33rd Annual International Conference of the IEEE EMBS
Boston, Massachusetts USA, August 30 - September 3, 2011

Automatic Classification of Fish Germ Cells Through Optimum-Path Forest

João P. Papa, Mario E. M. Gutierrez, Rodrigo Y. M. Nakamura, Luciene P. Papa
Irene B. F. Vicentini, Carlos A. Vicentini

Abstract— **The spermatogenesis is crucial to the species reproduction, and its monitoring may shed light over some important information of such process. Thus, the germ cells quantification can provide useful tools to improve the reproduction cycle. In this paper, we present the first work that address this problem in fishes with machine learning techniques. We show here how to obtain high recognition accuracies in order to identify fish germ cells with several state-of-the-art supervised pattern recognition techniques.**

I. Introduction

The histomorphometric study is an approach for better understanding the spermatogenic processes and also the testicular function [1]. This evaluation allows the estimation of the spermatogenic efficiency for each specie, through, for example, the length of seminipherous tubules, frequency of germ cysts, spermatogonium generations, percentage and length of different germ cells, among others [2]. The use of morphology allow us to get deeper with cell kinetics and histophysiology of germ and Sertoli cells, being currently a valuable tool to the interpretation of the gonads in fishes [3].

However, one has no studies guided for automatic identification of such germ cells up do date. Based on this assumption, we propose in this work the automatic identification of fish germ cells using machine learning techniques. Henceforth, we focus on Optimum-Path Forest (OPF) [4], which is a recently developed technique that has demonstrated to be similar to Support Vector Machines (SVM) [5], but much faster for training.

The OPF classifier models the problem of pattern recognition as a problem of a graph partition into optimum-path trees (OPTs), which are rooted at a given set of key samples, and the optimality criterion is given by a smooth-path cost function [4]. Thus, depending on the way you build the graph and the adopted path-cost function, one can design a new OPF-based classifier.

The main contributions of this paper are threefold: (i) to present an automatic framework for fish germ cells classification, (ii) to introduce the OPF classifier for biological-based applications and (iii) to create a dataset composed by labeled germ cells. We would like to stress that, although we focus on fish cells, we can extend it to another kind of germ cells, like human ones. The remainder of this paper is organized as follows. Sections II and III present the OPF classifier and materials and methods, respectively. The experimental results are discussed in Section IV and conclusions are stated in Section V.

This work is supported by Capes and FAPESP Grants # 2009/16206-1 and # 2010/11676-7.

R. Nakamura and J. Papa are with Department of Computing, UNESP - Univ Estadual Paulista, Bauru, Brazil. {rodrigo.mizobe,papa}@fc.unesp.br

M. Gutierrez, I. Vicentini and C. Vicentini are with Department of Biological Sciences, UNESP - Univ Estadual Paulista, Bauru, Brazil. {memunozg,ibfv,carlosav}@fc.unesp.br

L. Papa is with Southwest Paulista College, Avaré, Brazil. lucienepapa@yahoo.com.br

II. Optimum-Path Forest

Let Z_1 and Z_2 be the training and test sets with $|Z_1|$ and $|Z_2|$ samples such as points or image elements (e.g., feature vectors, pixels, voxels, shapes and texture information). Let $\lambda(s)$ be the function that assigns the correct label i, $i = 1, 2, \ldots, c$, from class i to any sample $s \in Z_1 \cup Z_2$. Z_1 is a labeled set used to the design of the classifier and Z_2 is used to assess the performance of classifier and it is kept unseen during the project.

Let $S \subset Z_1$ be a set of prototypes of all classes (i.e., key samples that best represent the classes). Let v be an algorithm which extracts n attributes (color, shape or texture properties) from any sample $s \in Z_1 \cup Z_2$ and returns a vector $\vec{v}(s) \in \Re^n$. The distance $d(s,t)$ between two samples, s and t, is the one between their feature vectors $\vec{v}(s)$ and $\vec{v}(t)$. One can use any valid metric (e.g., Euclidean) or a more elaborated distance algorithm. Our problem consists of using S, (v,d) and Z_1 to project an optimal classifier which can predict the correct label $\lambda(s)$ of any sample $s \in Z_2$. The OPF classifier creates a discrete optimal partition of the feature space such that any sample $s \in Z_2$ can be classified according to this partition. This partition is an optimum path forest (OPF) computed in $\Re^n$ by the image foresting transform (IFT) algorithm [6].

Let (Z_1, A) be a complete graph whose the nodes are the samples in Z_1 and any pair of samples defines an arc in $A = Z_1 \times Z_1$. The arcs do not need to be stored and so the graph does not need to be explicitly represented. A path is a sequence of distinct samples $\pi = \langle s_1, s_2, \ldots, s_k \rangle$, where $(s_i, s_{i+1}) \in A$ for $1 \leq i \leq k-1$. A path is said *trivial* if $\pi = \langle s_1 \rangle$. We assign to each path π a cost $f(\pi)$ given by a path-cost function f. A path π is said optimum if $f(\pi) \leq f(\pi')$ for any other path π', where π and π' end at a same sample s_k. We also denote by $\pi \cdot \langle s,t \rangle$ the concatenation of a path π with terminus at s and an arc (s,t).

The OPF algorithm may be used with any *smooth* path-cost function which can group samples with similar properties [6]. We are interested in prototypes that fall in the region between classes, which are generally overlapped regions. So, we will address the path-cost function f_{max}, because of its

theoretical properties for estimating prototypes that have this behavior (Section II-A gives the details about this procedure):

$$f_{max}(\langle s \rangle) = \begin{cases} 0 & \text{if } s \in S, \\ +\infty & \text{otherwise} \end{cases}$$
$$f_{max}(\pi \cdot \langle s,t \rangle) = \max\{f_{max}(\pi), d(s,t)\}, \quad (1)$$

such that $f_{max}(\pi)$ computes the maximum distance between adjacent samples in π, when π is not a trivial path.

The OPF algorithm assigns one optimum path $P^*(s)$ from S to every sample $s \in Z_1$, forming an optimum path forest P (a function with no cycles which assigns to each $s \in Z_1 \backslash S$ its predecessor $P(s)$ in $P^*(s)$ or a marker nil when $s \in S$. Let $R(s) \in S$ be the root of $P^*(s)$ which can be reached from $P(s)$. The OPF algorithm computes for each $s \in Z_1$, the cost $C(s)$ of $P^*(s)$, the label $L(s) = \lambda(R(s))$, and the predecessor $P(s)$, as follows.

Algorithm 1: – OPF ALGORITHM

INPUT: *A λ-labeled training set Z_1, prototypes $S \subset Z_1$ and the pair (v,d) for feature vector and distance computations.*
OUTPUT: *Optimum-path forest P, cost map C and label map L.*
AUXILIARY: *Priority queue Q and cost variable cst.*

1. For each $s \in Z_1 \backslash S$, set $C(s) \leftarrow +\infty$.
2. For each $s \in S$, do
3. $\quad C(s) \leftarrow 0$, $P(s) \leftarrow nil$, $L(s) \leftarrow \lambda(s)$, and insert s in Q.
4. While Q is not empty, do
5. $\quad$ Remove from Q a sample s such that $C(s)$ is minimum.
6. $\quad$ For each $t \in Z_1$ such that $t \neq s$ and $C(t) > C(s)$, do
7. $\quad\quad$ Compute $cst \leftarrow \max\{C(s), d(s,t)\}$.
8. $\quad\quad$ If $cst < C(t)$, then
9. $\quad\quad\quad$ If $C(t) \neq +\infty$, then remove t from Q.
10. $\quad\quad\quad$ $P(t) \leftarrow s$, $L(t) \leftarrow L(s)$, $C(t) \leftarrow cst$
11. $\quad\quad\quad$ Insert t in Q.

Lines $1-3$ initialize maps and insert prototypes in Q. The main loop computes an optimum path from S to every sample s in a non-decreasing order of cost (Lines $4-10$). At each iteration, a path of minimum cost $C(s)$ is obtained in P when we remove its last node s from Q (Line 5). Ties are broken in Q using first-in-first-out policy. That is, when two optimum paths reach an ambiguous sample s with the same minimum cost, s is assigned to the first path that reached it. Note that $C(t) > C(s)$ in Line 6 is false when t has been removed from Q and, therefore, $C(t) \neq +\infty$ in Line 9 is true only when $t \in Q$. Lines $8-11$ evaluate if the path that reaches an adjacent node t through s is cheaper than the current path with terminus t and update the position of t in Q, $C(t)$, $L(t)$ and $P(t)$ accordingly.

A. Training

We say that S^* is an optimum set of prototypes when Algorithm 1 minimizes the classification errors for every $s \in Z_1$. S^* can be found by exploiting the theoretical relation between minimum-spanning tree (MST) and optimum-path tree for f_{max} [7]. The training essentially consists of finding S^* and an OPF classifier rooted at S^*.

By computing an MST in the complete graph (Z_1, A), we obtain a connected acyclic graph whose nodes are all samples of Z_1 and the arcs are undirected and weighted by the distances d between adjacent samples. The spanning tree is optimum in the sense that the sum of its arc weights is minimum as compared to any other spanning tree in the complete graph. In the MST, every pair of samples is connected by a single path which is optimum according to f_{max}. That is, the minimum-spanning tree contains one optimum-path tree for any selected root node.

The optimum prototypes are the closest elements of the MST with different labels in Z_1 (i.e., elements that fall in the frontier of the classes). By removing the arcs between different classes, their adjacent samples become prototypes in S^* and Algorithm 1 can compute an optimum-path forest with minimum classification errors in Z_1.

B. Classification

For any sample $t \in Z_2$, we consider all arcs connecting t with samples $s \in Z_1$, as though t were part of the training graph. Considering all possible paths from S^* to t, we find the optimum path $P^*(t)$ from S^* and label t with the class $\lambda(R(t))$ of its most strongly connected prototype $R(t) \in S^*$. This path can be identified incrementally, by evaluating the optimum cost $C(t)$ as

$$C(t) = \min\{\max\{C(s), d(s,t)\}\}, \ \forall s \in Z_1. \quad (2)$$

Let the node $s^* \in Z_1$ be the one that satisfies Equation 2 (i.e., the predecessor $P(t)$ in the optimum path $P^*(t)$). Given that $L(s^*) = \lambda(R(t))$, the classification simply assigns $L(s^*)$ as the class of t. An error occurs when $L(s^*) \neq \lambda(t)$.

III. MATERIALS AND METHODS

In this section we describe the proposed method to tackle the problem of automatic classification of fish germ cells. As aforementioned in Section I, one can assesses several crucial information about the germinative process in fishes by considering the amount of germ cells at the seminipherous tubules. Figure 1 displays an image obtained from the seminipherous tubule of *Leporinus macrocephalus*, a typical Brazilian fish. This image was obtained through an optical microscope with $40\times$ of magnification.

Thus, a specialist system to automatic quantify germ cells can be described by two stages: (i) cell segmentation and (ii) cell classification. In the former, possible germ cells may be identified for further classification in the latter step. In addition, a post-processing step after first stage may be incorporated to the system in order to remove non-cell objects, or even to refine the segmentation of true ones.

In this work, we are facing the problem of classifying the four main fish germ cell types as follows: (i) spermatogonium, (ii) spermatocyte, (iii) spermatid and (iv) sperm. Therefore, only the second step will be addressed here, since we already have ground truth images, which were previous segmented and labeled by a technician. It is important to shed light over that the automatic segmentation step concerns with our next work. Figure 2 displays some manually segmented examples.

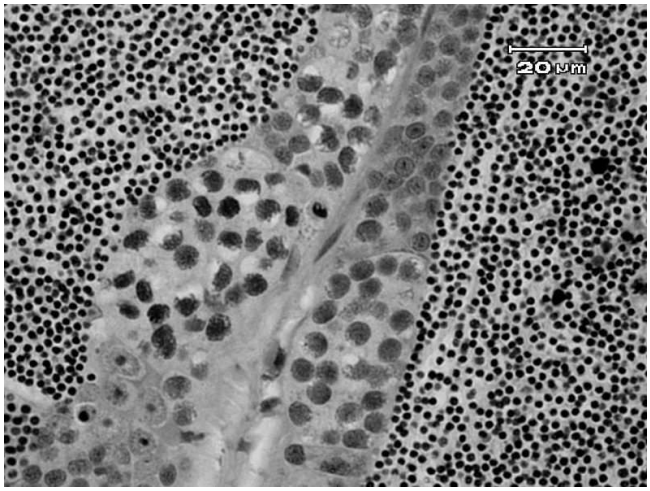

Fig. 1. An 8/bits image obtained from the seminipherous tubule of *Leporinus macrocephalus*.

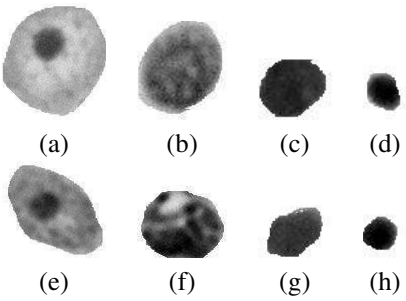

Fig. 2. Examples of (a) and (e) spermatogonium, (b) and (f) spermatocyte, (c) and (g) spermatid and (d) and (h) sperm.

The main feature that allow us to distinguish the different germ cells is their perimeter, since the cell division begins with the spermatogonium type and ends over with the sperm. Therefore, it is expected that the size of the cells will decrease whether we go deeper onto the evolution process among them. However, we may find some overlapping regarding the perimeter between spermatogonium and spermatocyte. Thus, we decided to apply texture features together with the perimeter information, since the spermatogonium cell is characterized by a dark gray nucleolus, as one can see in Figures 2a and 2e.

In order to extract texture information from these images, we applied the Gabor filter [8] only in the image foreground, i.e., in the object of interest, say that cell. The Gabor filter can be mathematically formulated as follows:

$$G(x, y, \theta, \gamma, \sigma, \lambda, \psi) = e^{\frac{x'^2 + y'^2 \sigma^2}{2\sigma^2}} \cos\left(2\pi \frac{x'}{\lambda} + \psi\right), \quad (3)$$

where $x' = x\cos(\theta) + y\sin(\theta)$ and $y' = x\sin(\theta) + y\cos(\theta)$. In the above equation, λ means the sinusoidal factor, θ represents the orientation angle, ψ is the phase offset, σ is the Gaussian standard deviation and γ is the aspect spatial ratio.

The main idea of Gabor filter is to perform a convolution between the original image I and $G_{\theta,\gamma,\sigma,\lambda,\psi}$ in order to obtain a Gabor-filtered representation, as follows:

$$\hat{I}_{\theta,\gamma,\sigma,\lambda,\psi} = I * G_{\theta,\gamma,\sigma,\lambda,\psi}, \quad (4)$$

in which $\hat{I}_{\theta,\gamma,\sigma,\lambda,\psi}$ denotes the filtered image. Thus, one can obtain a filter bank of Gabor filtered images by varying its parameters. In this work we used a convolution filter of size 3×3 with the following Gabor parameters:

- 6 different orientations: $\theta = 0°$, $45°$, $90°$, $135°$, $225°$ and $315°$;
- 3 spatial resolutions: $\lambda = 2.5$, 3 and 3.5. Notice that, for each one of λ values, we applied different values for σ, say that $\sigma = 1.96$, 1.40 and 1.68;
- $\psi = 0$ and
- $\gamma = 1$.

Recall that these values were empirically chosen and based on our previous experience.

Once we get the Gabor-filtered images (one can see that we have $6 \times 3 = 18$ images), we then compute the energy ϵ of them, given by

$$\epsilon_{\theta,\gamma,\sigma,\lambda,\psi} = \sqrt{\sum_{x,y} I^2_{\theta,\gamma,\sigma,\lambda,\psi}(x, y)}, \quad (5)$$

in which $\epsilon_{\theta,\gamma,\sigma,\lambda,\psi}$ denotes the energy at image $\hat{I}_{\theta,\gamma,\sigma,\lambda,\psi}$.

Thus, each image is described by 19 features, being 18 of them related with texture and the remaining one is the perimeter. The whole proposed procedure for feature extraction is described by Figure 3.

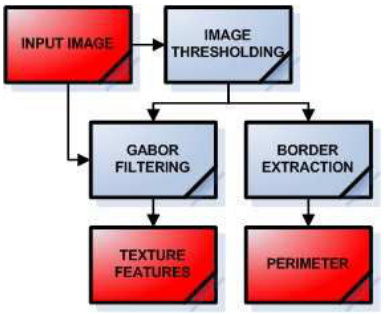

Fig. 3. Pipeline of the proposed method for feature extraction.

Figure 4 displays the proposed pipeline with a spermatogonium cell. The thresholding step was performed by Otsu's method [9].

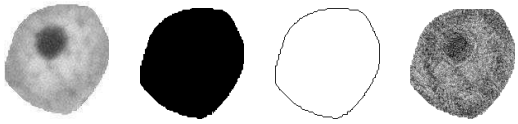

Fig. 4. Proposed pipeline: (a) Original image, (b) thresholded image, (c) extracted border, (d) Gabor-filtered image using $\lambda = 3.5$, $\theta = 315°$, $\sigma = 1.96$, $\gamma = 1$ and $\psi = 0$.

The dataset we used is composed by 80 images equally distributed in 4 classes, i.e., we have 20 images for each

germ cell type: spermatogonium, spermatocyte, spermatid and sperm. All images were obtained from *Leporinus macrocephalus* fish.

Regarding classifiers, we used OPF, SVM with Radial Basis Function as kernel mapping (SVM-RBF), SVM with linear kernel mapping (SVM-LINEAR), SVM without kernel mapping (SVM-NOKERNEL) and Self Organizing Maps (SOM) [10]. For OPF implementation we used the LibOPF package [11], and with respect to SVM classifiers we applied LibSVM [12] for SVM-RBF and SVM-LINEAR, and LibLINEAR [13] for SVM-NOKERNEL. Finally, for SOM networks we used our own implementation. Recall that SVM parameters were optimized through cross-validation and for SOM we used a 100×100 neural lattice with 10 iterations for learning.

IV. Experimental Results

In this section we discuss the experimental results using the classifiers highlighted in the previous section. We conducted the experiments using 50% of the dataset for training and the remaining 50% for testing, which were repeated 10 times with randomly generated training and test sets. Table I displays the results.

Classifier	Accuracy	Training time [s]	Classification time [s]
OPF	98.33±0.000008	0.0002	0.0001
SVM-RBF	98.33±0.000008	1.34	0.004
SOM	96.66±0.000008	0.004	0.00006
SVM-LINEAR	98.338±0.000008	0.88	0.004
SVM-NOKERNEL	95.00±0.00	2.46	0.004

TABLE I

MEAN ACCURACY AND EXECUTION TIMES (TRAINING AND TESTING) AFTER 10 RUNNINGS.

One can see that OPF, SVM-RBF and SVM-LINEAR achieved similar results, but OPF was 6700 times faster than SVM-RBF and 4400 times faster than SVM-LINEAR for training. Regarding the classification time, OPF was 40 times faster than SVM-RBF and SVM-LINEAR.

Note that all classifiers achieved good results, which can demonstrate the effectiveness of the proposed methodology to classify germ cells. In order to analyze the OPF misclassification errors, we compute the confusion matrix from a randomly selected execution. Table II shows this information.

Label	1	2	3	4
1	9	1	0	0
2	0	10	0	0
3	0	0	10	0
4	0	0	0	10

TABLE II

CONFUSION MATRIX.

As one can see, the only misclassification was due to labels 1 and 2, i.e., spermatogonium and spermatocyte. This is probably because of the similar perimeters of the images and also the spermatogonium nucleolus might be brighter than usual.

V. Conclusions

The monitoring of the spermatogenic cycle is very important to retain information about the reproduction of the species, as well as to develop methods to deal with possible problems of that.

In this paper, we address the problem of automatic classification of germ cells, and we validate our approach for a Brazilian typical fish called *Leporinus macrocephalus*. A dataset of labeled germ cells was built in order to accomplish with this task.

We propose to extract Gabor-based texture features and to use them together with the perimeter of the cells to automatic identify four types of germ cells, say that spermatogonium, spermatocyte, spermatid and sperm. For that, we used five state-of-the-art supervised pattern recognition techniques, in which OPF, SVM-RBF and SVM-LINEAR achieved similar results, with OPF much faster for training and classification. As far as we know, we are the first to propose an automatic classification methodology for fish germ cells, and also to introduce OPF in the context of biological-based applications.

References

[1] R. W. Schulz, L. R. França, J. J. Lareyre, F. LeGac, H. Chiarini-Garcia, R. H. Nóbrega, and T. Miura, "Spermatogenesis in fish," *General and comparative endocrinology*, vol. 165, pp. 390–411, 2009.

[2] R. W. Schulz, S. Menting, J. Bogerd, França, Luiz. R., D. A. R. Vilela, and H. P Godinho, "Sertoli cell proliferation in the adult testis-evidence from two fish species belonging to different orders," *Biology of reproduction*, vol. 73, pp. 891–898, 2005.

[3] R. H. Nóbrega, "Capacidade suporte das células de sertoli e alterações do epitélio germinativo masculino e do tecido intersticial durante o ciclo reprodutivo de peixes neotropicais," 2003, Monografia para obtenção do título de Bacharel em Ciências Biológicas. Departamento de Morfologia do Instituto de Biociências/UNESP- Botucatu, São Paulo, Brasil.

[4] J. P. Papa, A. X. Falcão, and C. T. N. Suzuki, "Supervised pattern classification based on optimum-path forest," *International Journal of Imaging Systems and Technology*, vol. 19, no. 2, pp. 120–131, 2009.

[5] Corinna Cortes and Vladimir Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, pp. 273–297, 1995.

[6] A.X. Falcão, J. Stolfi, and R.A. Lotufo, "The image foresting transform theory, algorithms, and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 1, pp. 19–29, 2004.

[7] C. Allène, J.-Y. Audibert, M. Couprie, J. Cousty, and R. Keriven, "Some links between min-cuts, optimal spanning forests and watersheds," in *Proceedings of the International Symposium on Mathematical Morphology*, São José dos Campos, 2007, vol. 1, pp. 253–264, Instituto Nacional de Pesquisas Espaciais (INPE).

[8] Hans G. Feichtinger and T. Strohmer, Eds., *Gabor Analysis and Algorithms: Theory and Applications*, Birkhauser Boston, 1st edition, 1997.

[9] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man and Cybernet*, vol. 9, no. 1, pp. 62–69, 1979.

[10] Teuvo Kohonen, Ed., *Self-organizing maps*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1997.

[11] J.P. Papa, C.T.N. Suzuki, and A. X. Falcão, *LibOPF: A library for the design of optimum-path forest classifiers*, 2009, Software version 2.0 available at `http://www.ic.unicamp.br/~afalcao/LibOPF`.

[12] C. C. Chang and C. J. Lin, *LIBSVM: A Library for Support Vector Machines*, 2001, Software available at url http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[13] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.