

Carnegie Mellon University
Department of Electrical and Computer Engineering

Master of Science
Project Report

**Cell Detection and Counting for Microscope Images with
Applications to Stem Cell Engineering**

Alexander Samborskiy

Advisors: Professors Jelena Kovacevic and Markus Pueschel

Department of Electrical and Computer Engineering
Carnegie Mellon University, Pittsburgh, PA 15213

November 1, 2007

Contents

1	Introduction	5
1.1	Motivation	5
1.2	Paper Overview	6
2	Stem Cell Engineering	7
2.1	Stem Cell Isolation	7
2.1.1	Immunofluorescent Staining	7
2.1.2	Fluorescent Activated Cell Sorting	8
2.2	Stem Cell Microencapsulation	9
3	Basics of Blob Detection	9
3.1	State of The Art	10
3.2	Differential Blob Detectors	10
4	Multi-Scale Blob Detector	11
4.1	Scaling Property of Local Extremum Over Scales	14
4.2	Blob Detection with Automatic Scale Selection	15
4.2.1	Automatic Scale Selection	15
4.2.2	Gaussian Blob Models	15
5	Multiresolution Blob Detector	16
5.1	Multiresolution Transform	16
5.2	Scale Selection Refinement	17
6	Experimental Results	18
7	Conclusions and Future Work	20

Acknowledgments

Interdisciplinary research has great potential for innovation if in collaboration with experts from different areas. This project combines expertise in biomedical engineering, applied mathematics and computer engineering. I am very happy for the opportunity to thank all the collaborators who together have made this project a success.

First, I would like to express my gratitude to my advisor from the University of Missouri, Professor Pete Casazza, whose work in time-frequency analysis inspired me greatly to move towards solving real-world engineering problems. The project would not have been possible without his invaluable support.

Second, I am grateful to my advisors Professor Jelena Kovacevic and Professor Markus Pueschel who guided me through this project, and whose experience I appreciate. This project shows how much progress can be achieved when all researchers unite their efforts at the same time.

Third, I would like to take an opportunity to thank Professor Stefan Zappe, whose course on BioMEMS enriched my knowledge in the area of Biomedical Engineering.

My sincere gratitude goes to my fellow student Sasha Bakhru who took me to the Mellon Institute labs for experimental part of the work in stem cell extraction and culturing.

Finally, I would like to thank my parents for their love and support throughout my studies.

Abstract

We report cell counting tools developed to automate analysis of microscope images in cell engineering. The analysis of cell images poses several problems, including cell detection, counting, segmentation, and classification. The main application that drives the project is the urgent need for such tools in the automated analysis of stem cells behavior, making it attractive from the point of view of biomedical engineering. The mathematical models we develop are of great interest from the point of view of applied mathematics. Efficient numerical implementation is explored and gives credit to computer engineering.

1 Introduction

1.1 Motivation

Stem cells play a crucial role in medical therapy due to their ability to renew themselves through cell division and to differentiate into a wide range of specialized cell types. There are two major categories of stem cells: embryonic and adult stem cells, which are the primitive precursors to the working cells within an embryo and a living organism, respectively. "Both embryonic and adult stem cells are potential sources for regenerative medicine and tissue replacement after injury or disease."¹

Embryonic stem cells derive from the inner cell mass of a blastocyst and differentiate into three germ layers of an embryo, as shown in Fig. 1.

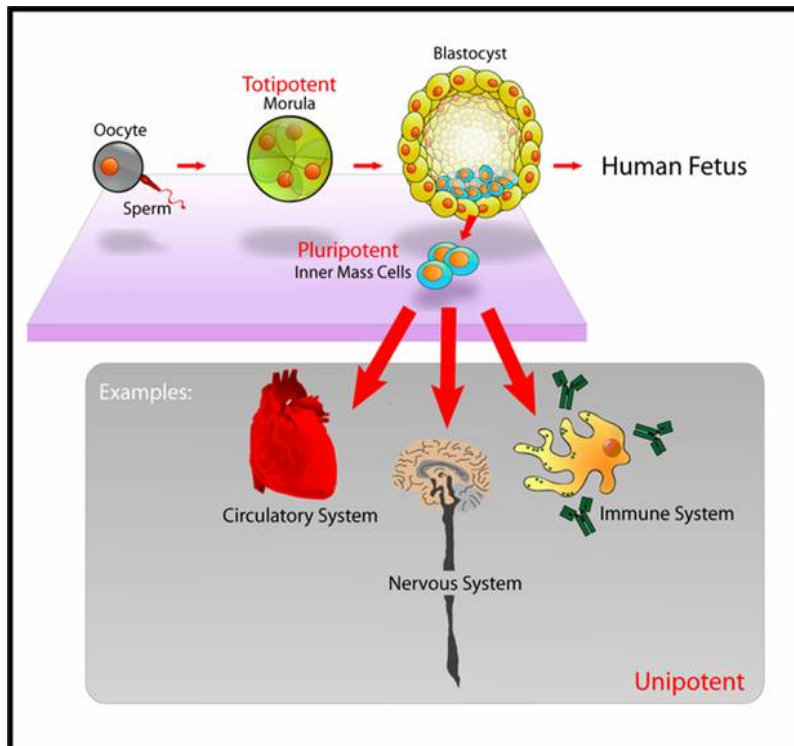


Figure 1: Embryonic stem cells.¹

Adult stem cells reside in a specific area of the tissues such as brain, bone marrow, blood

¹http://en.wikipedia.org/wiki/Stem_cell

vessels, skeletal muscle and etc. Activation mechanism forces adult stem cells to renew themselves and to differentiate into major specialized cell types of the tissue or organ. Each tissue has a very small number of adult stem cells. For effective medical treatment, adult stem cells can be grown ex-vivo in cell culture and then put in microcapsules, which then implanted into the damaged tissue or organ. In the nutrient medium of a microcapsule, stem cells keep growing, dividing and differentiating. When the membrane of the capsule degrades, the cells are released and proper types of differentiated cells recover the tissue. Fig. 2 shows differentiation of neural stem cells into neurons, oligodendrocytes, and astrocytes.

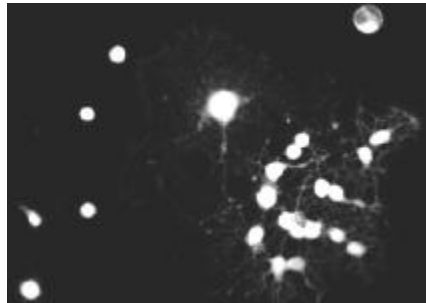


Figure 2: Differentiation of neural stem cells [Courtesy of S. Bakhru and S. Zappe, CMU].

Stem cell engineering poses several problems in the analysis of microscope images of cells. Counting different types of cells in microscope images of a microcapsule allows testing its design for the required cell proliferation rate. Classification simplifies analysis of stem cell differentiation. Tracking studies migration of cells under control signaling. All those procedures need to be automated.

1.2 Paper Overview

In Section 2 we describe experimental work on stem cell engineering conducted at Mellon Institute. Second, we introduce mathematical models of cell detection and counting in Section 3. Third, we discuss multi-scale blob detection with automatic scale selection in Section 4. Fourth, we move to the multiresolution approach used to speed up multi-scale blob detection in Section 5. Finally, we discuss experimental results in Section 6 and offer conclusions with pointers to future work in Section 7.

2 Stem Cell Engineering

In the experiments, we work with neural stem cells that can differentiate into three cell types: neurons, oligodendrocytes, and astrocytes. Stem cell differentiation into neurons would replenish those neurons degraded in Parkinson's disease, for example.

Why do we implant stem cells and not neurons? Stem cells can migrate to the area of a disease and then differentiate into neurons. The number of adult neural stem cells in-vivo is not large enough for effective treatment. Fig. 3 shows the complete cycle of stem cell engineering experiments. We ex-vivo culture stem cells obtained from a healthy mice and then implant them to a sick one.



Figure 3: Stem cell engineering [Courtesy of J. Kovacevic and S.Zappe, CMU].

We now explain each step of the system in more detail.

2.1 Stem Cell Isolation

2.1.1 Immunofluorescent Staining

Stem cells can be marked with immunofluorescent staining. They have special protein that can be uniquely recognized by another protein called antibody. In most immunofluorescence experiments, two antibodies are employed. The primary antibody binds to the special stem cell protein. The secondary antibody binds to the primary antibody and has a fluorescent dye attached to it.² Stem cells with the fluorescent dye can be detected as discussed further.

²<http://simple.wikipedia.org/wiki/Antibody>

2.1.2 Fluorescent Activated Cell Sorting

Once stem cells are marked via immunofluorescent staining, they can be isolated in Fluorescent Activated Cell Sorting (FACS) machine. "The cell suspension is entrained in the center of a narrow, rapidly flowing stream of liquid to a FACS machine. A vibrating mechanism causes the stream of cells to break into individual droplets (one cell per droplet). Before the stream breaks into droplets the flow passes through a fluorescence measuring station where the fluorescent character of interest of each cell is measured. An electrical charging ring is placed just at the point where the stream breaks into droplets. A charge is placed on the ring based on the immediately prior fluorescence intensity measurement and the opposite charge is trapped on the droplet as it breaks from the stream. The charged droplets then fall through an electrostatic deflection system that diverts droplets into containers based upon their charge. The stream is then returned to neutral after the droplet breaks off. Thus, stem cells that were stained would be diverted into one container, where they would be collected."³

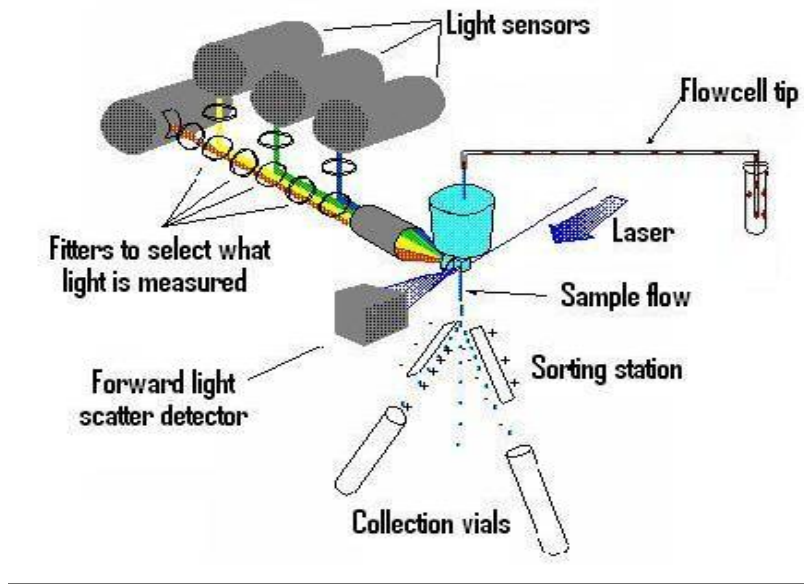


Figure 4: FACS Machine.³

In the experiments, S.Bakhru and S.Zappe did not manage to achieve good sorting quality. The resulted suspension had some stem cells, and yet many other cells and their contents. The quality

³<http://meds.queensu.ca/qcri/flow/cr-fc.htm>

of cell sorting is very sensitive to physical parameters of FACS machine.

2.2 Stem Cell Microencapsulation

Why are encapsulated cells instead of directly implanted? The strong signal in the area of implantation due to surgical intrusion distracts stem cells from moving towards the area of a disease. "The core of the microcapsule provides a micro-environment conducive to the maintenance of cell viability" [1]. The nutrient media of a capsule controls proliferation rates of differentiated cells. "Membrane of the capsule mediates the exchange of nutrients, wastes, gasses, and signaling molecules with the ambient media" [1]. Thickness of the membrane governs time during which the membrane degrades completely and all cells are released.

By designing the nutrient media and thickness of the membrane, one can control differentiation of stem cells to the desired number of particular type of cells, released from the capsule after a defined period of time. If capsules are properly designed, the number of released cells would be sufficient to recover the tissue. The period of time over which the membrane degrades is important. If cells are released shortly after implantation, they will be distracted by a strong signal caused by the surgical intrusion and not by the disease.

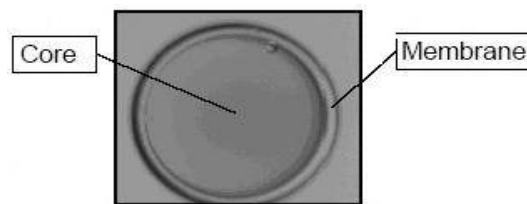


Figure 5: Polymeric Microcapsule [Courtesy of S. Bakhru, CMU].

3 Basics of Blob Detection

We now move to the specific task we addressed in this work. Many methods in automated cell counting are based on segmentation techniques that require intensive user interaction to obtain parameter settings for accurate cell segmentation for different type of images [2]. An alternative approach suggests that most cells can be approximated by circular shapes, which are much brighter

or darker than the surrounding. It is shown in [3] that blob detectors provide an accurate, simple and fast method to detect cells of circular shapes and this approach can be applied to any type of microscope images.

3.1 State of The Art

There are two main classes of grey-scale blob detectors, one based on local extremum in the intensity landscape called watershed detection and the other on derivative expressions called differential detectors. We do not consider watershed based blob detectors due to their inherent noise sensitivity and oversegmented results [4].

The response of most commonly used blob detectors strongly depends on the relationship between the size of blob structures and the scale of the Gaussian kernel used for pre-smoothing. The blob detector described in [3] assumes all blobs are of equal size, known in advance. In [2], T. Lindeberg proposed a multi-scale blob detector with automated scale selection that finds size and position of each detected blob.

The main drawback of grey-scale blob detectors is that they are not using advantages of color vision. Color is an essential component that discriminates among objects. Some attempts to extend grey-scale blob detectors to the color domain have been done in [5], where the authors explore correlation between color planes.

3.2 Differential Blob Detectors

We now briefly review differential blob detectors upon which our work is based. Let us consider the Gaussian $g(x, y; \sigma)$:

$$g(x, y; \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}, \quad (1)$$

where σ is the standard deviation. We define scale-space representation $L(x, y; t)$ of the image $f(x, y)$ by

$$L(x, y; t) = f(x, y) * g(x, y; \sigma), \quad (2)$$

where $*$ is the convolution operator and $t = \sigma^2$ is the variance.

Most common differential blob detectors use the Laplacian operator ∇^2 applied to the scale-space representation of an image:

$$\nabla^2 = \frac{d^2}{dx^2} + \frac{d^2}{dy^2}, \quad (3)$$

$$\nabla^2 L = \nabla^2(f * g) = (\nabla^2 g) * f. \quad (4)$$

Applying (3) to (4) yields Laplacian of the Gaussian (LoG):

$$\nabla^2 g = -\frac{1}{\pi\sigma^4} \left[1 - \frac{x^2 + y^2}{2\sigma^2} \right] e^{-\frac{(x^2 + y^2)}{2\sigma^2}}. \quad (5)$$

The Gaussian is often used to smooth an image and to attenuate noise. The Laplacian highlights regions of rapid intensity change and is therefore often used for edge detection [6]. As scale t of LoG increases, edge-like points converge to the unique point of local extrema at some scale. This intuition motivates using LoG filter in blob detection.

The second type of blob detectors is based on the difference of Gaussians, which approximates LoG:

$$\nabla^2 L(x, y; t) \approx \frac{1}{2\Delta t} (L(x, y; t + \Delta t) - L(x, y; t - \Delta t)). \quad (6)$$

The third popular blob detector, a determinant of the Hessian, exploits other second order derivatives to characterize blob structures:

$$\det H(L(x, y; t)) = L_{xx}L_{yy} - L_{xy}L_{yx}. \quad (7)$$

While it is not clear which blob detector can be used for cell detection, in this report we explore and implement LoG, known to be the most popular one. Fig. 6 and 7 provide the discrete approximation of (5) and the corresponding plot, respectively.⁴

4 Multi-Scale Blob Detector

To detect blob of size s , we use LoG with standard deviation σ , computed by

$$\sigma = \frac{s - 1}{3}. \quad (8)$$

⁴<http://homepages.inf.ed.ac.uk/rbf/HIPR2/log.htm>

0	1	1	2	2	2	1	1	0
1	2	4	5	5	5	4	2	1
1	4	5	3	0	3	5	4	1
2	5	3	-12	-24	-12	3	5	2
2	5	0	-24	-40	-24	0	5	2
2	5	3	-12	-24	-12	3	5	2
1	4	5	3	0	3	5	4	1
1	2	4	5	5	5	4	2	1
0	1	1	2	2	2	1	1	0

Figure 6: Discrete approximation to LoG with $\sigma = 1.4$.⁴

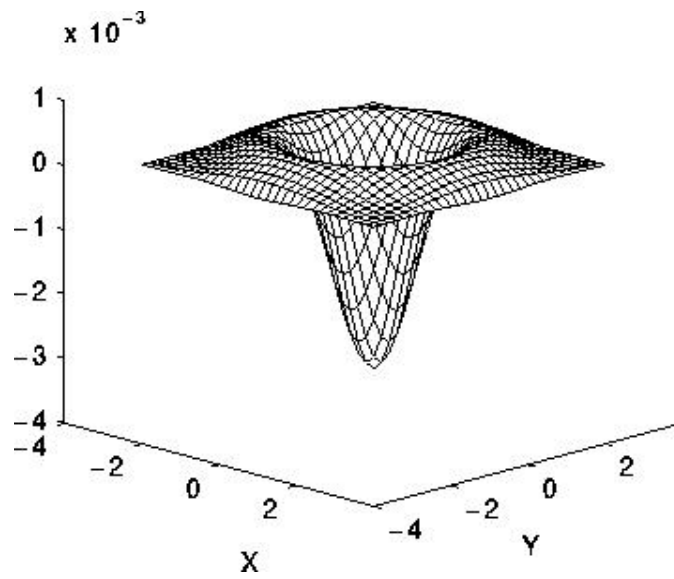


Figure 7: 2D LoG. The x and y axes are marked in standard deviations.⁴

This formula is motivated by the "3 σ " rule that 99% of energy of the Gaussian is concentrated within three standard deviations of the mean. The interval that has 99% energy of the Gaussian is called the support of LoG.

Let us demonstrate LoG response on blobs of different sizes. Fig. 9 illustrates strong responses of LoG on some blob structures from the test image in the Fig. 8.

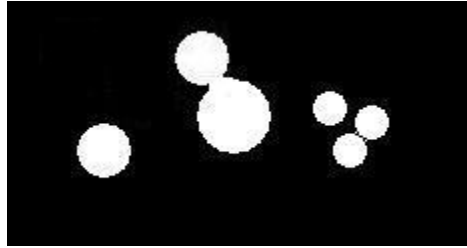


Figure 8: Synthetic test image with blobs of different sizes.

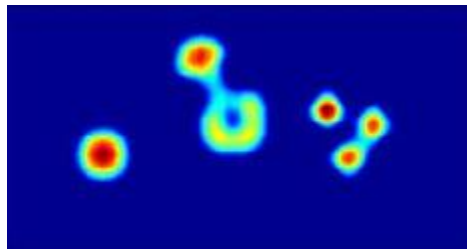


Figure 9: Negative LoG response on blob structures. Greyscale is under the color map, where blue corresponds to black and red corresponds to white.

We compute the support of LoG by (8) that detects the blob of medium size. In the above figures, we observe that the support of LoG is not large enough to yield the maximum response in the center of the largest blob being left undetected.

From the example above it follows that the blob detector at a fixed scale can detect blobs of one size. To detect blobs of different sizes, a multi-scale blob detector needs to be designed. It is desirable for a multi-scale blob detector to be robust with respect to scaling in the image domain. In the next section we discuss a multi-scale blob detector that meets this requirement.

4.1 Scaling Property of Local Extremum Over Scales

Let us consider a signal f and its scaled version f' defined by

$$f(x) = f'(sx) \quad (9)$$

The scale-space representations of f and f' are then

$$L(x; t) = g(x; t) * f(x), \quad (10)$$

$$L'(x'; t') = g(x'; t') * f'(x'), \quad (11)$$

where the spatial and scale variables are transformed according to

$$x' = sx, \quad (12)$$

$$t' = (\sigma')^2 = (s\sigma)^2 = s^2\sigma^2 = s^2t. \quad (13)$$

It follows that

$$L(x; t) = L'(x'; t'), \quad (14)$$

$$\partial_{x^m} L(x; t) = s^m \partial_{x'^m} L(x'; t'). \quad (15)$$

To obtain derivative scale-invariance, m -th order spatial derivatives in (15) are normalized:

$$\partial_{\xi} = t^{1/2}(\partial_x), \quad (16)$$

$$\partial_{\xi'} = t'^{1/2}(\partial_{x'}), \quad (17)$$

and we have

$$\partial_{\xi^m} L(x; t) = \partial_{\xi'^m} L(x'; t'). \quad (18)$$

Scale-invariance (14) and (18) support the scaling property of local extrema over scales for the scale-space representation of the signal (2) and its normalized spatial derivatives (15). It states that if a scale extrema is assumed at $(x; t)$ in the scale-space representation of the signal f , then in a rescaled signal f' , a corresponding scale extrema is assumed at $(sx; s^2t)$ in the scale-space representation of f' [7].

4.2 Blob Detection with Automatic Scale Selection

A blob is detected, if one can find a scale of the filter, at which it has the strong response to that blob. Since size of blob structures is not known in advance, we would try to extract size information from the scale of the filter, which detects the blob. For this purpose, a multi-scale blob detector should have one-to-one correspondence between the selected scale of the filter and the size of blobs that can be detected by the filter at that scale. For some blob models this correspondence might be obtained, if one selects the scale of the filter, at which it has the strongest response to the detected blob. This approach is called blob detection with automatic scale selection [4].

4.2.1 Automatic Scale Selection

The framework for a scale-invariant multi-scale blob detector based on LoG is given by:

$$\nabla_{norm}^2 g = t \cdot \nabla^2 g, \quad (19)$$

$$L = \nabla_{norm}^2 (g * f) = (\nabla_{norm}^2 g) * f, \quad (20)$$

$$(\hat{x}, \hat{y}; \hat{t}) = \arg[\text{extremum}_{(x,y;t)} L(x, y; t)], \quad (21)$$

where (\hat{x}, \hat{y}) corresponds to the center and \hat{t} to the size of the detected blob.

First, we compute discrete scale-normalized LoG matrix kernel that approximates (19) at different scales. Second, we apply normalized LoG at different scales to obtain a 3D discrete scale-space volume $L(x,y;t)$ in (20). Third, we find all local extrema of the scale-space volume $L(x,y;t)$. The argument of each local extrema in (21) corresponds to a center of the detected blob (\hat{x}, \hat{y}) and its radius \hat{t} . It can be expected that for most common Gaussian blob models the solution to (21) would match the size and the center of each detected blob.

4.2.2 Gaussian Blob Models

Let us consider a Gaussian function as a model of a 2D blob of size t_0 :

$$f(x, y; t_0) = \frac{1}{2\pi t_0} e^{-(x^2+y^2)/2t_0}. \quad (22)$$

From the semi-group property of the Gaussian kernel

$$g(\cdot, t_A) * g(\cdot, t_B) = g(\cdot, t_A + t_B), \quad (23)$$

it follows that

$$g * f = g(x, y; t + t_0). \quad (24)$$

It is easy to see that the response of the filter is the strongest at the center point of the blob:

$$L(0, 0; t) = \nabla_{norm}^2(g * f)(0, 0; t) = \frac{t}{\pi(t + t_0)^2}. \quad (25)$$

The maximum over scales is given by

$$\partial_t(\nabla_{norm}^2(g * f)(0, 0; t)) = 0 \Leftrightarrow t = t_0. \quad (26)$$

Thus, for Gaussian blob models, local scale-space extrema over scales of normalized LoG convolved with an image, detects blobs of different sizes, while adapting the local scales of processing to image data.

In the next section we discuss multiresolution techniques that speed up calculations (19)-(21).

5 Multiresolution Blob Detector

5.1 Multiresolution Transform

Multiresolution techniques are extensively used in the analysis of image and signal data at different scales [8]. The most popular tool is the Discrete Wavelet Transform that decomposes a signal into a coarse space and many detail spaces through a set of filters called filter bank and downsampling (see Fig. 10).

Assume that the filters $h_{i,l}, g_{i,l}, i = 0, \dots, M - 1, l \in l_2(Z)$ are all FIR, causal, of length L and that the family $\Phi = \phi_{i,n-Nk}, i = 0, \dots, M - 1, n, k, \in Z$, with $\phi_{i,n} = g_{i,n}$ spans $l_2(Z)$. Let also $\bar{\Phi} = \bar{\phi}_{i,n-Nk}, \bar{\phi}_{i,n} = h_{i,n}$ be the span of dual space. Then $\Phi\bar{\Phi}^* = I$.

Given the input $x \in l_2(Z)$, the M channels of the analysis filter bank decompose x into its transform coefficients $y = \bar{\Phi}^*x$ performing a signal expansion in $l_2(Z)$, whereas the synthesis filter

bank reconstructs the same signal from its transform coefficients $x = \Phi y$. We call this a perfect reconstruction filter bank. Two level Discrete Wavelet Transform Decomposition of the test image from Fig. 8 is provided in the rightmost image of Fig. 11.

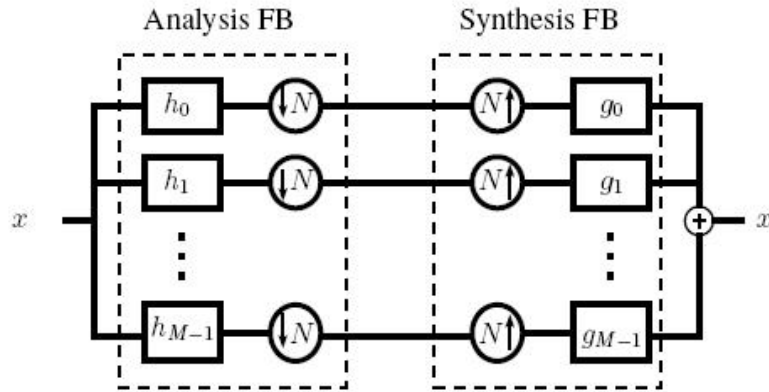


Figure 10: M-channel filter bank with downsampling by N .

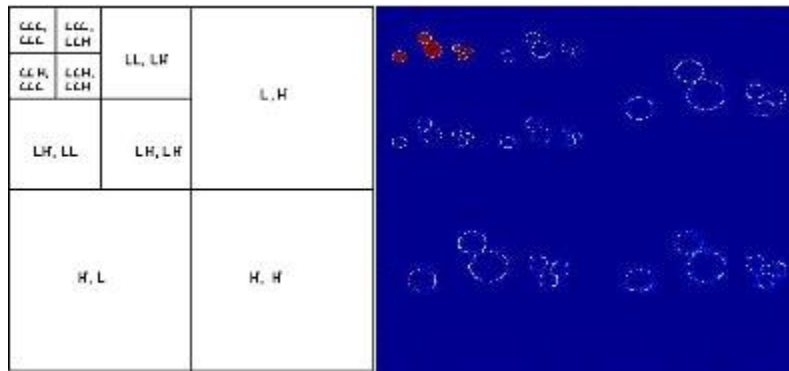


Figure 11: Two level filter bank decomposition of the test image.

5.2 Scale Selection Refinement

We now estimate computational cost of blob detection using two level filter bank decomposition of the test image. We apply LoG at different scales to solve (21) for LL band at each level of decomposition.

Let T denote the scale range among which the local extrema (21) is sought in the original image.

At the second level of decomposition, the size of the LL band is $(\frac{N}{4})^2$. The scale-search range is then reduced to $\frac{T}{4}$. The number of operations to solve (21) at the LL band of the second level of decomposition is then $O(\frac{T}{4} \times (\frac{N}{4})^2)$. The solution $(\hat{x}, \hat{y}; \hat{t})$ approximates solution to (21) for LL at the finer scale. Therefore, at the first level of decomposition, the solution of (21) is in the neighborhood of $(2 \times \hat{x}, 2 \times \hat{y}, 2 \times \hat{t})$. Refinement takes $O(1)$ operations. Thus, the computational cost is reduced to $O(\frac{T}{4} \times (\frac{N}{4})^2)$.

In the following figures we move through LL subbands from the coarsest scale to the original image. In each figure we detect blobs of a particular size. The leftmost figure detects blobs of the smallest size, the middle figure blobs of medium size and the rightmost figure blobs of the largest size. The size and the center of each detected blob is refined as we proceed to the finer scales.

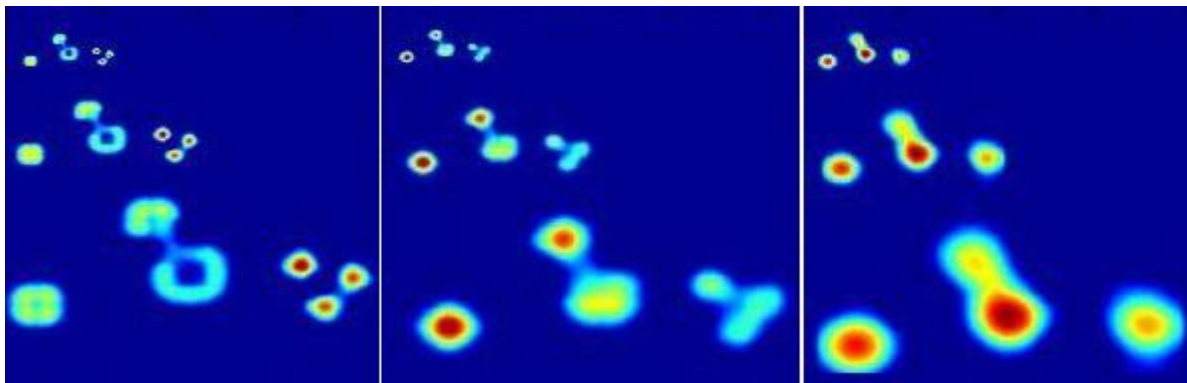


Figure 12: Three level filter bank decomposition of the test image.

6 Experimental Results

We now turn our attention to the real images. The experimental results are provided in Fig. 13. The first image is the test image with muscle cells of fly embryo, which are to be detected and counted. The second image is the filter action on the test. The third image points to local maximums, each of which corresponds to one detected cell. The size of detected cells varies between 6 and 8 pixels.

It is not surprising that blob detection with automatic scale selection works on fly embryo image data. Fig. 14 shows that the profiles of muscle cells are of Gaussian shape.

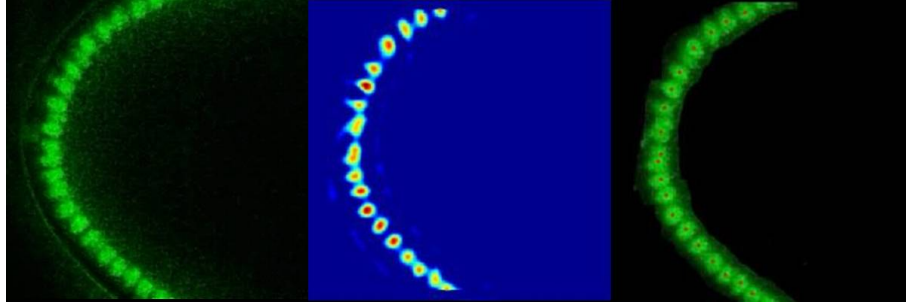


Figure 13: Automated cell detection and counting (test, filter, result) [Original image courtesy of S. Zappe, CMU].

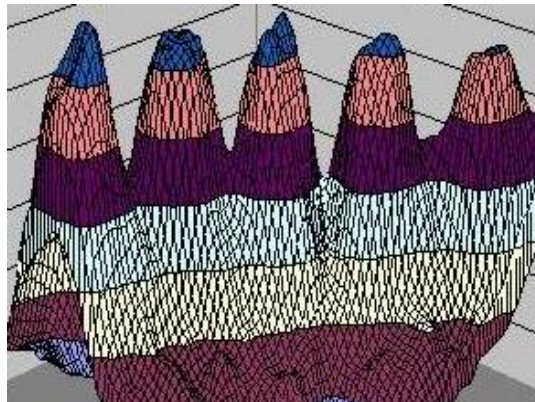


Figure 14: A 3D profile of five cells of the fly embryo.

7 Conclusions and Future Work

We presented an implementation of a blob detector that automatically finds the size of each detected blob.

Future work includes developing techniques to detect and count cells distributed in volume as confocal microscopy provides 2D slices of cells in the form of a z-stack. Correlation between slices could be effectively used to improve robustness of the method.

Automated classification and tracking of stem and differentiated cells are under current research. Automated cell detection, counting, classification, and tracking could then be built into a toolbox that would facilitate automation analysis of stem cell behavior.

References

- [1] S. Bakhru, “Functional polymeric microcapsules for neural stem cell culture,” M.S. thesis, Johns Hopkins University, May 2006.
- [2] N. Malpica, C. Ortiz de Solzano, J. Vaquero, A. Santos, I. Vallcorba, M. Garcia-Sagredo, and F. Del Pozo, “Applying watershed algorithms to the segmentation of clustered nuclei,” *Cytometry*, vol. 28, pp. 289–297, 1997.
- [3] J. Byun, M. Verardo, B. Sumengen, G. Lewis, B. Manjunath, and S. Fisher, “Automated tool for nuclei detection in digital microscopic images: Application to retinal images,” *Molecular Vision*, vol. 12, pp. 949–960, August 2006.
- [4] T. Lindeberg, “Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention,” *International Journal of Computer Vision*, vol. 11, no. 3, pp. 283–318, 1993.
- [5] A. Ming and H. Ma, “A blob detector in color images,” *6th ACM international conference on Image and video retrieval Proc.*, pp. 364–370, January 2007.
- [6] R. Gonzales, R. Woods, and S. Eddins, *Digital image processing with Matlab*, Pearson Prentice Hall, 2004.

- [7] T. Lindeberg, “Feature detection with automatic scale selection,” *International Journal of Computer Vision*, vol. 30, no. 2, pp. 77–116, 1998.
- [8] J. Kovacevic and M. Vetterli, *Wavelets and subband coding*, Prentice Hall, 2005.