Journal of Imaging Science and Technology® 50(3): 233–242, 2006.

Review of the Basic Image Processing and Segmentation Techniques for Biological Images

Anil Tarachandani and Dutch Boltz
Merck & Co. MRL, Cardiovascular Diseases, Rahway, New Jersey 07065
E-mail: a_tarachandani@merck.com

Abstract. *High throughput screening has been used to rapidly screen for chemical compounds in a biological assay. Until recently, many of the biological assays utilized simple biochemical techniques, the result of which could be interpreted in single or at most a few numerical values. That made it easy to evaluate, without bias, any unique chemical entities screened. However, with biological cells or tissue images, the information was qualitative or at best limited to simplified algorithms. Recently, it is now becoming possible to perform standardized assays and utilize complex image data to derive reproducible information which could be utilized to precisely quantify the efficacy of compounds. Much of this is possible due to the precise mathematical algorithms that are used to compute image data to derive information. This review will discuss some of the basic algorithms involving kernel operations that are commonly used and how they can be applied for any image or picture data.* © *2006 Society for Imaging Science and Technology.*
[DOI: 10.2352/J.ImagingSci.Technol.(2006)50:3(233)]

WHAT IS IMAGE PROCESSING AND WHY IS IT USED IN BIOLOGY?

A lot of the raw data in biology is represented as images, either as photographs of gels, autoradiograph patterns, or cell and tissue pictures. Until recently, many of those pictures were captured and stored as photographic film, but with the advent of cheaper computers, disks, and digital cameras, the trend is to capture and store pictures digitally. Thus, instead of storing patterns of silver grains in a matrix, pictures are stored as series of magnetically encoded bits on a disk or tape.

The benefit of storing images digitally as a series of numbers is that they can be easily used to extract quantitative information, once the areas of interest are defined. The information extracted can then be used to measure the density of spots in an image or bands on a gel. If such a spot in an image can be discriminated as an area of interest from the background, then it is possible to calculate the number of spots such as nuclei or mitochondria.

When the image is represented as a series of numbers, the identification and quantitation of objects is primarily a series of mathematical operations. However, some image processing operations are more easily accomplished than others. For example, quantitation of spots in an image from a gene chip experiment might be easier to accomplish because the area where the DNA has been spotted is defined and all that is required is measurement of average/median intensity of the spot. In contrast, determining the pattern of staining in a PAP smear between normal and malignant tissues might be more challenging.

The image processing algorithms that are applied in biology can be used on various types of images, thus, routines to extract a spot from a gel may be similar to those used to identify the nucleus in a histological preparation.[1–5]

One of the questions frequently asked is why would you want to image process or alter the image? The reasons are many, but here are three most common reasons. Image processing is frequently used to quantify objects in an image and so as a first step in identification of the image, a copy of the image is enhanced or changed in some way which will help in identifying objects. Once those object outlines are identified and a mask created, many diverse measurements can be done on the raw image. The second is image enhancement. Sometimes, lighting or camera conditions or printing alter the color or image intensity in the image. To correct such an effect and make a better representation of the image that was observed, brightness, contrast, or hue/saturation is altered. It must be noted that such "altered" images cannot be used for quantitation and may be used for illustrative purposes only. Third, modern professional cameras can capture variations in intensity of incident light to 65,535 individual gray levels. Most printers or monitors cannot represent this full gradation of intensity and, hence, display only a portion of the intensity. Thus, though the whole gamut of intensity is available in the raw image, only a portion of that is shown in the "best-fit" image or "contrast-stretched" image on the screen/printer.

Most biological laboratories capture images in single steps with the user of the camera actively involved in image capture. This allows for a lot of flexibility during capture, thus, if the light is not uniform, the user can adjust the light to capture a visually pleasing image. However, recent trend in biotechnology is to capture thousands of images from a multiwell plate or tissue arrays and analyze the image to give quantitative data. This has led to an explosion in the number of images that are acquired and the techniques that are used to analyze the images. Cellomics Inc. was the first to coin the term "high content screening" which implied high resolution image analysis of thousands of individual biological samples treated with unique chemical entities. The techniques mentioned in this review were originally designed for

Received Feb. 25, 2005; accepted for publication Jun. 16, 2005.
1062-3701/2006/50(3)/233/10/$20.00.

image processing of any picture data and have now been adapted for high content screening.[6]

HIGH CONTENT SCREENING INVOLVES THREE BASIC STEPS

(1) Biological processing to generate the plates or images that will be analyzed: This would entail all the staining and preparation steps for cells in Multiwell plates and is absolutely assay dependent.
(2) Acquisition and storage of images: This step is important for data quality and depends on the microscope or instrument that is used to acquire the images. This part is not assay dependent and somewhat irrespective of what is being analyzed assuming that the machine is capable of acquiring the object/region of interest of varying intensity.
(3) Image analysis and data generation: This is the most crucial step since without the judicious analysis the image is just a pretty picture. It is not assay dependent but rather method dependent. For example, it does not matter which protein is being measured in the nucleus but it does matter that the nuclear signal is being measured as opposed to the cytoplasmic intensity.

This review will primarily address the third step of high content screening involving image analysis and data generation which is done in two fundamental stages:

(a) Image segmentation: Separation and/or marking of cell regions or any regions of interest from the rest of the image such that they can be specifically selected. This is the most crucial step and much of image processing techniques target this step. This could as simple as separating bright cells from the background or very complicated determination of cellular patterns in histological preparations.
(b) Measurement: Once the region of interest has been identified, the objects or its intensities are measured, the objects in the image counted and numerical data generated. This stage would also involve any secondary computation as in object parameters such as roundness, length of its axis and background subtraction. One can then generate a table of results and make graphical plots of different parameters for subsequent comparison.

Since both of the steps involve image processing, they will be discussed in the context of image processing, starting from the basics of image representation to data extraction.

HOW IS THE IMAGE REPRESENTED IN A COMPUTER?

Cameras are made up of many individual detectors arrayed in a matrix. In the simplest case of the black and white camera, the number of individual detectors along the horizontal and vertical axis determines the resolution of the camera. Thus 640×480 implies 640 individual detectors in horizontal axis along with 480 detectors along the vertical axis for a total of 307 200 total detectors. These individual detectors are called pixels and each of the detectors can convert the photons into electrons. The intensity of light at each pixel can be expressed as number of gray levels. A consumer grade camera can distinguish 255 different intensities of light or 255 gray levels. Microscopy or professional grade cameras can distinguish up to 65 535 levels of gray. The number of grays that a camera can distinguish determines the variation in intensity of light that the camera can see in the same field. Thus, the professional camera will be able to visualize the brightest object and the faintest object in a field at the same time. When a picture is taken, each of the detectors in a camera sends a number representing the intensity or the photon count to the computer. In a black and white camera each number is represented as 1 or 2 bytes for a professional camera. The intensity level in each pixel is represented as a series of numbers and stored in a computer file to represent the image. When we display or print this image, each of these numbers are represented as intensity values on the screen in the same horizontal $\times$ vertical format as the detector. To keep this article simple we will mention only the gray scale image processing. One of the important gray scale cameras used for scientific work is called a charge coupled device (CCD) because of its linear sensitivity to light and relatively low noise. In case of color pictures there are three individual values for the detector array for each of red, green, and blue and they can be processed together or independently of each other.[7]

COMMON IMAGE ENHANCEMENT/PROCESSING METHODS

Before trying to understand image processing methods, it is important to know how the pixel intensity values of any image are represented. In our examples we will use 0 as the minimum value and 255 as the brightest value. As a note, in a 2 byte/pixel image, the brightest value is 65 535 but otherwise the operations are similar. Thus, an important way that any image can be represented is to plot its brightness histogram which plots the distribution of intensities of pixels in that image. For example, Fig. 1(b) is a grayscale image of the original image in Fig. 1(a) and Fig. 1(e) is the brightness histogram. In some fluorescent images, e.g., Fig. 2(a), when the fluorescent image of the cell is bright against a dark background it is possible to distinguish two peaks in the brightness histogram. One corresponds to the background as dark pixels and the other corresponds to the cells. This is the basis of one common method of selecting objects of interest by using the brightness histogram to do thresholding. This is the ability to identify objects by selecting pixels above certain threshold intensity and thus identifying bright cells, Fig. 2(b).

Brightness and Contrast

These are the two most commonly used methods in image enhancement. They were the most common image process-

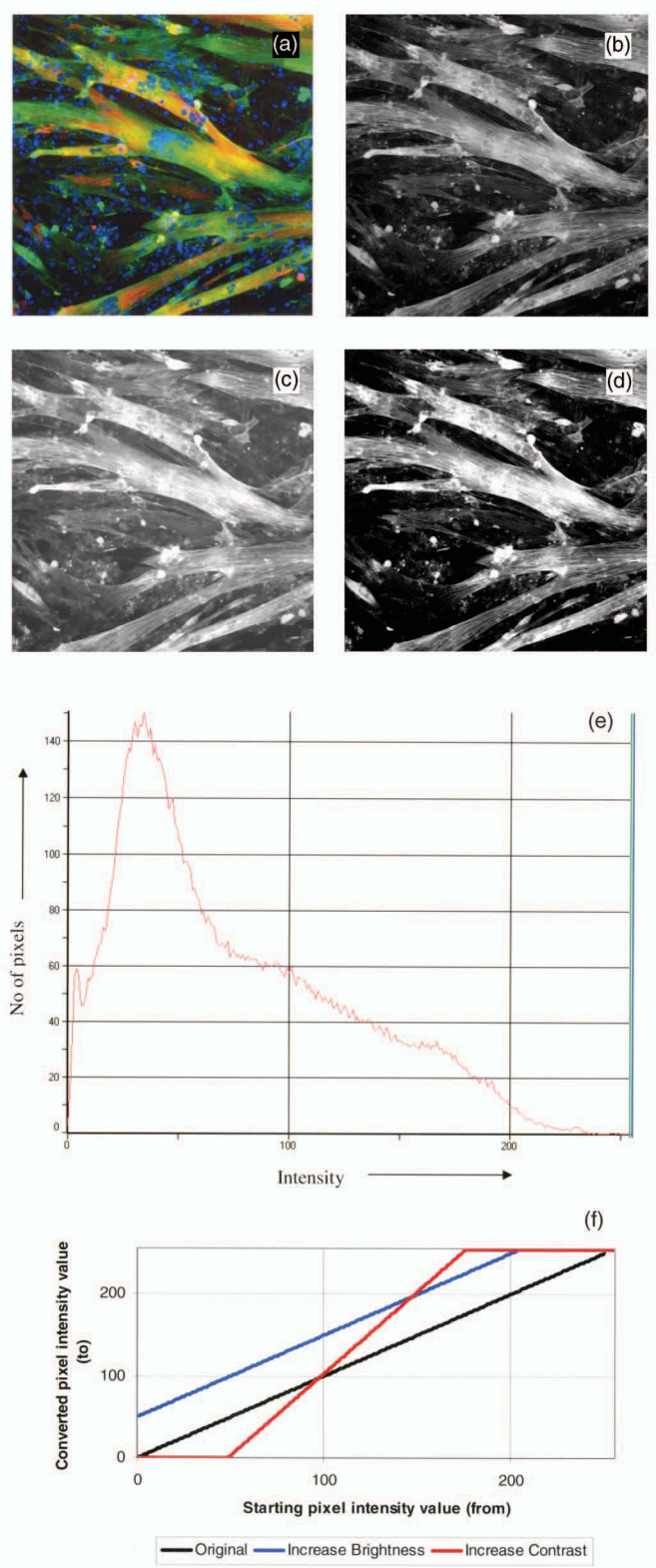


Figure 1. The color image in (a) was converted to the grayscale image (b) and its intensity histogram plotted (e). Using the look up table (LUT) shown in (f) the brightness of the image was increased (c) or the contrast increased (d).

ing tool used, including the first black and white TV sets, and can be intuitively understood so are a good starting point to understand image processing routines. One way to

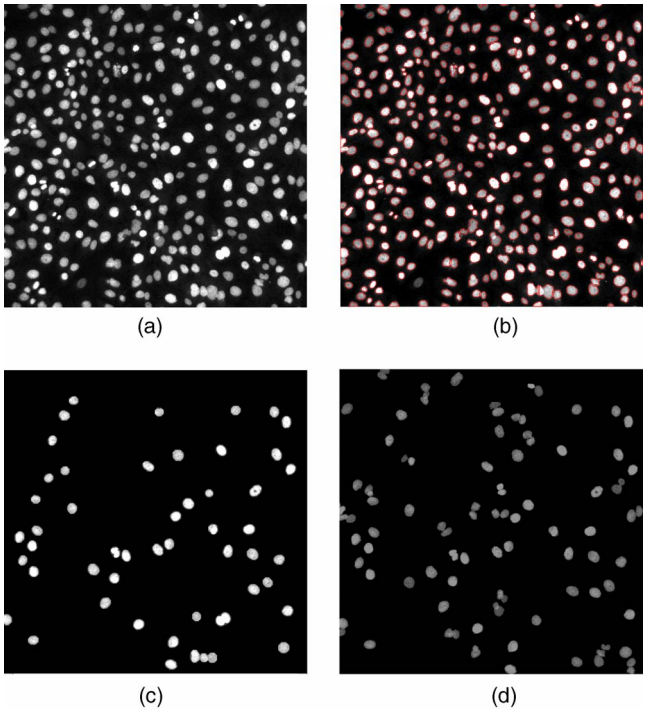


Figure 2. Picture of cell nuclei stained with Hoescht 33342 (a). Since the intensity of nuclear staining was above the background they were selected (b) based on their higher intensity values. Once separated that way, it's also possible to select only the very high intensity of staining (c) vs. the low intensity values (d).

change how the image appears is to change the way in which the pixel intensities are increased or decreased relative to the whole image. What brightness does is to increase the intensity of all the pixels in an image and so the image looks brighter. The simplest way to conceptualize this is as a simple addition in a 1 byte/pixel image. A predetermined number is added individually to all the pixel intensity values and anything which adds up to above 255 is given a value of 255 or "clipped" to 255. However, in a computer, it is implemented by using a lookup table (LUT). This is a table which has the starting intensity values of pixels from 1 to 255 along one axis versus the final values of pixels on the other axis [Figs. 1(c) and 1(f)]. The reason to use these LUT tables is speed. So instead of adding values for thousands of pixels in an image, one pixel at a time, all that a LUT requires is a simple substitution. So in our example of brightness, Fig. 1(f) shows the LUT graphically. It tells us that all the intensities of pixels have been increased. Therefore, brightness processing tends to make all values increase towards the highest pixel value. Similarly, contrast enhancement increases the observed difference in the middle ranges while it pushes the darker pixels to zero and the brighter ones to 255 as shown in Figs. 1(d) and 1(f).

Basic Image Operations

Since images in computer are represented as numbers, it is possible to do arithmetic operations on images. Thus, as we saw earlier, it is possible to add numbers to image intensity values. Correspondingly since two images are a series of

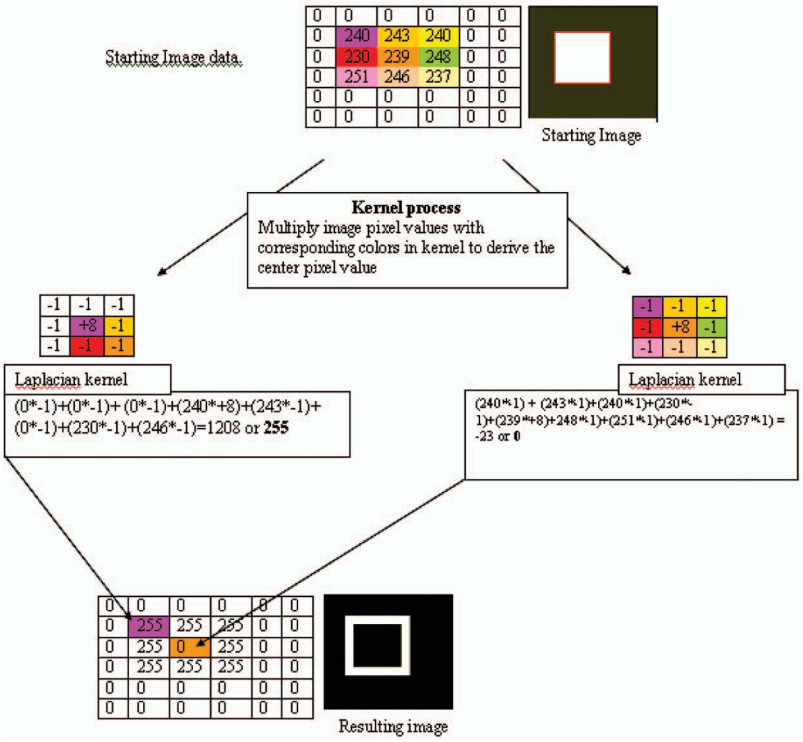


Figure 3. The Laplacian kernel enhances the edges of objects as shown in the resulting image at the bottom. This is implemented using the 3×3 kernel matrix as shown and requires 9 multiplications and 8 additions per pixel processed. The color coding signifies which locations in the kernel and the image are multiplied together.

numbers, it is possible to subtract one image from the other, pixel by pixel, assuming they are of similar dimensions. This could be useful while trying to determine if anything is changed between two images. In regions that nothing has changed the resulting image would appear black whereas in other regions it would be some other value besides zero. This method is very useful in trying to spot temporal changes in images. Similarly, dividing one image by the other quickly highlights the differences between the two images. These methods are often used in object detection.

Object Detection or Image Segmentation

In fluorescent images, such as those which have bright cells/objects against a relatively dark background, it is possible to select the cells/objects of interest using mathematical methods. As a first step, we can convert the standard grayscale image to a binary image which is the same dimensions of the grayscale image but instead of intensity values, each pixel is represented as either black (zero) or white (one). For example, in Fig. 2(b), to select bright fluorescent cells against the duller cells in a dark background one would select a specific pixel intensity value. This intensity is selected to be above the pixel intensity of duller cells. This is called the threshold value. Any pixel above this intensity is made white and the rest black. This helps us identify any bright cells that are marked as shown in Fig. 2(b). Once the cells are identified against the background it is possible to select on the basis of intensity for only the brightest cells in a population as shown in Fig. 2(c) or the dullest in Fig. 2(d). Again, practically, in a computer, this is accomplished by using a LUT which has a list of values that can be substituted dependent on the input value. However, this simple method of object identification works only in the simplest cases and more complicated kernel operations may be required to identify and quantitate objects.

KERNEL OPERATIONS OR MORPHOLOGICAL PROCESSES

In much of the image manipulation operations mentioned earlier all the pixel values of an entire image are compared to, or used to compute, the values in another image. However, many times it is important to process the individual pixels dependent on the neighboring pixels within the same image, which is important in object identification or noise reduction. In this method, the resulting output image is derived from a single input image. The pixels in the output image are calculated using a series of operations on pixel groups in the input image. As shown in Fig. 3, each pixel in the output image is calculated from the source image. Each pixel value of the output image, at every location, is computed from the surrounding pixels in the source image using a weighted matrix, also called a kernel. The numbers or values that constitute the kernel determine the action of this mathematical operation. In the example in Fig. 3, the Laplacian kernel will enhance the edges in an image. This is useful in quickly highlighting objects in an image. This processing is done with each pixel and its neighbor and moves

across all the pixels in the image. The weights and dimension of matrix/kernel determine the strength and effect.

In the early days of image processing, the kernel operations were done mainly on binary images—an image which has only two values for each pixel, dark (0) or white (1) images. This image is derived from a prior thresholding operation in which the features had been roughly detected by a process in which any pixel above a threshold value is made 1 and all the others pixels have the value 0. As a second step, these kernel operations help in refining the selected object. It takes millions of mathematical calculations to do a single image. A binary image can be easily calculated using the quick boolean and operation but using these on grayscale images involves multiple mathematical multiplication and addition operations on each pixel and can take lot of computation time. With the advent of cheaper and faster computers, it is feasible to do these operations on grayscale images as some form of image improvement or image augmentation to facilitate better object segmentation, Fig. 3. It should be pointed out that in many cases the results of kernel operations for any particular pixel result in values which may be below 0 or above 255 (65 535 for a 2 byte/pixel image). In those cases the values are clipped to 0 or 255.

There are many types of kernel operations and this review will address only some of the most useful and common ones. Though the binary image and grayscale image operators are different, the operations can be discussed in general except for the open and close operations which are more powerful for binary images. All kernels will be represented in the simpler 3×3 matrix for grayscale processing as shown in Table I.

Local Equalization or Low Pass Filter

This is the simplest operator to understand. The center pixel in the resulting image is an average of the nine pixels—the center pixel and the eight surrounding pixels. In this filter all the nine pixels in a 3×3 matrix are multiplied by 1 and the result divided by nine. Thus, each of the nine pixel values have equal weight. Using this filter removes any sudden variations in pixel intensity. Thus, any noise in the form of very high or very low pixel intensity is averaged out though the resulting image may look fuzzier. When expressed in frequency terms, images with rapid changes in intensity across different pixels are said to have a higher frequency. This filter therefore lowers the rapid change in frequency and can be thought of as a low-pass filter, i.e., it allows lower frequency variations to pass through. Where this filter might be useful is in making the images with a lot of noise look better to the eye. For example, an image acquired from a camera at extremely low light levels or low fluorescence signal may have lot of noise in the picture. Applying this filter to the image will make the picture appear better and may be a good first step in analysis to help the scientist, Table I.

Edge Detection

Edge detection is one of the more fundamental operations which is probably the most common operation performed by the human eye and, hence, the edge enhanced image sometimes appears sharper to the observer. There are many operators that are used and only two of them will be discussed—Laplacian and Sobel.

Laplacian

Applying a Laplacian kernel to the image highlights the edges in the image. The sharper the transition from dark to light the brighter the transition point will be. Each of the values in the kernel is multiplied by the corresponding image pixel intensity values and the sum of nine multiplications is the center pixel. The kernel works by being center weighted and will thus enhance the centermost pixel while the neighboring pixels will have little influence. But since the coefficients all add up to zero, only the transition point will be accentuated and uniform areas will calculate to 0 (black). This kernel is omnidirectional—it does not matter along which axis the transition takes place—it will highlight axis from 0 to 360 deg.

Sobel

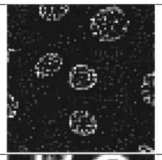
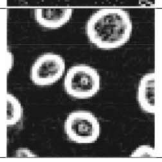
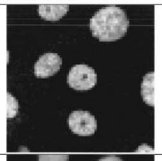
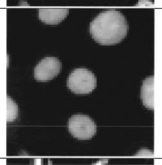
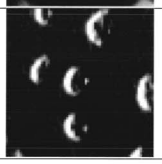
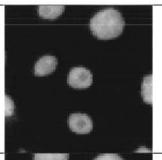
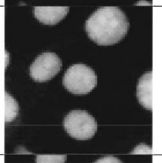
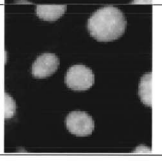
Another edge detection filter but the way it is implemented is to calculate a vertical edge followed by horizontal edge and then sum the result. The vertical and horizontal edge detectors can be used independently too if required. Analogous to Laplacian, each value in the kernel is multiplied by its respective image pixel intensity value and the sum of nine multiplications results in the center pixel. This kernel is weighted along any one axis and this higher pixel value is amplified in only one direction. For example, the operations for vertical operator in vertical axis leads to a positive value but in horizontal direction it tends to zero. This operator is not as sensitive to noise as Laplacian but on the other hand it might have a problem with fine details in biological images.

Though these are omnidirectional operators, other operations like Prewitt gradient can detect edges along any particular direction too.

SHARPENING OR HIGH PASS FILTER

In an image, each pixel can have distinct values. The change in pixel intensity values from a pixel to its adjacent pixel can be thought of in terms of frequency. Thus, adjacent pixels change rapidly from one another signifies high frequency. Filters can be designed to influence a specific range of frequencies. It is possible to create a high pass filter which accentuates rapid change while being unaffected by regions of uniform intensity. The way to understand the sharpening filter is to appreciate that the center pixel has the highest value and will have the largest influence on the image as shown in Table I. In implementation, it is similar to Laplacian, all the values in the kernel are multiplied by the pixel intensity values and the sum of nine multiplications computes to the center pixel. Thus, a change in the pixel intensity accentuates a high center value and at the same time the neighboring pixels obtain a lower than the average intensity value, thus highlighting the change. This operation makes

Table I. The picture is a cellular preparation stained for DNA in the nucleus with Hoechst 33342. The original source image (A) was processed with different 3×3 kernel operators to show the effect of kernel operations. The images were cropped similar to the source image (B) to show a close up of the effect of different operations. The name of each operation is followed by the kernel and the result image. Often, kernel filters are combined and used in different strengths. This example shows a single operation performed on the starting image. The image (C) is a binary image of (B) using a fixed threshold to detect bright nuclei and is used as a starting point for illustrating binary dilation and erosion in which objects have greater intensity value than the background.

Image processing filter or kernel operator name	Kernel used for operation.	Alternative kernel or kernel used for similar operations	Resulting image after applying kernel on source image (B)	Comments	Biological image processing uses.
Edge detection: Laplacian kernel:	-1 -1 -1 -1 +8 -1 -1 -1 -1	0 -1 0 -1 +4 -1 0 -1 0		These two kernels differ in kernel corner value, but work similarly. The central value is a positive sum of surrounding values.	Detect cells against a background or find punctuate spots within cells.
Edge detection: Sobel using directional operators	-1 0 -1 -2 0 -2 -1 0 -1 Edge detection along Vertical axis	-1 -2 -1 0 0 0 1 2 1 Edge detection along horizontal axis		Sobel filters can detect edges in particular directions and can be used additively for edge detection.	Besides cell detection can be used to determine fibers along any one axis.
Sharpening or high pass operator:	-1 -1 -1 -1 +9 -1 -1 -1 -1	-1 -1 -1 -1 +17 -1 -1 -1 -1 Unsharp mask		Both the kernels work but the strength varies with the kernel used – the example shown is after processing with the left kernel.	Use to correct out of focus pictures or highlight local bright spots or organelles.
Local equalization:	1 1 1 1 1 1 1 1 1 Divide kernel result by 9.			This has an effect of local averaging. In this case each pixel is processed by kernel and the result divided by 9.	Used to reduce fluorescent noise & as a first pass before edge detection to find large objects
Directional edge enhancement along a particular axis.	1 1 1 -1 -2 1 -1 -1 1 For North –East direction use this kernel	-1 0 -1 -2 0 -2 -1 0 -1 For Vertical axis use this kernel		The example shown is for vertical axis edge detection. See also: Sobel kernel.	These filters works for detecting tissue or fiber along any particular axis.
Erosion : For grayscale images	0 0 0 0 0 0 0 0 0 Min (kernel)	Explanation: The center pixel of result image is the **minimum** value of the 9 pixels in the 3 x 3 matrix of starting image.		Erosion expands the dark areas while reducing the bright areas.	Used to highlight intracellular unstained organelles or separate touching cells/nuclei.
Dilation: For grayscale images.	0 0 0 0 0 0 0 0 0 Max (kernel)	Explanation: The center pixel of result image is the **maximum** value of the 9 pixels in the 3 x 3 matrix of starting image.		Dilation expands the bright areas while reducing the dark areas.	Used to highlight brightly stained organelles or "smooth" bright cells.
Close	Dilation operation followed by erosion operation.			Dilation and erosion used in various combinations or strengths to extract the features of interest.	Mainly used to clean up boundaries or objects and separate touching cells/nuclei.

the image appear sharper and removes haziness in the image. It is of interest to note that the edge detector Laplacian kernel is also a high pass filter since it emphasizes the change in pixels intensity.

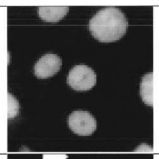
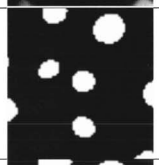
Unsharp Mask

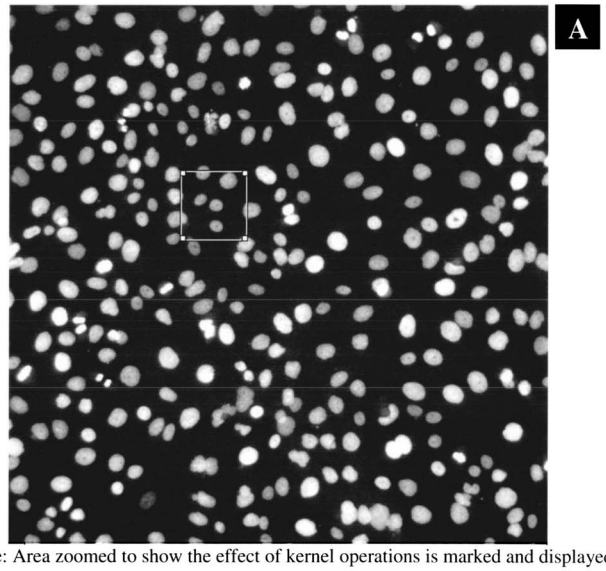
This is another sharpening or high pass filter with similar results and similar logic except due to a high center pixel value it leads to a nonzero sum of operations and thus might even brighten the image.

Dilation and Erosion

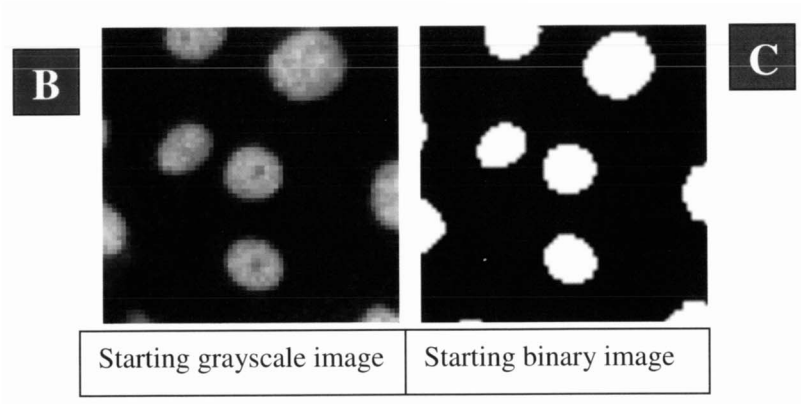
These operations have been used primarily with binary images to remove or smooth out surfaces though the method is slightly different with grayscale images. In explaining this method we will assume that the object of interest is bright

Table I. (*Continued.*)

Open	Erosion operation followed by dilation operation.			Dilation and erosion used in various combinations or strengths to extract the features of interest	Mainly used to clean up boundaries or objects and separate touching cells/nuclei.
Erosion: for binary images: (These involve "logic" operations, and not arithmetic calculations.)	1 1 1 1 1 1 1 1 1	Explanation: The center pixel of result image is **1** only if all the 9 pixels in the 3 x 3 matrix of starting image = **1**. Else center value = 0.		Binary operations are done to clean up outlines or masks of found objects. Erosion reduces the bright objects and expands the dark objects.	Erosion is used to separate touching cells or nuclei.
Dilation: For Binary images: (These involve "logic" operations, and not arithmetic calculations.)	0 0 0 0 0 0 0 0 0	Explanation: The center pixel of result image is **0** only if all the 9 pixels in the 3 x 3 matrix of starting image = **0**. Else center value = 1.		Binary operations are done to clean up outlines or masks of found objects. Dilation reduces the dark objects and expands the bright objects	Dilation is used to expand the mask to include cellular structures missed by the mask.



Original image: Area zoomed to show the effect of kernel operations is marked and displayed below.



(high intensity values) and the background is dark (low intensity values) for both binary images and grayscale images. The individual operations have different effects upon repeated application or in alternating cycles and lead to more pronounced effects.

Binary Dilation

This has the effect of enlarging the edges of bright objects and erodes the dark ones. This also uses the kernel operations as shown in Table I. The value of the center pixel is determined by comparing the value of the 9 pixels in the

image with the nine pixels in the kernel and whenever any value in the kernel or the input pixel is 1 (bright) then the center pixel gets the bright (1) value. Otherwise, the result is a black pixel. The effect of such an operation would be to expand the bright (1) values since having a bright neighbor would in effect make the center pixel bright. Thus, when we use this operation in binary image the white objects grow in size. This operation is useful in getting rid of black areas as well as joining bright objects together. It can also be used in filling holes in bright objects.

Binary Erosion

This operation erodes the edges of bright objects and enlarges dark ones. This uses a different kernel from that of binary dilation shown in Table I. In this case the value of the center pixel is determined to be bright only if all the nine pixels match that of the kernel which is all bright (1). Thus, if any one pixel in the background is dark (0) then the value of the center pixels is dark. This thus converts single bright pixels to dark and expands the dark regions. When combined with a compensating dilation this operation is useful in separating bright attached objects and smoothing edges.

Grayscale Dilation and Erosion

These are similar to the binary operation as the dilation expands the white regions whereas the erosion decreases the white regions. Dilation and erosion are implemented by adding the contents of the mask, which is zero, to the 3×3 matrix from the source image. In dilation the maximum value becomes the center pixel value whereas in erosion the minimum value becomes the center pixel value. Thus, in dilation the effect is that the center pixel tends to expand and become as bright as the neighbor whereas in erosion the central pixel tends to decrease in intensity with the neighboring pixel. They are used for similar purposes to binary dilation and erosion. The effect of many dilation operations is to make all the pixel intensity in the image equal to maximum intensity value. Similarly repeated erosion will make all the intensity values equal to the minimum value of the original image.

Opening and Closing

The combination of erosion followed by dilation is called opening. Opening darkens small objects and is one of the best ways to remove noise within the image. Closing, in contrast, is dilation followed by erosion. Closing brightens small objects and can fill small holes within objects. Both of these operations are used to smooth jagged objects if required since they will both remove pixel anomalies.

There are other filters that are useful in separating selected objects. The watershed filter, like the opening operation, is also useful in separating touching objects. The top hat filter and well filter are peak and valley detectors, respectively, and are useful in finding sharp transitions in images. These filters are derivations of the kernel operations mentioned above. For example, if an "opened" image is subtracted from the original image, the resulting image shows only peak intensity values similar to a top hat filter. In conclusion, many of these filters are usually used in combination to derive the most benefit from image processing.

FREQUENCY DOMAIN OPERATIONS

One of the common methods to remove repetitive noise or analyze regular patterns in images uses a technique called Fourier transform and its computationally fast equivalent—the fast Fourier transform.[8] This technique uses Fourier's theorem which states that it is possible to form any one dimensional function as a summation of a series of sine and cosine terms of increasing frequency. The simplistic, nonmathematical explanation of the Fourier transform in one dimension is that it is a way to represent a line using the sum of multiple frequencies. Thus, summation of tens or hundreds of frequency curves can recreate the original line which can be represented in terms of the frequencies used and their respective phases. This method in two dimensions can be used to represent any image as a set of frequencies along the two axes that together makes up the spatial image. The Fourier transform converts the image to two parts—the magnitude and the phase. These can be graphically drawn as an image as shown in Fig. 4(e). The magnitude part is drawn as the pixel intensity and the phase part is drawn along the vertical and horizontal axis. Thus, in an image with a repetitive pattern—the frequency representing that pattern will be highlighted as shown in Fig. 4(e). Selectively removing the frequency will remove the noise, Fig. 4. The frequency domain operations have other uses too such as image compression and image orientation.

FUTURE OF IMAGING

With the advent of bigger CCD arrays, higher resolution and cheaper computing and storage, imaging technology is finding more and more uses in industrial style imaging: from high content high throughput screening to industrial inspections of soda cans. This large amount of image data will require unique methodologies in image analysis and database technologies. Currently, the method of storing an image requires some form of annotation such that the image can be retrieved easily using conventional string searches. For example a cellular image can be stored with an annotation such as "treated with Colchicine" and we might easily search for "Colchicine." However, in a high throughput assay it would be tremendously useful if we could search for patterns such as "find cells with disrupted microtubules." Unfortunately, current methodology would require processing each and every image to find patterns of normal and disrupted microtubules, assuming of course the sample had been stained for microtubules. And, if we decided that we wanted to look for intact nuclei as opposed to necrotic nuclei then that would require another set of processing.

The kernel operations using smaller arrays tend to work as mentioned before. As the matrix gets bigger the operations tend to be more selective for features of the matrix dimensions. Therefore, a 21×21 matrix will be specific for features that are 21 pixel dimensions while ignoring smaller objects. Similarly, a 3×3 matrix operation will appear to have a very different effect on an image with eight million

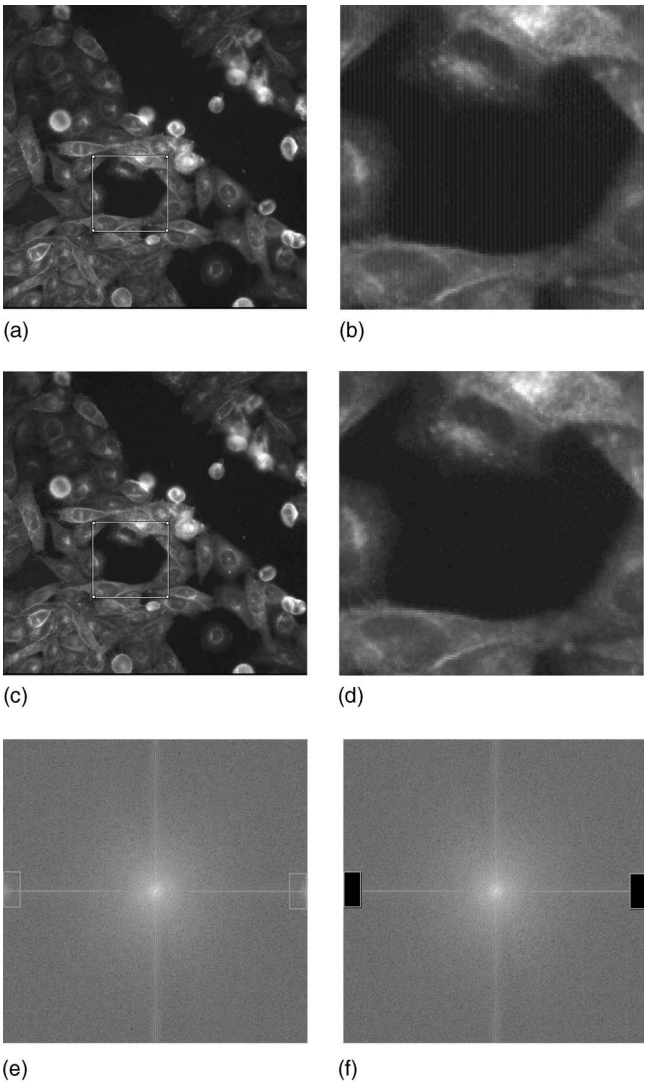


Figure 4. A fluorescent image of a group of cells was captured with a CCD camera (a) when zoomed, (b) showed a series of patterns possibly derived from the CCD elements. To remove this periodic noise a Fast Fourier transform of the image was created (e). A box was drawn around the regions where the noise frequency was expected (e) and the box filled with black (f) to eliminate those frequencies. The inverse transform of (f) yielded the image (c) which when zoomed (d) showed that the noise had been specifically eliminated.

pixels as opposed to one million pixels. So one has to be careful in using matrix operations for a particular microscope magnification, camera resolution and image depth.

Another technology that will be very useful in imaging is to use field programmable gate arrays (FPGA) for image processing.[9–11] At present, the CCD captures the image, transfers it to the computer in which a micro processor calculates the results one pixel at a time to come up with the result. In parallel architectures, multiple processors handle the task of working on different parts of the image or different images to give the result faster than a single computer. Whereas, FPGA can be thought of as a computer hardwired to do one thing extremely fast though it can be reprogrammed to do something else too using the same hardware. Thus, first a FPGA is programmed which would make a series of circuits in a particular configuration on a virtual circuit board. Once programmed, the FPGA can calculate the result almost instantaneously. Thus, it may be possible to have multiple circuits, with each circuit representing one pixel do instantaneous calculation and thus come up with the result. When multiple operations are required the circuit can be reprogrammed again or multiple circuits used to get almost instantaneous results. These arrays are being built capable of direct attachment to CCD cameras thus replicating the image processing that happens in our eye.

CONCLUSIONS

Image processing as it is used in biology has been a new development even though many of the algorithms used in image analysis have been mathematically well understood for a long time. This is because of three major developments. One, with cheaper processing power it is possible to do thousands of computations quickly to derive meaningful data from images and/or store some of these images for off-line processing. Second, automation to prepare and treat biological samples with multiple chemical entities has made it possible to bring industrial style operations to biological enterprise. Third, biological systems have been better understood with the availability of unambiguous fluorescent probes labeling specific cellular compartments with superb fluorescent properties such as green fluorescent protein allowing observation of transfected proteins in living cells.[12–15]

However, with all the developments, the critical and sometimes the rate limiting component of any image analysis or image processing in biology are still the identification of objects of interest. Only when the cells, tissue, or the organelle have been identified can the components be measured. For instance, the rate limiting step in image based pathology is not the ability to prepare more tissue sections or process data or even stain the specimen, instead it is the ability to identify and analyze the objects of interest. Those difficult steps are where image segmentation methods using kernel and morphological operations, presented in this review, are extremely useful. These tools are complex and sometimes require judicious use of combining various methods to successfully identify objects. But even then, they may work only within a defined set of conditions like a specific microscope magnification with precise sample preparation.

The use of kernel methods can also be challenging and it is possible that the complexity of image segmentation has encouraged the development of fluorescent technology for biology rather than traditional histological techniques. The big advantage of fluorescence from image analysis point of view is that the object is brighter than the background. This simplifies object identification considerably. Using a simple threshold function delineates the objects against the background. But, since biological objects have complex morphologies, it still requires the use of kernel operations to complete the segmentation process. Another advantage of using kernel methods with fluorescence is that it analyzes groups of pixels rather than single pixel values. So, the difference from the background can be easily highlighted.

Even with the availability of these kernel and morpho-

logical methods, it requires an understanding of the image to interpret the data. Maybe, for true and accurate identification segmentation and analysis of images will require a knowledge base at the backend to understand biological objects rather than mere pixel number crunching engine. Thus, the next big step in kernel and morphological processing will be more intelligent identification of images assuming we have resolved how to store and retrieve the large amounts of data that are generated.

ACKNOWLEDGMENT

The authors are thankful to the laboratory of Dr. Douglas K. Miller for sample preparation.

REFERENCES

[1] G. A. Baxes, *Digital Image Processing: Principles and Practice* (Wiley, New York, 1994), pp. 69–122.
[2] J. C. Russ, *The Image Processing Handbook*, 3rd ed. (CRC Press, Cleveland, 1999), pp. 227–304.
[3] J. Loebl, *Image Analysis, Principles and Practice* (Short Run, Exeter, 1985), pp. 84–122.
[4] W. K. Pratt, *Digital Image Processing*, 2nd ed. (Wiley, New York, 1991).
[5] A. K. Jain, *Fundamentals of Digital Image Processing* (Prentice Hall, Englewood Cliffs, NJ, 1988).
[6] K. A. Giuliano, J. R. Haskins, and D. L. Taylor, "Advances in high content screening for drug discovery", Assay Drug Dev Technol. **1**(4), 565 (2003).
[7] R. S. Gentile, J. P. Allebach, and E. Walowit, "Quantization of color images based on uniform color spaces", J. Imaging Technol. **16**(1), 12 (1990).
[8] C. M. Harris, "The fourier analysis of biological transients", J. Neurosci. Methods **83**(1), 15 (1998).
[9] L. Salwinski and D. Eisenberg, "In silico simulation of biological network dynamics", Nat. Biotechnol. **22**(8), 1017 (2004).
[10] U. Meyer-Base and H. Scheich, "Artificial implementation of auditory neurons: A comparison of biologically motivated models and a new transfer function oriented model", Biol. Cybern. **77**(2), 123 (1997).
[11] B. Fagin, J. G. Watt, and R. Gross, "A special-purpose processor for gene sequence analysis", CABIOS, Comput. Appl. Biosci. **9**(2), 221 (1993).
[12] BD Biosciences Ds Red product line: http://www.bdbiosciences.com/clontech/archive/JAN02UPD/pdf/DsRed2.pdf.
[13] T. F. Massoud and S. S. Gambhir, "Molecular imaging in living subjects: Seeing fundamental biological processes in a new light", Genes Dev. **17**(5), 545 (2003).
[14] F. Eisenhaber and P. Bork, "Wanted: Subcellular localization of proteins based on sequence", Trends Cell Biol. **8**, 169 (1998).
[15] J. C. Simpson, R. Wellenreuther, A. Poustka, R. Pepperkok, and S. Wiemann, "Systematic subcellular localization of novel proteins identified by large-scale DNA sequencing", EMBO Rep. **1**(3), 287 (2000).