these three databases consists of sixty-seven 8-bit 512 x 512 images.

3.2. Calculation of Global Error

Let K be the number of clusters in a metaphase image. The global error ξ_G was calculated using the observed total count $\hat{N}$ and correct total count N of chromosomes as follows:

$$\xi_G = \frac{\left|\hat{N} - N\right|}{N} = \frac{\left|\sum_{i=1}^{K}\left(\hat{N}_i - N_i\right)\right|}{N}$$

where $\hat{N}_i$ is the observed count of chromosomes in the ith cluster, and N_i is the correct count of chromosomes in the ith cluster.

3.3. Calculation of Cluster-Based Error

If the system counts one chromosome less in one cluster and one chromosome more in another cluster, then these errors will cancel, yielding 0% for the global error. To perform a more useful performance evaluation, the cluster-based error ξ_C was defined:

Let K be the number of clusters in a metaphase image. Cluster-based error ξ_C was calculated using $\hat{N}_i$, the observed count of chromosomes in the ith cluster, and N_i, the actual count of chromosomes in the ith cluster, as follows:

$$\xi_C = \frac{\sum_{i=1}^{K}\left|\hat{N}_i - N_i\right|}{N}$$

The cluster-based error consists of two components, namely the cluster-based error caused by chromosomes being cut, ξ_{Cut}, and the cluster-based error caused by chromosomes being connected, ξ_{Conn}.

$$\xi_{Cut} = \frac{\sum_{i=1}^{K} \left| \hat{N}_{Cut,i} \right|}{N} \qquad \xi_{Conn} = \frac{\sum_{i=1}^{K} \left| \hat{N}_{Conn,i} \right|}{N}$$

$$\xi_C = \xi_{Cut} + \xi_{Conn}$$

where $\hat{N}_{Cut,i}$ is the number of errors caused due to chromosomes being cut in the cluster *i,* and $\hat{N}_{Conn,i}$ is the number of errors caused due to chromosomes being connected in cluster *i*.

3.4. Performance of the Complete System

The complete chromosome counting system was tested using the complete set of metaphase images in the Background-Noise-Free Image Database and the Metaphase Image Database. The 10 training images that were used in the preliminary experiments were not included in these experiments. Using the correct counts from the karyotype images, the accuracies were calculated, and the averages were tabulated in Table 1.

The table shows two rows of numbers. The first represents the results from the experiments conducted with the Background-Noise-Free Image Database, and the second represents the results from the experiments with Metaphase Image Database.

Case	Errors due to cut chromo-somes (%)	Errors due to linked chromo-somes (%)	Global error (%)	Cluster-based error (%)
Bg-Noise-Free Image Database	1.8	2.1	3.1	3.9
Metaphase Image Database	3.1	3.3	5.7	6.4

Table 1: Performance of the complete chromosome counting system.

3.5. Computation Time

The prototype software system was developed in Mathworks MATLAB on a Pentium III computer with 128 MB RAM, using the Microsoft Windows operating system. The execution time for the automatic chromosome counting system, starting from feeding in the raw input image to getting the output count of chromosomes, ranges from 2.3 to 5.5 minutes. The algorithm could easily be re-implemented in a compiled language to significantly reduce computation time.

4. CONCLUSION

In this work, a new pre-processing system was developed, which uses existing digital image analysis techniques, such as Otsu's threshold selection method, hysteresis thresholding, thinning and median filtering. In addition, a new counting algorithm was developed, which uses topological analysis to derive the chromosome count. The entire system, including the pre-processing stages, had 6% error. The main source of error is in the pre-processing stages, where the background noise can lend to unwanted connections or breaks in the chromosomes.

REFERENCES

[1] G. Agam, I. Denstein, "Geometric separation of partially overlapping non-rigid object applied to automatic chromosome classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, 1997, pp. 1212-1222.

[2] D. A. Denisov, A. K. Dudkin, "Model-based chromosome recognition via hypothesis construction/verification," *Pattern Recognition Letters*, vol. 15, 1994, pp. 299-307.

[3] A. K. Jain, "Fundamentals of Digital Image Processing", Prentice Hall Publications, Englewood Cliffs, New Jersey, 1989.

[4] B. Lerner, "Toward a completely automatic neural network-based human chromosome analysis," *IEEE Transactions on Systems, Man, & Cybernetics*, Part B: Cybernetics, vol. 28, no. 4, 1998.

[5] J. Liang, "Fully automatic chromosome segmentation," *Cytometry*, vol. 17, 1994, pp. 196-208.

[6] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, & Cybernetics*, vol. 9, no. 1, 1979, pp. 62-66.

[7] J. Piper, et al., "Automation of chromosome analysis," *Signal Processing 2*, 1980, pp. 203-221.

[8] M. Popescu, et al., "Automatic karyotyping of metaphase cells with overlapping chromosomes," *Computers in Biology & Medicine*, vol. 29, 1999, pp. 61-82.

[9] W. P. Sweeny, et al., "Classification of chromosomes using a probabilistic neural network," *Cytometry* 16, 1994, pp. 17-24.

[10] L. Vanderheydt, et al., "Two dimensional shape decomposition using fuzzy subset theory applied to automated chromosome analysis," *Pattern Recognition*, vol. 13, no. 2, 1981 pp. 147-157.