FAST DETECTION OF CELLS USING A CONTINUOUSLY SCALABLE MEXICAN-HAT-LIKE TEMPLATE

K. N. Chaudhury[⋆]*, Zs. Püspöki*[⋆]*, A. Muñoz-Barrutia*[♯]*, D. Sage*[⋆]*, M. Unser*[⋆]

[⋆]Biomedical Imaging Group, Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland
[♯] Center for Applied Medical Research, University of Navarra, Pamplona, Spain

ABSTRACT

We propose a fast algorithm for the detection of cells in fluorescence images. The algorithm, which estimates the number of cells and their respective centers and radii, relies on the fast computation of intensity-based correlations between the cells and a near-isotropic Mexican-hat-like detector. The attractive features of our algorithm are its speed and accuracy. The former attribute is derived from the fact that we can compute correlations between a cell and detectors of various sizes using $O(1)$ operations; whereas, it is our ability to continuously control the center and the radius of the detector that results in a precise estimate of the position and size of the cell. We provide experimental results on both simulated and real data to demonstrate the speed and accuracy of the algorithm.

Index Terms— Fluorescence microscopy; cell detection; box spline; Mexican-hat; LoG; scalable detector.

1. INTRODUCTION

The segmentation of cells is often crucial for the quantitative analysis of microscopic images. Operations such as the counting of cells, study of their spatial organization, and the distribution of fluorescence signals on the cell nuclei require a precise delineation of the cell boundaries.

In fluorescence imaging, the cells (or the cell nuclei) appear as bright blobs on a dark background. The difficulty, however, is that the images are often corrupted with large amounts of noise owing to the limited laser excitation used to avoid excess photobleaching [1]. Added to this, there is typically the problem of uneven illumination, where the intensities within the cell are significantly varying, making it difficult to segment the cell using a single global threshold. Adaptive thresholding methods, often combined with region growing, have commonly been used to circumvent this problem [2, 3]. More recently, accurate and sophisticated segmentation algorithms based on level-sets and graph-cuts have been proposed for this task [4, 5]. The present limitations of such methods are (i) the difficulty to automate them, and (ii) their slow-to-moderate speed of computation. For certain applications, especially those related to high-throughput screening, it is absolutely necessary to adopt techniques that are fast and fully automated. Simple but efficient detection methods have been proposed to suit these requirements [6, 7].

In this paper, we propose a simple and fast linear filtering based algorithm for the detection of round cells in fluorescence images. The core of our detection paradigm, the fast computation of correlations between the cell and the scalable Mexican-hat detector, is based on an algorithm proposed in [8] for space-variant smoothing of images using certain Gaussian-like box splines, the *radially-uniform box splines*. In the present context, we are interested in the quasi-isotropic variant of the radially-uniform box splines which resemble the isotropic Gaussians. Akin to the popular means of approximating the Mexican-hat using the difference of Gaussians, we realize Mexican-hat-like detectors using the difference of two such near-isotropic box splines. These are then used as templates for detecting round cells/nuclei in fluorescence images, and for estimating their positions and radii.

This work was funded (in part) by the Swiss National Science Foundation, under Grant 200020-121763, and the Ramon y Cajal program of the Spanish Ministry of Education.

2. SCALABLE MEXICAN-HAT DETECTOR

To introduce the notations, we recall the construction of the isotropic variants of the radially-uniform box splines introduced in [8, 9]. The detector construction is discussed next.

2.1. Gaussian-like box spline

The particular quasi-isotropic box splines that we are interested in are realized through the convolution of four "uniformly-rotated" box functions of identical width. More specifically, let $\mathrm{Box}_r(x) = 1/r$ for $|x| \leq r/2$, and equals 0 otherwise, be the (normalized) one-dimensional box function of width r, and let $\boldsymbol{u}_\theta = (\cos\theta, \sin\theta)$ and $\boldsymbol{u}_{\theta_\perp} = (-\sin\theta, \cos\theta)$ be the unit vectors along direction θ and perpendicular to it. Then the bivariate box spline $\beta_r(\boldsymbol{x})$ under consideration is specified by

$$\beta_r(\boldsymbol{x}) = (\varphi_{r,0} * \varphi_{r,\pi/4} * \varphi_{r,\pi/2} * \varphi_{r,3\pi/4})(\boldsymbol{x}),$$

where $\varphi_{r,\theta}(\boldsymbol{x}) = \mathrm{Box}_r(\boldsymbol{u}_\theta^T \boldsymbol{x})\delta(\boldsymbol{u}_{\theta_\perp}^T \boldsymbol{x})$ is obtained by rotating the tensor $\mathrm{Box}_r(x_1)\delta(x_2)$ through an angle θ. This, in fact, can be seen as an improvement over the standard separable construction

$$\phi_r(\boldsymbol{x}) = (\varphi_{r,0} * \varphi_{r,0} * \varphi_{r,\pi/2} * \varphi_{r,\pi/2})(\boldsymbol{x})$$

which uses box functions solely along the horizontal and the vertical directions. Although $\beta_r(\boldsymbol{x})$ and $\phi_r(\boldsymbol{x})$ have the same order–a total of four box functions in each case–the "rounding-effect" of the box functions placed along the diagonals tends to make the former more isotropic. Indeed, judging by the shape of the support and the distribution of the intensity of $\beta_r(\boldsymbol{x})$ and $\phi_r(\boldsymbol{x})$, as shown in figure 1, $\beta_r(\boldsymbol{x})$ clearly looks more Gaussian-like than $\phi_r(\boldsymbol{x})$. A quantitative justification of this fact can be obtained through the computation of the following *isotropy index*

$$\varrho = \frac{1}{2\pi||f||^2} \int_0^{2\pi} \langle R_\theta f, f \rangle \, d\theta,$$

which measures the rotational symmetry of a non-negative function $f(\boldsymbol{x})$ by correlating it with its rotated versions (R_θ is the rotation operator). The isotropy index of $\phi_r(\boldsymbol{x})$ was found to be 98.8%, while a higher index of 99.7% was recorded for $\beta_r(\boldsymbol{x})$.

2.2. Detector specification and characteristics

The Laplacian-of-a-Gaussian (LoG), also known as the Mexican-hat, is widely used for detecting radial singularities in images. In practice, the LoG is often approximated by the Difference-of-Gaussians (DoG), which is easier to implement. In our case,

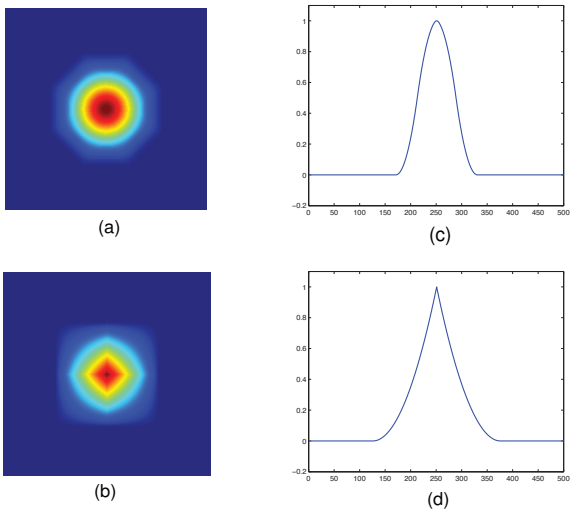

Fig. 1. Intensity distributions of (a) $\beta_r(\boldsymbol{x})$ and (b) $\phi_r(\boldsymbol{x})$; (c) and (d): Respective scan profiles along $\pi/8$.

we approximate the Mexican-hat detector as the difference of a narrow and a wide box spline:

$$\psi_r(\boldsymbol{x}) = \beta_r(\boldsymbol{x}) - \beta_{\sqrt{2}r}(\boldsymbol{x}), \tag{1}$$

where $r > 0$ is a real-valued parameter that controls the scale (or radius) of the detector. Figure 2 shows the intensity distribution of $\psi_r(\boldsymbol{x})$ and the profile along a scan-line passing through the origin.

We note that the total mass of $\beta_r(\boldsymbol{x})$ is independent of r, so that

$$\int \psi_r(\boldsymbol{x}) d\boldsymbol{x} = 0$$

In particular, while the detector tends to suppress uniform-intensity regions, a large response is generated by singularities along the radial direction. To make this precise, we consider the ideal blob function

$$b_R(\boldsymbol{x};\boldsymbol{x}_c) = \begin{cases} 1 & \text{for } 0 \le \|\boldsymbol{x}-\boldsymbol{x}_c\| \le R, \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

with centre $\boldsymbol{x}_c$ and radius R. The radial singularity (jump) in this case is along the circle $\|\boldsymbol{x}-\boldsymbol{x}_c\| = R$. The corresponding detector response

$$M_r(\boldsymbol{x}_c) = \int b_R(\boldsymbol{x};\boldsymbol{x}_c)\psi_r(\boldsymbol{x}-\boldsymbol{x}_c)d\boldsymbol{x}$$

turns out to be a smooth unimodal function with a peak at $r = \gamma R$, where $\gamma = 1.33$ is a calibration factor. The unimodal nature is due to the cancellation that takes place between the constituent box splines when $r \le \gamma R$; the cancellation keeps increasing as r goes from γR to zero resulting in the progressive drop in the response. On the other hand, the decay of $M_r(\boldsymbol{x}_c)$ for $r > \gamma R$, when no cancellation occurs, comes as a consequence of the $1/r^4$ normalization of the box spline.

Though (2) is a rather idealized model for representing the cell/nucleus in real images, the responses obtained turn out to be very similar. For instance, figure 3 shows the response obtained for one of the cells in the fluorescence image shown in figure 5.

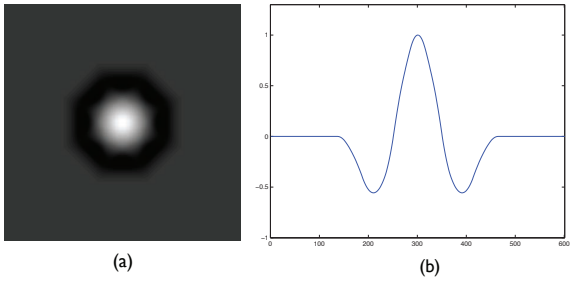

Fig. 2. (a) Intensity distribution of the Mexican-hat-like detector $\psi_r(\boldsymbol{x})$; (b) Scan profile along $\pi/8$.

2.3. Computational aspects

Before presenting the algorithm, we elaborate on the computation of the quantity

$$M_r(\boldsymbol{x}) = \int f(\boldsymbol{y})\psi_r(\boldsymbol{y}-\boldsymbol{x})d\boldsymbol{y} \tag{3}$$

for a given discrete image $f(\boldsymbol{x}) = \sum_{\boldsymbol{n}\in\mathbb{Z}^2} f[\boldsymbol{n}]\delta(\boldsymbol{x}-\boldsymbol{n})$. A fast and efficient evaluation of (3) will turn out to be the workhorse of our cell detection algorithm, which requires the rapid evaluation of $M_r(\boldsymbol{x})$ at several image positions and at different radii per position.

From (1), it is clear that $M_r(\boldsymbol{x})$ can be expressed as the difference of two box-spline averages, namely as

$$M_r(\boldsymbol{x}) = A_r(\boldsymbol{x}) - A_{\sqrt{2}r}(\boldsymbol{x})$$

where

$$A_r(\boldsymbol{x}) = \int f(\boldsymbol{y})\beta_r(\boldsymbol{y}-\boldsymbol{x})d\boldsymbol{y}.$$

It was shown in [8] that, by taking advantage of the quasi-separable structure of $\beta_r(\boldsymbol{x})$, one could exactly determine $A_r(\boldsymbol{x})$ using the following fast algorithm.

(1) (**Pre-integration**) We pre-integrate the discrete image $f[\boldsymbol{n}] = f[n_1, n_2]$ along the four cardinal directions using running-sums, which is efficiently implemented using the following recursion:

(i) Horizontal, $R_0[n_1,n_2] = \sum_{k=0}^{\infty} f[n_1-k, n_2]$.
(ii) First-diagonal, $R_{\pi/4}[n_1,n_2] = \sqrt{2}\sum_{k=0}^{\infty} R_0[n_1-k, n_2-k]$.
(iii) Vertical, $R_{\pi/2}[n_1,n_2] = \sum_{k=0}^{\infty} R_{\pi/4}[n_1, n_2-k]$.
(iv) Second-diagonal, $F[n_1,n_2] = \sqrt{2}\sum_{k=0}^{\infty} R_{\pi/2}[n_1+k, n_2-k]$.

(2) (**Finite-difference**) For a given position $\boldsymbol{x}$ and radius r, we compute $A_r(\boldsymbol{x})$ by taking a finite-difference of a continuous form of the pre-integrated image. In particular, we use the formula

$$A_r(\boldsymbol{x}) = \sum_{i=0}^{15} w_i F_{\text{int}}(\boldsymbol{x} + \boldsymbol{\tau} - \boldsymbol{x}_i)$$

where the notations are as follows: $w_i = (-1)^{q_1}\cdots(-1)^{q_4}(1/r^4)$ and $\boldsymbol{x}_i = r(q_1\boldsymbol{u}_0 + q_2\boldsymbol{u}_{\pi/2} + q_3\boldsymbol{u}_{3\pi/4} + q_4\boldsymbol{u}_{\pi/4})$ are the weights and positions of the FD mesh, the tuple (q_4,q_3,q_2,q_1), running from $(0,0,0,0)$ to $(1,1,1,1)$, is the binary representation of the indices $0 \le i \le 15$; $\boldsymbol{\tau} = (\tau_1,\tau_2)$ where $\tau_1 = (r-1)/2$ and $\tau_2 = (\sqrt{2}r + r - 3)/2$; and

$$F_{\text{int}}(\boldsymbol{x}) = \sum F[\boldsymbol{n}]\beta_{\text{ZP}}(\boldsymbol{x}-\boldsymbol{n})$$

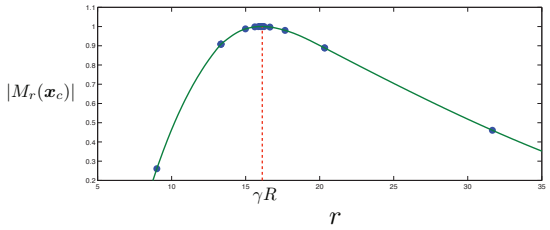

Fig. 3. A typical detector characteristic for one of the cells in Fig. 5. The solid curve shows the variation of the detector response $M_r(\boldsymbol{x}_c)$ with radius r, where $\boldsymbol{x}_c$ is the exact cell center. The dots on this curve correspond to the successive estimates of the optimal point γR^* obtained from the golden section search.

is the interpolation of $F[\boldsymbol{n}]$ using the box spline $\beta_{\text{ZP}}(\boldsymbol{x})$. The box spline $2\beta_{\text{ZP}}(\boldsymbol{x})$ is popularly known as the Zwart-Powell (ZP) element in box spline literature [10]. The samples $F_{\text{int}}(\boldsymbol{x}+\boldsymbol{\tau}-\boldsymbol{x}_i)$ are rapidly evaluated using a method that takes advantage of the finite support and the piecewise-quadratic structure of $\beta_{\text{ZP}}(\boldsymbol{x})$.

Once the pre-integration is over, $M_r(\boldsymbol{x})$ can thus be computed using a single FD mesh obtained by combining the two FD meshes of radii r and $\sqrt{2}r$ in step 2. The crucial aspect of the above computation is that number of operations (modulo the mild interpolations) required in step 2 is independent of r, so that the cost of computing $M_r(\boldsymbol{x})$ does not vary with the scale of the detector. This $O(1)$ complexity is clearly a significant improvement over a naive implementation of (3) involving the discretization of the Mexican-hat and the integral, which would require $O(r^2)$ computations.

3. FAST DETECTION OF CELLS

We model the image $f(\boldsymbol{x})$ as the superposition of circular blobs $b_{R_i}(\boldsymbol{x};\boldsymbol{x}_i)$ of unknown intensities α_i, along with background noise $n(\boldsymbol{x})$:

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i b_{R_i}(\boldsymbol{x};\boldsymbol{x}_i) + n(\boldsymbol{x}). \tag{4}$$

We consider a more realistic model of the blobs $b_{R_i}(\boldsymbol{x};\boldsymbol{x}_i)$ than the one in (2), whereby we do assume each cell to be localized within a disk of radius R_i, but we do not assume the blobs to have the same height or uniform intensity distributions.

The proposed algorithm estimates (without a priori knowledge) the number of cells (N), the centre (x_i) and the radius (R_i) of each cell. The only assumption used is that the radii of the cells are bounded, that is, $R_{\min} \le R_i \le R_{\max}$, where $R_{\min}$ and $R_{\max}$ are provided as user-inputs. Our approach involves the joint-estimation of the centers and the radii, whereby we first sample the values of $M_r(\boldsymbol{x})$ at discrete image positions and radii to obtain a coarse-to-fine estimate of the center $\boldsymbol{x}_i$, and then proceed to derive a fine estimate of R by the optimizing $|M_r(\boldsymbol{x}_i)|$. The main steps of the algorithm are as follows:

(1) **Coarse estimation of the centers, rejection of background points**: To obtain a coarse estimate of the centers by restricting the potential cell centers, aka the *candidate points*, to a lattice of resolution $\lceil 1.8R_{\min}\rceil \times \lceil 1.8R_{\min}\rceil$. Let us denote these candidate points by $\hat{\boldsymbol{x}}_1,\ldots,\hat{\boldsymbol{x}}_P$; in general, $P >> N$ but is small compared to the size of the image. The particular choice of resolution ensures that at least one lattice point intersects every cell, the smallest cells of interest in particular.

At every $\hat{\boldsymbol{x}}_i$, we compute $M_r(\hat{\boldsymbol{x}}_i)$ at $r_k = R_{\min} + k(R_{\max}-R_{\min})/4$, for $k = 1, 2$, and 3, and set $M_i = \max_k |M_{r_k}(\hat{\boldsymbol{x}}_i)|$. We then remove those $\hat{\boldsymbol{x}}_i$ (typically the background pixels) from the list of candidates where M_i is smaller than a specific threshold ε, and sort the remaining points as $\hat{\boldsymbol{x}}_1,\ldots,\hat{\boldsymbol{x}}_K$ ($K << P$) using the criterion that $\hat{\boldsymbol{x}}_j$ comes before $\hat{\boldsymbol{x}}_k$ if and only if $M_j \ge M_k$. Those $\hat{\boldsymbol{x}}_i$ that are close to the actual centers $\boldsymbol{x}_i$ tend to generate larger responses than those that are further off; the above ordering places such points in the foremost part of the candidate list.

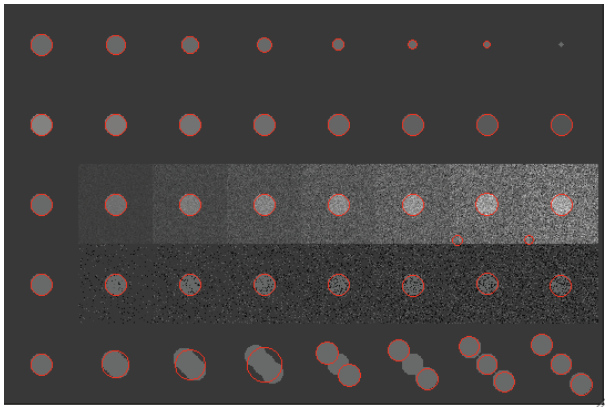

Fig. 4. Detection result on simulated data. **Row** 1: Blobs of different size; **Row** 2: Blobs of different intensities; **Row** 3: Identical blobs corrupted with different levels of additive Gaussian noise; **Row** 4: Identical blobs corrupted with different levels of speckle noise; and **Row** 5: Cluster of identical blobs with varying margin of separation.

(2) **Fine estimation of the center and the radius**: We set $N = 0$ and $i = 1$. We visit the foremost candidate point $\hat{\boldsymbol{x}}_1$, place an appropriate window W around it (at original resolution), and similar to the coarse estimation phase, we compute the maximum response $M(\xi) = \max_k |M_{r_k}(\xi)|$ at every $\xi \in W$. We use the result of this fine search to select the centre $\boldsymbol{x}_i$ as the point corresponding to the local maxima, that is,

$$\boldsymbol{x}_i = \arg\max_{\xi\in W} M(\xi).$$

Having estimated the centre, we use the unimodal characteristics of the detector (see Fig. 3) and the hypothesis that $R_{\min} \le R_i \le R_{\max}$ to estimate the radius as

$$R_i = \gamma^{-1} \arg\max \left\{ |M_r(\boldsymbol{x}_i)| : \gamma R_{\min} < r < \gamma R_{\max} \right\}.$$

We perform this optimization using the *golden section search*, which is an efficient algorithm for finding the extremum of a unimodal function where one localizes the extremum by successively shrinking the size of the interval within which this is known to exist.

This gives us the i-th cell with centre $\boldsymbol{x}_i$ and radius R_i, and we increment N to $N+1$. We then proceed to remove the candidate points $\hat{\boldsymbol{x}}_k$ belonging to the region of the detected cell, that is, the region $\{\boldsymbol{x} : \|\boldsymbol{x}-\boldsymbol{x}_i\| \le R_i\}$. We then set $K = K'$, where K' is the number of retained candidate points, and form the new candidate list $\hat{\boldsymbol{x}}_1,\ldots,\hat{\boldsymbol{x}}_K$. We increment i to $i+1$ and keep repeat the above process, namely the estimation of the centre and radius of the $(i+1)$-th cell, and the crucial speed-up step involving the truncation of the candidate list.

(3) **Convergence**: The iteration is terminated when either (i) the list of candidate points is exhausted, or (ii) the maximum projection goes below ε. The latter typically occurs when all the bright cells have been detected, leaving behind the weak-intensity cells and the background pixels.

4. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed algorithm, we simulated a single 800×400 image (cf. Fig. 4) using different variations of the cell model in (4). The idea was to study the effect of the following on the performance of the algorithm: (i) the size and intensity of the blobs (rows 1 and 2), (ii) the presence of ambient noise (rows 3 and 4), and (iii) the separation between the cells (row 5). The SNRs (dB) of the images in row 3 (Gaussian noise) and row 4 (speckle noise) were $\infty, 37.5, 31.5, 27.8, 25.4, 23.4, 21.8, 20.6,$ and $+\infty, 40.1, 37.2, 35.8, 34.6, 33.7, 33.3, 32.5$, respectively.

The algorithm was implemented in JAVA on a Macintosh 2.8 GHz Intel dual-core system. We set $R_{\min} = 3, R_{\max} = 20$, and $W = 3R_{\min}$. The execution time was about 0.5 seconds, and the detected blobs are shown in Fig. 4 using red circles. A near-exact estimate of the positions and radii was obtained for the blobs in the first four rows; the error in localization was within 1 pixel, and that for the radius was within 1%. There were however two false detections in presence of noise, the algorithm failed to detect the smallest blob in row 1and the overlapping blobs in row 5; the rest of the contiguous blobs in this row 5 were detected as a single object.

Figure 5 shows the results of our detection algorithm applied to real fluorescence images. The top figure shows a NIH3T3 cell line (mammalian cells for circadian cycle analysis) of size 512×512, which stably expressed the nuclear fluorescent protein under circadian *Reverba* promoter regulation. The second fluorescence image (cropped to 512×312 pixels) was obtained from an experiment on the migration and proliferation of stem cells. We used $R_{\min} = 3, R_{\max} = 32$ and $W = 3R_{\min}$ in either case. Most of the bright blobs were detected in both the images, including the slightly elongated cells in the second image. The very faint nuclei in the first image and the tiny blobs in the second image were not detected. The average execution time for the NIH3T3 and the stem cell image was 0.7 and 0.4 seconds respectively.

5. CONCLUSION

We presented an algorithm for the detection of round cells in fluorescence images using a Mexican-hat-like detector, which offers a nice trade-off between the quality of approximation and the cost of computation. The attractive features of our algorithm are (i) the use of real-valued scale to continuously control the size of the detector, which allowed us to obtain a very precise estimate of the position and the size; and (ii) the fast computation of the response using $O(1)$ operations. Higher computational costs would be involved to achieve similar results using pre-computed filters.

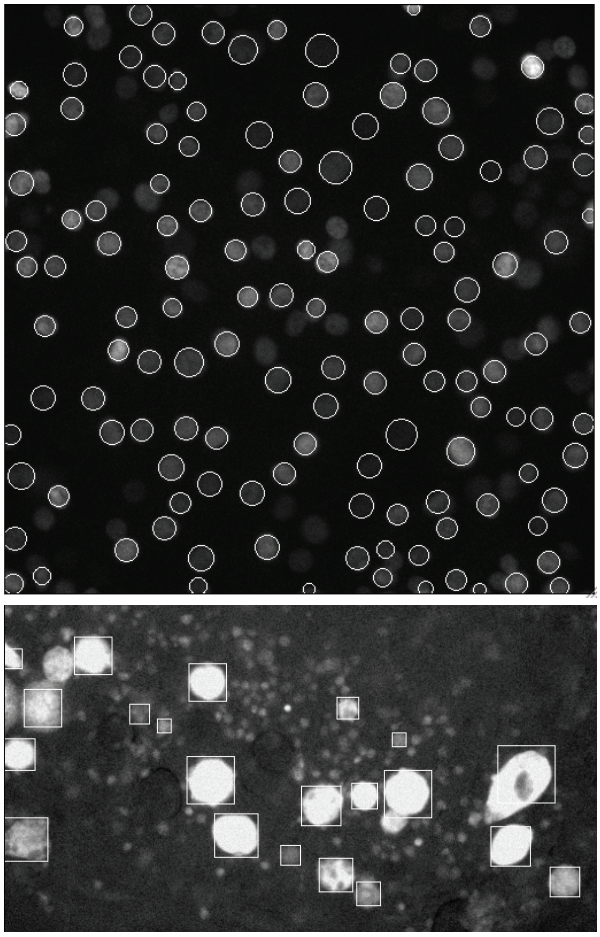

Fig. 5. **Top**: Snapshot of moving cell nuclei (shown using white circles) detected from an image sequence (Courtesy of C. Dibner, University of Geneva); **Bottom**: Bright stem cells (shown using white bounding boxes with a cross) detected from a fluorescence image using our algorithm (Courtesy of N. Garin, ISREC, EPFL).

6. REFERENCES

[1] C. Vonesch, F. Aguet, J.-L. Vonesch, and M. Unser, "The colored revolution of bioimaging," *IEEE Signal Processing Mag.*, vol. 23(3), pp. 20–31, 2006.

[2] R. G. Prabhakar, K. Nandy, J. Collins, K. J. Meaburn, T. Misteli, and S. J. Lockett, "A high-throughput system for segmenting nuclei using multiscale techniques," *Cytometry*, vol. 73A(5), pp. 451–466, 2008.

[3] N. Malpica, C. Ortiz-de Solorzano, J.J. Vaquero, A. Santos, I. Vallcorba, J.M. García-Sagredo, and F. del Pozo, "Applying watershed algorithms to the segmentation of clustered nuclei," *Cytometry*, vol. 28(4), pp. 289–297, 1997.

[4] C. Ortiz-de Solorzano, R. Malladi, S.A. Lelievre, and S.J. Lockett, "Segmentation of nuclei and cells using membrane related protein markers," *Journal of Microscopy*, vol. 201(3), pp. 404–415, 2001.

[5] J. Deng and H.T. Tsui, "A fast level set method for segmentation of low contrast noisy biomedical images," *Pattern Recognition Letters*, vol. 23, pp. 161–169, 2002.

[6] J. Byun, M. R. Verardo, B. Sumengen, G. P. Lewis, B. S. Manjunath, and S. K. Fisher, "Automated tool for the detection of cell nuclei in digital microscopic images: Application to retinal images," *Molecular Vision*, vol. 12, pp. 949–960, 2006.

[7] X. Zhou and Wong S. T. C., "High content cellular imaging for drug development," *IEEE Signal Processing Mag.*, vol. 23(2), pp. 170–174, 2006.

[8] K. N. Chaudhury, A. Muñoz-Barrutia, and M. Unser, "Fast adaptive elliptical filtering using box splines," *IEEE Intern. Conf. on Image Processing (ICIP)*, pp. 785–788, 2008.

[9] K. N. Chaudhury, A. M. Barrutia, and M. Unser, "Fast space-variant elliptical filtering using box splines," accepted in IEEE Trans. on Image Processing.

[10] C. de Boor, K. Höllig, and S. Riemenschneider, *Box Splines*, Springer-Verlag, 1993.