CELL RECOGNITION BASED ON PCA AND BAYESIAN CLASSIFICATION

[1]Saeid Sanei and [2]Tracey K.M. Lee

[1]Centre for Digital signal processing Research, King's College London, UK, saeid.sanei@kcl.ac.uk
[2]School of Electrical and Electronic Engineering, Singapore Polytechnic, Singapore, tlee@sp.edu.sg

ABSTRACT

Recognition of the blood cell types is of great importance due to its high clinical diagnostic applications. Here, a method based on PCA followed by Bayesian classification, for identification of blood cells has been introduced. We have modified the work by Turk and Pentland on face recognition and extended it to cell recognition. Their method uses the standard method of eigenvector selection. Also only monochrome images have been considered and the method is not tolerant enough to geometrical changes. Here the idea has been extended to colour patterns. We pre-processe the images adjusting their size and rotation, with fast methods. The eigencells are selected based on minimisation of similarities among various sets and finally a classifier identifies the cell types by looking at the three-fold intensity-colour information. This overcomes many problems in cell classification where either certain cells are recognised or some constraints such as geometrical variations are incorporated.

1. INTRODUCTION

An automated method for classification and recognition of all types of blood and bone marrow cells has always been required due to its crucial role in study of molecular biology and diagnosis of many diseases such as various types of leukaemia. Computerised blood cell morphology has been under research for about three decades [1]. The work sped up after development of fast PCs in the last decade but faced many constraints in using parametric methods [2,3]. In parametric techniques the researchers have focused on measurement and identification of various features such as area, eccentricity, compactness, area of central pallor (for red cells), nucleus position, number of nuclear lobes, nucleus-cytoplasm ratio and colour of nucleus and cytoplasm. However, almost all these methods are computationally very costly and more focused on detection of certain cells.

Currently mechanical systems are used in blood cell counts and Human Visual System (HVS) is the most reliable tool in diagnosis of blood abnormalities. However mechanical systems can hardly distinguish between various types of immature cells (blasts). Moreover only haematologists can diagnose the disease by careful and timely observation of the stained blood. A reliable automated cell counter based on advance pattern recognition and classification can significantly assist clinicians.

Turk and Pentland developed a computerized method for automatic recognition of human faces [4]. They estimated the principal components of the faces as their features. These features were not necessarily the physical features such as eyes and lips. Each face is characterized by a weighted sum of the eigenvectors. In that case, to recognize a particular face it is enough to compare its weighted sum of eigenfaces with those of known faces after the corresponding class is determined. Moghaddam, et al [5], has reported that a Bayesian classification of the faces can perform better subject to having known Gaussian prior distributions. This assumption is not always true especially when the data sets are not very large. In this paper the Bayesian rule classifies the eigencells rather than the cell images without forcing any constraint on the likelihood and prior distributions. The method is reasonably simple and accurate. In addition, it does not depend on measurement of the physical features and detailed geometry. Moreover, the program has been modified to work on three-fold data corresponding to colour information. Then the similarities between the eigenvectors of the patterns have been minimised by eliminating the most similar vectors. Finally a Bayesian classifier is used to classify the eigencells and we have incorporated intensity and colour in the final decision making process.

The data here are images of stained peripheral blood consisting of a number of different blood cells and blasts. Over 40 kinds of cells may appear in the samples of normal blood or bone marrow. The captured images may be in any arbitrary size or direction. Moreover, in the bone marrow the calls may be deformed. The cells may have from zero to more than three nuclei and can vary in all features stated at the beginning of this part. In this work we pre-process the images by first rescaling the input images. Second the individual cells are separated and rotated and finally three vectors for representing intensity and colours are defined.

2. PRE-PROCESSING

In general, for typical images of blood cells, the background, the cytoplasm and the nuclei of the cells have distinct intensity levels. So they can be easily separated based on two threshold levels in the histogram. To do that the histogram, $f(n)$ is smoothed by a large-window averaging. The gradient of the histogram, $\nabla f = \partial f / \partial n$, approximated by $\nabla f = f(n) - f(n-1)$ is calculated. $d(n) = \log((\nabla f)^2+1)$ presents nulls at minimum and maximum points of the histogram. Figure 1 shows $f(n)$ and $d(n)$ for a typical blood pattern. By ignoring

the first and the last minima corresponding to the beginning and end of the frame, The threshold points, *thr(k)*, will be

$$thr(k) = \arg_{n}(d(n) = 0),\ for\ k = 2,4 \qquad (1)$$

For the image of Figure 1 the two points are 54 and 150. By using the second threshold value, the cell images are converted to bi-level patterns for which the entire cell areas can be measured. All images are then rescaled based on this measure. The procedure is repeated for the images in the database and any new input image. By using both threshold values, the centre of mass for both cytoplasm and nucleus can be easily marked. The link between the two centres, pointing to the nucleus centre, represents the direction of the cell. Figure 2 clarifies the concept; the arrow shows the direction. Figure 3 illustrates the change in directions for some of the cells numbered from 1 to 4 in Figure 1.a.

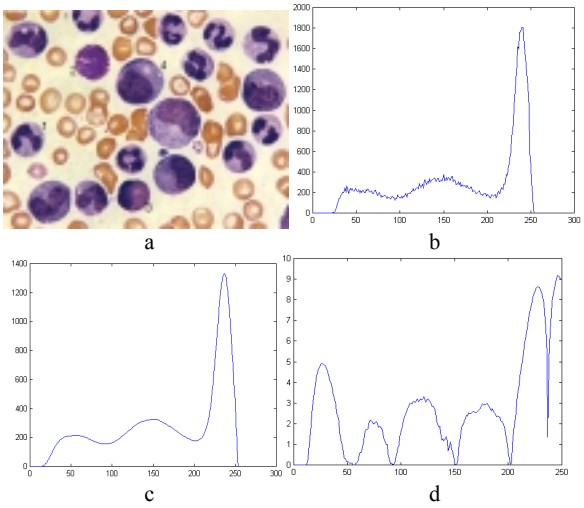

Figure 1. a. The image b. The original histogram, *f(n)*, c. The smoothed histogram d. *d(n)*

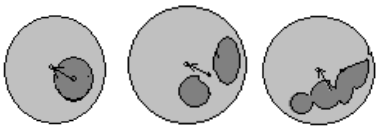

Figure 2 Identification of the cell direction

The cells are rotated using cubic spline interpolation, by the angle between the cell direction and the vertical line. Figure 2.b shows the rotated cells. Obviously such a simple criterion may not be used in rotation of faces.

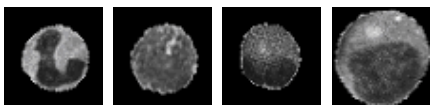

Figure 3. The new directions for the cells number 1 to 4 in Figure 1 respectively

Each cell is located inside a 39×39 pixel frame. Although larger sizes lead to better classification, it requires larger databases and more computation cost. Each frame, as an RGB image, is then integrated to its intensity, Y_m, and two chrominance components, I_m and Q_m, (m= cell number) [6].

This model is useful since the Y component provides the monochrome information; further, it exploits advantage of the HVS, in particular our sensitivity to luminance. Figure 4 shows the three components (4.c, 4.d and 4.e).

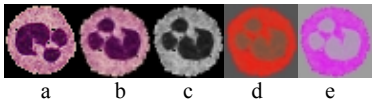

Figure 4. a. Selected original, b. Rotated, c. Y, d. I and e. Q components of a segmented neutrophil cell.

There may be some cells overlapping each other in the image. We have used a method based on watershed detection explained in [7] to segment the multi-lobe regions into the constituent cells. Figure 5 illustrate some of the cells in our database

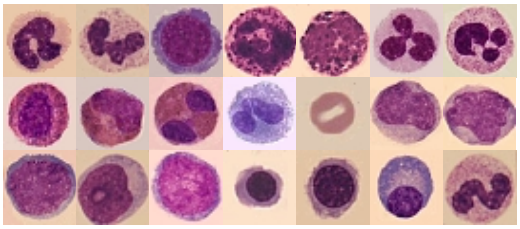

Figure 5 Samples of the cells in the database

3. PCA APPLICATION

Principal Component Analysis (PCA) has been a powerful tool in image decorrelation and compression since 1987 [8] [9]. A best set of eigenvectors also called eigenpictures, were used to approximate the corresponding images. As in eigenfaces [4], a cell image can be reconstructed by weighted sum of a small collection of such features called eigencells. A new cell may be classified the cells can be classified by comparison between the above features. This is summarized in the following steps.

1. Provision of an initial set of cell images
2. Separation of the cells into Y_m, I_m and Q_m components (training cells)
3. Calculation of the eigencells from the training set for each component
4. Selection of a number of dominating eigencells based on the similarity minimisation criterion. These images define the cell M-dimensional space. In case any new cell of certain type is encountered, its eigencells are recalculated.
5. Calculation of the corresponding distribution in M-dimensional weight space for each known cell by projecting their cell images onto the cell space

After the system is initialised, the following steps will classify new cell images.

1. Calculating the a set of weights based on the input cell image and the *M* eigencells by projecting the input image onto each of the eigencells.
2. Classification of the weight pattern and identification of the cell type.
3. If the weights are very different from those in the data base, label the cell as "Unknown"

3.1. Eigencell Computation

In PCA, we seek a set of *M* orthonormal vectors, $\mathbf{u}_n$, which best describes the distribution of a set of sample data [9]. The *k*th vector, $\mathbf{u}_k$, is chosen such that

$$\lambda_k = \frac{1}{M}\sum_{n=1}^{M}(\mathbf{u}_k^T \mathbf{\Phi}_n)^2 \qquad (2)$$

has its maximum value when

$$\mathbf{u}_l^T \mathbf{u}_k = \begin{cases} \boldsymbol{I} & \textit{for } l = k \\ \boldsymbol{0} & \textit{elsewhere} \end{cases} \qquad (3)$$

where $\mathbf{\Phi}_n$ is the difference between each cell and the average over one set, $\boldsymbol{I}$ is the unitary matrix and λ_k are the eigenvalues of the covariance matrix

$$C = \frac{1}{M}\sum_{n=1}^{M}\mathbf{\Phi}_n\mathbf{\Phi}_n^T = \boldsymbol{A}\boldsymbol{A}^T \qquad (4)$$

where $\boldsymbol{A} = \left[\mathbf{\Phi}_1, \mathbf{\Phi}_2, \mathbf{\Phi}_3, \ldots \mathbf{\Phi}_M\right]$. The matrix $\boldsymbol{C}$ is $N^2 \times N^2$ (here $N = 39$). Computation of the N^2 eigenvectors is very intensive. The undertaken number of vectors is reduced to $M << N^2$. Practically, we assign zero as eigenvalues for the remaining eigenvectors. Fortunately we can solve $\boldsymbol{C}$ for $N^2 \times N^2$ size eigenvectors initially by solving for the eigenvectors of $M \times M$ matrix e.g., 8×8 instead of $39^2 \times 39^2$. Then an appropriate linear combination of the cell images $\mathbf{\Phi}_i$, is encountered. Consider the eigenvectors $\mathbf{v}_i$ of A^TA such that $\boldsymbol{A}^T\boldsymbol{A}\mathbf{v}_i = \mu_i\mathbf{v}_i$. Multiplying both sides by $\boldsymbol{A}$, we have $\boldsymbol{A}\boldsymbol{A}^T\boldsymbol{A}\mathbf{v}_i = \mu i\boldsymbol{A}\mathbf{v}_i$ from which we see that $\boldsymbol{A}\mathbf{v}_i$ are the eigenvectors of $C = \boldsymbol{A}\boldsymbol{A}^T$. By knowing the eigenvectors $\mathbf{v}_l$ the eigencells are defined as

$$\mathbf{u}_l = \sum_{k=1}^{M}\mathbf{v}_{lk}\mathbf{\Phi}_k \quad l = 1,...,M \qquad (5)$$

C is $M \times M$ and the number of eigenvalues and eigenvectors in this case will be M. The number is even more reduced since we choose a smaller number of eigenvectors, in such a way to achieve minimum similarities between the sets.

3.2. Similarity Minimisation

Although the M vectors well describe the image, with them, there is no guarantee to present maximum-distance classes of eigencells for various images. In order to exploit this fact, initially we solve the eigenvectors for a bigger value of M'. Then we select those vectors for which

$$\gamma = \sum_{p,q=1}^{No.of\ images}\mathbf{\Phi}_p.\mathbf{\Phi}_q \qquad (6)$$

is minimum. p and q refer to the eigenvectors of the pth and qth cell images stored in the database. γ is measured for all the eigencells of the same cell. It is very similar to vector quantization problem for which a codebook with maximum distance between the codes (clusters), has to be designed. Different techniques can solve this problem. In our work we change the problem to a routing problem but unlike in communication, we try to find and sort the paths with higher link weights. Each path link will then have a weight equal to γ. The algorithm used is similar to the Viterbi algorithm. The difference is that it maximizes the sum of link weights among the vectors of different classes. Figure 6.a includes two vectors of size 3. To construct two new vectors where each consists of only two entries of above vectors we consider them as three layers (first one repeats at the end) and connect the elements of all the layers together and allocate weights to the links equal to the difference between the two corresponding nodes. Figure 6.b shows the concept.

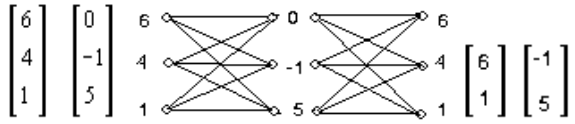

Figure 6 a. Original vectors, b. all possible paths, c. Final vectors

Sum of the link weights between the nodes is proportional to γ in equation 6. In this case for m vectors of n entries, the overall possible routes will be $(n)^{m+1}$. This is by knowing that the path starting from component k in the first column has to go back to the same component in the last column. In this example if these values are sorted and only two of them are selected, in fact the two with maximum distances (or less similarities) have been retained. Here the final two vectors are given in Figure 3.c. Although for a large number of vectors this computation is exhaustive, the result is optimum. In our experiment the number of initially selected eigenvalues is M=16 and the final number is only M'=8. We also recall that here, each link weight will be the sum of Euclidean distances between the two vector entries. Figure 5 represents the eight selected eigencells for one type of band neutrophils (The images have been rescaled for better visualization)

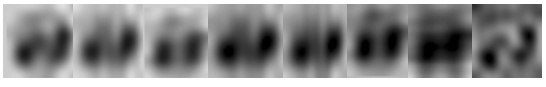

Figure 7 Eight eigencells belonging to a band neutrophil, the last cell in Figure 5

3.3. Classification of Images

To recognize a new cell, F_{new}, it needs to be transformed into its eigencell components by the linear transformation

$$\omega_k = \mathbf{u}_k^T(\mathbf{F}_{new} - \mathbf{F}_{ave}) \quad k = 1,2,\ldots,M' \qquad (7)$$

where F_{ave} is the pixel-by-pixel average of the cell images in the database. The weights ω_k, are the elements of $\mathbf{\Omega}^T = [\omega_1, \omega_2, \omega_k, \ldots \omega_{M'}]$ which shows the contribution of each eigencell in representing the input cell image. $\mathbf{\Omega}$ is then used by the algorithm to find which predefined cell classes best describe the input cell. Assume $\mathbf{\Omega}_k$ (cell classes) are calculated by averaging over the eigencells of each individual image. Using Turk method, a cell is classified as belonging to class l if $d_l = \left\|(\mathbf{\Omega} - \mathbf{\Omega}_l)\right\|^2$ is below a threshold level. Another variable is $\delta_l = \left\|(\mathbf{\Phi} - \mathbf{\Phi}')\right\|^2$ where $\mathbf{\Phi} = \mathbf{F}_{new} - \mathbf{F}_{ave}$ and $\mathbf{\Phi}' = \sum_{i=1}^{M''} \omega_i \mathbf{u}_i$ its projection onto the cell space. When the new eigencell is classified, the second parameter exactly specify the cell if δ_l is below a small threshold level. The procedure works well for most of the cases. But since it does not take into account the distribution of data, the performance can be improved. One of the most well known methods in vector classification is Fisher's linear discriminant [10]. In this method, the data in the classes classes are transformed using a linear operation. Then the best discriminant for classification of eigenvectors is used [11]. The real utility of Fisher's method is when the size of the feature vector is very large and the pdfs are close to Gaussian [12]. Although for large vector sizes the later assumption in most of the cases is acceptable, for small data sets the performance is not the same.

We include 15 classes of mature cells and blasts, excluding mature platelets, and an average of only four variations in our database. The cell patterns were selected from the haematology websites *haematological.pl, pathy.med.Nagoya-u.ac.jp, and bloodline.net*. For this small number of classes Fisher`s criterion does not necessarily perform well. In fact for non-Gaussian distributed data most of the classical classification methods do not perform well. Here, instead of comparing d_l and δ_l with fixed threshold levels, as in [4], a simple but efficient means of classification has been proposed based on posterior probability measurement. Assume that the *a priori* probability of class l is $p_l(k)$ and the class-conditional pdf of a cell belonging to class l be d_l as defined below. Therefore the *a posteriori* probability of a new eigencell $\mathbf{\Omega}$ belonging to class l will be $p(l \mid \mathbf{\Omega})$. Using Bayes' formula;

$$p(l_k \mid \mathbf{\Omega}) = \frac{p(\Omega \mid l_k)p_l(k)}{\sum_j p(\Omega \mid l_j)p_l(j)} = \frac{d_k \cdot p_l(k)}{\sum_j d_j \cdot p_l(j)} \qquad (8)$$

where $d_l = \left\|(\mathbf{\Omega} - \mathbf{\Omega}_l)\right\|^2$. Empirical results show that classifying by using posterior probability gives better results when compared to the previously mentioned methods. In using the three colour components *Y*, *I* and *Q*, we have to adjust for the colour based on the staining method. By considering these components to be independent (which may not be true for all the cases but it is a reasonable assumption) we can define a new factor as

$$r(l_k \mid \Omega) = \prod_{m=1}^{3} \alpha_m p_m(l_k \mid \Omega) \qquad (9)$$

where *m* refers to either Y, I or Q. α_m is a weight factor which has to be properly set for different staining method. The final decision is made based on

$$l_k = \arg\max_{l_k}(r(l_k \mid \Omega)) \qquad (10)$$

For Romanovsky stains, [13] we empirically set α_m at 0.58, 0.24 and 0.18 for Y, I and Q respectively. The results of fifteen major cells (mature and blasts) classification are been depicted in the confusion matrix of Table 1. It is seen that all the mature cells in normal blood are classified almost with 96.5% accuracy. Immature cells or blasts in the blood are also classified with more than 85% accuracy. However, in bone marrow, Myelocytes, Promyelocytes, Monocytes and Metamyelocyes are classified wrongly in about 21% of the cases. The reason is mainly because the cells in bone marrow can be deformed to any arbitrary shapes due to the environment pressure. We remember that for parametric methods the accuracy has usually been less than 70% for normal blood cell counts only. There has not been any report of complete blast count using image processing in the literature.

Table 1. The confusion matrix for various blood cells

Cell name	Classified No	Misclassified No / As
Basophil	15	0
Immature Basophil	13	2/ Immature Eosinophil
Eosinophil	15	0
Immature Eosinophil	14	1/ Immature Basophil
Segmented Neutrophil	11	4/ Band Neutrophil
Band Neutrophil	11	2/ Seg. Neutrophil 2/ Metamyelocite
Metamyelocyte	11	2/ Band neutrophil 2/ Immature Monocyte
Myelocyte	12	2/ Metamyelocyte 1/ Immature Monocyte
Monocyte	13	2/ Metamyelocyte
Immature Monocyte	15	0
Erythrocyte	14	1/ Polychromatic Eryth.
Polychromatic Erythrocyte	14	1/ Erythrocyte
Orthochromatic Normoblast	15	0
Polychromatic Normoblast	13	2/ Myelocyte
Lymphocyte	15	0

The result of the proposed method has been compared with the parametric feature detection and the original PCA method proposed by Pentland in Figure 8.

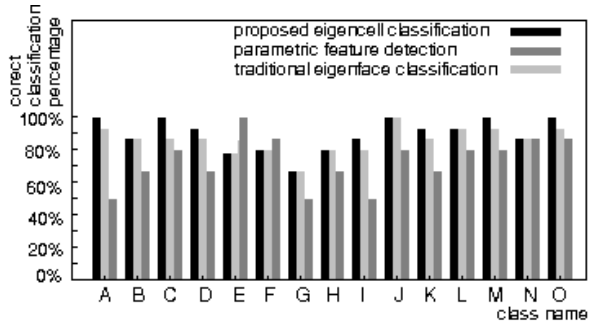

Figure 8 The comparison between the three cell recognition techniques. A to C are similar to those in Table 1.

4. SUMMARY AND CONCLUSIONS

In comparison with the parametric methods, the proposed technique presents a significant improvement in accurate and fast detection of the cell types. The method can also be applied to classification of a wide range of objects. The introduced strategy in minimisation of similarities among the eigencells and also the final eigencell classification method provide more improvement and attraction over previous algorithms. Multi-fold strategy avoids expanding the database when including luminance and chrominance at the same time. Although almost all mature and immature cells in the blood can be recognized using this technique, the application fails in classification of some of the cells in bone marrow where the blasts are deformed. The method may be improved to better classify the cells in bone marrow if a deformation compensation method such as morphing analysis, is applied prior to classification.

5. REFERENCES

[1] Lin, W. et al, "A computational intelligence system for cell classification.", ITAB98, *Proc.of IEEE Int. Conf. on Inf. Tech., applications to Biomedicine*, pp. 105-109, 1998

[2] http://sun16.cecs.missouri.edu/jpark/index.html

[3] Muthiah K. S, S. Sanei and S.H. Ong , "Application of CL-NN to blood cell morphology", *Proc. Of 10th int. conf. on biomed. Eng*., Singapore, pp. 630, 2000

[4] Turk M, and A. Pentland, "Eigenfaces for recognition", *Journal of Cognitive Neuroscience, Vol.3, No.1*, MIT, pp. 71-86, 1991.

[5] Moghaddam B, "Beyond eigenfaces: probabilistic matching for face recognition", Proc. Of IEEE, ICIP, pp.30-35, 1998.

[6] Smith A. R, "Colour Gamut transform pairs" *Computer Graphics, Vol. 12, No.3*, pp. 2-19, 1978.

[7] Sanei S, M Azron & S.H. Ong, "A fast hybrid SST-watershed segmentation of images", *Proc. of the ASTED int. conf. On Visualization, Imaging and Image processing,* Spain, pp. 444-447, 2001.

[8] Duda R. O., P.E. Hart and D.G. Stork, Pattern Classification, John Wiley, 2000.

[9] Jolliffe, I. T, Principal component analysis, Springer-Verlog, NY, 1986.

[10] Duda, R. O, D. G. Stock, P. E. Hart, Pattern Classification, John Wiley, 1999.

[11] Belhumeur, P. N, J.D. Hespanha and D.J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection, *IEEE Trans. On Pattern Analysis & Match. Vol. 19, No. 7*, pp. 711-720, 1997.

[12] Therrien, C. W, Decision estimation and classification: An introduction to pattern classification and related topics, John Wiley, 1989.

[13] Carr J. H, and B.F. Rodak, Clinical Haematology and Atlas, W.B. Saunders Co., 1998.