THEORY AND PRACTICE OF HISTOLOGICAL TECHNIQUES

Sixth Edition

John D. Bancroft

Formerly Pathology Directorate Manager and Business Manager
Queen's Medical Centre
Nottingham, UK
Email: jcbancroft550@hotmail.com

Marilyn Gamble

Formerly Quality Assurance Coordinator
Cellular Pathology Services
Kaiser Permanente
Regional Reference Laboratories
North Hollywood, CA, USA

CHURCHILL LIVINGSTONE
ELSEVIER

CHURCHILL
LIVINGSTONE
ELSEVIER

CHURCHILL LIVINGSTONE
An imprint of Elsevier Limited

First edition 1977
Second edition 1982
Third edition 1990
Fourth edition 1996
Fifth edition 2002
Sixth edition 2008
Reprinted 2008 (twice), 2011

ISBN: 978-0-443-10279-0

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

Library of Congress Cataloging in Publication Data
A catalog record for this book is available from the Library of Congress

Notice
Medical knowledge is constantly changing. Standard safety precautions must be followed, but as new research and clinical experience broaden our knowledge, changes in treatment and drug therapy may become necessary or appropriate. Readers are advised to check the most current product information provided by the manufacturer of each drug to be administered to verify the recommended dose, the method and duration of administration, and contraindications. It is the responsibility of the practitioner, relying on experience and knowledge of the patient, to determine dosages and the best treatment for each individual patient. Neither the Publisher nor the author assume any liability for any injury and/or damage to persons or property arising from this publication.

The Publisher

Printed in China
Last digit is the print number: 9 8 7 6 5 4

case of solid-state cameras, the increased intensity level sensitivity, as compared to the eye, accentuates this problem. Since all solid-state cameras use some type of software, either within the capture system or on the capturing computer to control the camera, this software frequently contains some type of 'field flattening' or 'background subtraction' mechanism to correct this variation in intensity from center to edge of the image. Additional details of requirements for acquiring images through microscopes using electronic cameras can be found in Shotton (1993).

IMAGE ANALYSIS

Overview

Image analysis is a broad term that may be defined quite differently by those working in diverse fields. Originally, image analysis was used to describe the extraction of numerical information from pictures. Since the process of placing pictures into a form that could be analyzed digitally was cumbersome, and the computers used to analyze such pictures were slow, image analysis was performed 'off line' and quite often at sites far removed from where the image was originally recorded. As software techniques for image analysis improved, and computers became faster and more affordable, image analysis became more widely used, in many fields and disciplines. Image analysis encompasses many areas: machine vision, graphic arts, pattern matching, photometry, optical character recognition, surveillance, security, and scores of others. While many of the same techniques are used in each of these areas (at the software level), we will restrict the remainder of this discussion to use of image analytic techniques to extract numerical data from microscopic preparations. The emphasis will be on transmitted light preparations, although in many cases identical approaches are used for fluorescence preparations.

The minimal requirement for image analysis of microscope preparations is a microscope equipped with a camera that can capture and transmit images to a computer equipped with suitable image analytic tools. The camera requirements have been discussed above. Computers suitable for image analysis range from RISC-based workstations to personal computers, either IBM PCs (or clones) or Apple PCs. A variety of sophisticated image analytic tools (programs) are available for each of these platforms. In addition to commercial offerings, there are a number of freeware or shareware programs available for personal computers. Among the best known of these is the program originally designed for the Apple computers, named NIH Image. There are now versions of this program available for the IBM (Windows) computers as well. A recent addition to the list of available image analytic programs is ImageJ, written by the original author of NIH Image. This program is also freely available, and, because it is written in the JAVA language, can run on any computer which supports JAVA (http://rsb.info.nih.gov/ij/.)

All image analysis programs must provide mechanisms to display images, read images from a source (camera or storage), and ultimately save the image and any derived data to storage. In modern computers, these functions are part of a GUI (graphical user interface) that permits the user actually to see the image and the various alterations to it during and after various image analytic or manipulation steps.

As camera resolutions increase, they often exceed the display capability of many computer displays. As an example, consider a common camera resolution available today, 1024×1310. The actual image from such a camera is larger than the common display resolution of many computers, which may be 800×600. Another common display resolution is 1024×756. In both cases, the larger image is displayed completely on the monitor. This is accomplished within the display program, by simply reducing the image to fit within the monitor resolution. Therefore the displayed image may not accurately represent the 'real' image that has been captured and is available for analysis. Some capture/display programs provide tools to permit the user to display the image at actual resolution, even though only a portion of the image is seen on the screen. Such programs allow the user to scroll over the image in order to see the entire image. Note also that many output devices, such as printers, actually reduce the size of the image, and therefore lose resolution as compared to the original image. This display resolution, which may be different from the actual image resolution, is one reason why high-resolution images may not appear as 'sharp' when viewed on a display device.

It is important to realize that an image, to the computer, is simply an array of values of the individual pixels. For 8-bit monochrome images, this would be a sequence of numbers, with values for each ranging between 0 and 255. Image file storage formats specify the number of pixels per row, and the total number of rows. This information is part of the 'header' information in the file storage, and is required for proper display and analysis of the images. The user ordinarily does not have to worry about this information, since it is taken care of transparently by the image software. Because the computer software considers the image to be an array of X by Y dimension, any pixel in the image can be individually addressed if its location is known within the array. In the case of color images, the actual image is stored as a sequence of colored pixels, i.e., red, blue, and green. To extract the 'green' image, one would read every third pixel from the file. Note that there are a variety of formats for image file storage, and the programmer should be sure to verify the 'bit order' in use prior to attempting to extract specific information.

IMAGE ANALYSIS PROCESSES

Point processes

Most forms of image analysis of microscope images start with a category of operations classically defined as *point processes*. These process are relatively simple, yet are basic to most image operations. A point process acts on an individual pixel within the image, and may modify the value of that pixel depending on the previous value. A common use of a point process is to change the value of each pixel to some other value, depending on the original value. Such an operation may make use of a LUT (look-up table). A common use for such an operation is in a pseudo-color operation, where a gray scale image is divided into a number of 'levels', i.e., all pixel values between 0 and 20 might be colored 'red', all values between 21 and 50 might be colored 'blue', etc. Since the human eye recognizes color variations much more easily than density variations, such a point operation might well make interpretation of a gray scale image easier.

Point processes are used to change the overall intensity of an image. Suppose an image is captured, and the background appears too dark. By adding a constant to every pixel, the result is an image that looks brighter. (Note: This depends on the scale being used for display of images. In this case, it is assumed that '0' is black, and '255' is white. There are systems in which this is reversed.) Often, in the process of color balancing the individual color planes of a color image, it is a point process that is used to set the 'clear' or 'background' pixels to 'white'. Point processes can also be used to convert an image to a negative of the original image. In this process, each pixel is mapped to the value it would have if the scale of black to white values were inverted (black = 255, rather than 0). This is quite useful if the image is analyzed with a system different from the one used for capture. It is also a useful function for many intermediate image manipulations, particularly where images may be combined with one another.

Image contrast stretching is another example of a point process. In a contrast stretch (image equalization), the range of gray levels in the image is expanded. In many specimens, the actual image values cover a relatively narrow range of the total available gray values. As an example, in a nuclear preparation stained for DNA with the Feulgen procedure, the total gray levels represented by the stained nuclei occupy only about 30% of the total available levels. By contrast stretching these gray levels to cover the entire range of available values (256 levels), additional details can often be seen and/or measured in such contrast-stretched images.

By far the most common point process in image analysis is image thresholding. This is used to *segment* an image into areas that have some particular interest, such as a particular staining pattern. The action of a threshold is simple. The user selects a particular gray level (generally with some type of interactive tool in the graphical user interface). The point process then sets all pixels with a value lower than this threshold value to '0', and all values above this threshold value to '1'. In other words, the image is converted to a binary image. While this simple threshold is sufficient for some purposes, such as determining the total area in the image that is above some level (generally, the area of the image that is 'stained' by whatever is being analyzed), the simple binary image is more generally used to combine with the original image to produce some type of 'mask'. A common implementation is to combine the binary image with the original image in such a way that all '0' or background pixels are left '0', while all '1' pixels are left with their original image value. Such an operation leaves the desired portions of the image visible, with the remainder eliminated from the image. A second threshold step is often performed, thresholding from the opposite direction. After this step, a group of objects of 'medium' gray level could be separated or *segmented* from both lighter and darker objects. Another common implementation of threshold point operations is to combine the two threshold operations with a pseudo-color or LUT operation, and simply place a transparent colored mask over the desired objects, leaving the entire original image visible.

Area processes

Area processes use groups of pixels either to derive information from the image or to alter the image in some specific manner. In general, area processes involve a small portion of the image, in a two-dimensional matrix. The matrix is generally made up of an odd number of 'row' pixels and an odd number of 'column' pixels (a convolution kernel). It is the pixel in the center of this matrix that may be altered after performing the area process. Many of the area processes are often referred to as *convolutions*. Convolutions commonly are based on matrix sizes of 3, 5, 7, or sometimes larger dimensions. In the area process, a convolution matrix is defined. The convolution matrix is placed over the image, and each pixel over which the convolution mask lies is multiplied by the number contained within the convolution mask. All of these multiplied pixel values are then summed, and the sum is used to replace the central pixel. The mask is then moved one pixel further along, and the process repeated (Fig. 31.2). Note that, in practice, the 'changed' pixel is used to construct a new image, since the process of convolution would fail if the image being analyzed were being altered during analysis. In other words, the convolution does not

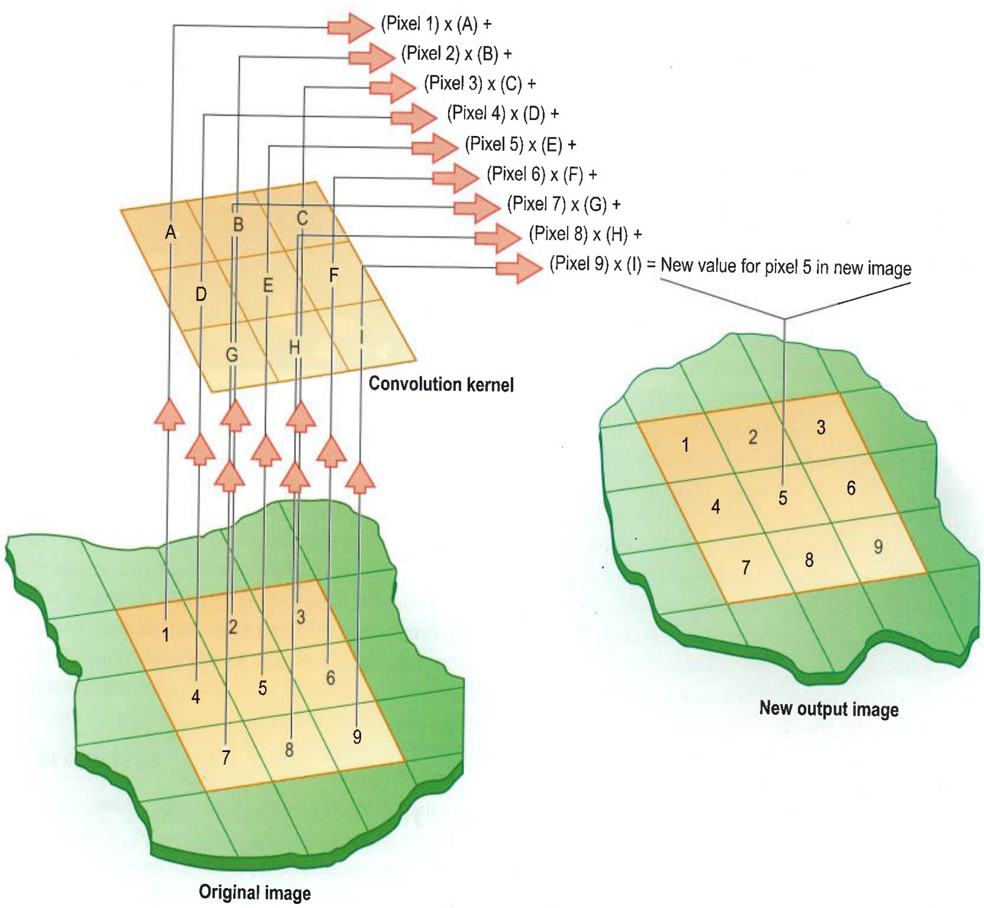

Fig. 31.2 The manner of implementation of a convolution. The diagram illustrates a convolution kernel of nine elements.

change the 'original' image, but creates a new, modified image based on the original (Fig. 31.3).

Area processes in general are often called spatial filtering operations, since they yield information about the rate of change of intensities within the image. In fact, it is these rates of change that are exploited by many common convolution filters. Typical area processes are those used for spatial filtering such as high-pass and low-pass filters. A low-pass filtered image will reduce the contrast of an image. Such an operation is often useful to remove unwanted noise spikes within an image. A high-pass filtered image increases contrast within the image, and is often used to improve the ability to detect edges or other structures within the image.

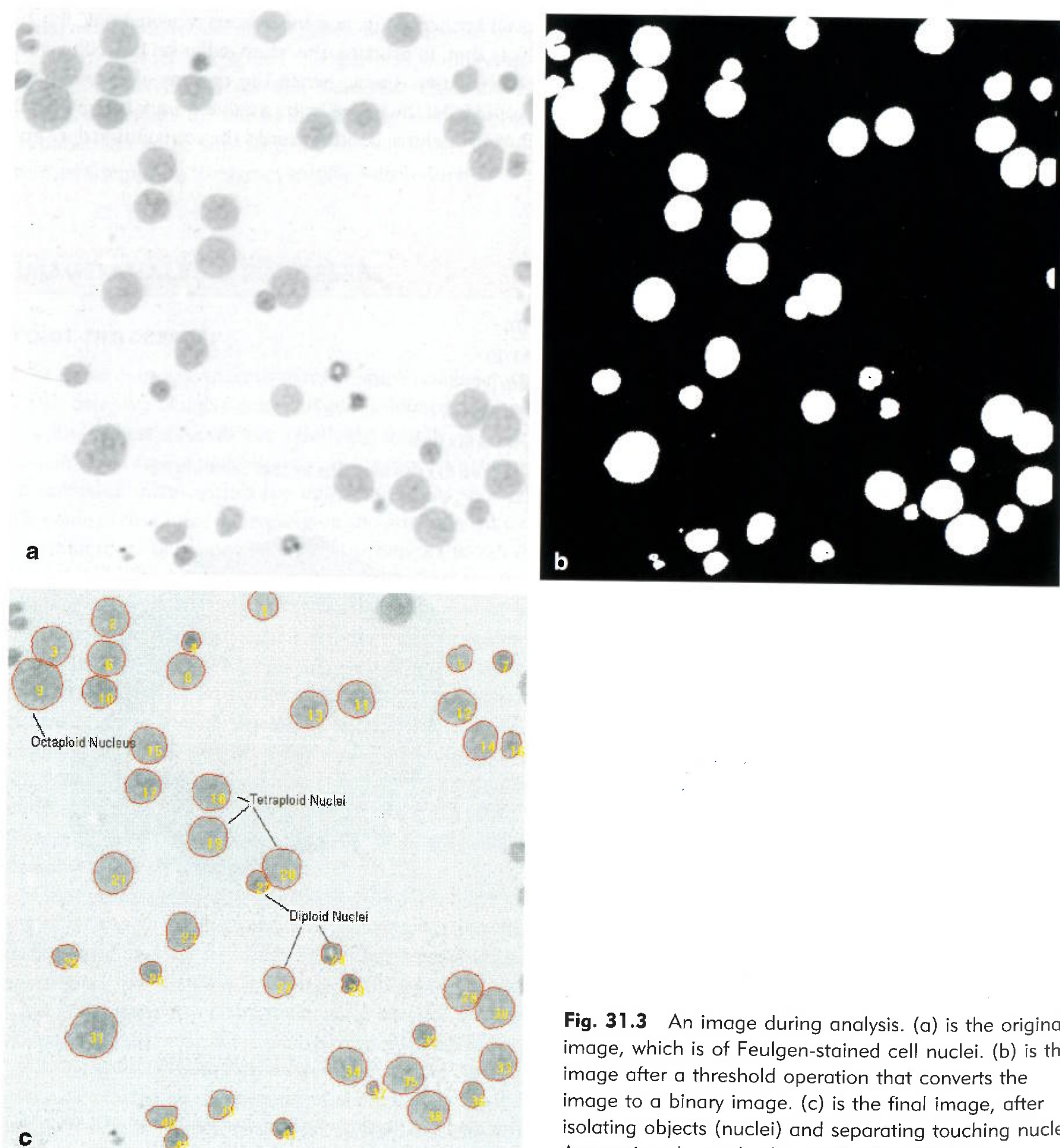

Fig. 31.3 An image during analysis. (a) is the original image, which is of Feulgen-stained cell nuclei. (b) is the image after a threshold operation that converts the image to a binary image. (c) is the final image, after isolating objects (nuclei) and separating touching nuclei. Annotations have also been added to this image.

An important type of spatial filtering is edge detection. A variety of convolution matrices are available to perform it. Often, some type of edge detection is used to perform segmentation within an image, particularly if the area to be segmented is close in gray value to other structures within the image, and thresholding is difficult.

While area processes are extremely important in image analysis, they are computationally intensive processes. As an example, a point process need only look at each pixel in an image one time. An area process, in the simplest case, must look at each pixel times the size of the convolution matrix. For a convolution matrix of 3×3, and an image of 1 million pixels, 9 million operations would have to be performed. For matrices of 7×7, 49 million operations would be required. In actuality, the number of operations required is somewhat greater than the figures given here, as impressive as they are. Because of the amount of computer processing required, image analysis requires fast computers, with large amounts of memory. Also, high-resolution, full-color images require a considerable amount of space for storage. A full-color image of 1024×1310 pixels will require almost 4 million bytes of storage (4 MB). While there are various methods of making images smaller (image compression), most of these forms of compression are 'lossy', that is, they discard image information, and this information cannot be retrieved from the stored image. Certain image storage formats allow a type of compression that is based on sequences of image data where all the pixels are the same (like large areas of background). This form of compression is called run length encoding, and does not discard any image information. However, for a typical image of a microscope specimen, where there are few or no 'constant value' areas, run length encoding may actually result in a larger image storage size than the original image. As a practical matter, any image intended for future analysis should not be stored in a compressed format, particularly in view of the low cost of large-capacity storage devices.

Frame processes

Both point processes and area processes treat the image as a series of pixels, and address each pixel in a specific manner. Frame processes, in contrast, operate on the entire image. Often frame processes use simple boolean (logic) operations to add, subtract, multiply, divide, or otherwise combine two images to produce a new third 'result' image. A common use of a frame operation is to correct a microscope image for uneven illumination. By collecting and 'temporarily' saving an image of the 'background' (when no specimen is present), the background image can then be subtracted from the specimen image. This will effectively remove any debris or other image-degrading elements that are inherent in the microscope and illuminator. A similar operation is commonly used in security systems, where a 'scene' image is collected at intervals. By subtracting the 'next' image from the previous image, any change in the 'scene' will be immediately recognized. A common biological application for a frame process would be to detect movement in a cell culture being observed at intervals. By tracking these changes over time, a 'trail' can be mapped and applied over the image to follow the movement of cells over time. Frame processes are common in many of the operators employed in image analysis systems, particularly with respect to boolean operations. They are frequently used in the display portions of image analysis systems, where they can be used to combine the results of various area processes or thresholding operations with the original image, in order to evaluate the effectiveness of a particular operation.

Geometric processes

Geometric processes are quite different from the processes discussed previously, since they are mainly used to reconstruct or correct images. Geometric processes actually move pixels within the image, and can therefore be used to correct defects such as geometric distortion. Geometric processes are used to rotate images, change scale, translate images, and produce mirror images. It is geometric processes that are used to interpolate images from 'real' size to a size that can be displayed on the monitor in use. Geometric distortion would include such image defects as a microscopic section that was attached to the slide in a manner that distorted normal morphology. In such a case, a geometric process could 'transform' the image to straighten or otherwise return it to the shape believed to be correct. Geometric processes are used to make the specimen 'look better', and to prepare the image for display or output to a print device. Until recently, geometric processes were not used extensively in microscope imaging, other than for display and printing. However, with the advent of microscope systems that can collect multiple images from various focal planes of the specimen (confocal microscopes), geometric processes have become much more widely used. These processes can be used to align 'stacks' of images which are used to reconstruct three-dimensional models of specimens.

Geometric processes can also be used to create mosaic images. A mosaic image is an image that is created from several smaller images. Using a motor-driven stage (scanning stage), an image system can travel over a slide, each time moving the exact width of the previous image, and collecting another image. Each collected image can then be 'added' to the previous one to create a large, mosaic image. With appropriate software, the area where these small images join can be a perfect, seamless match. Creation of such mosaic images requires an automatic focusing mechanism, to insure that each image collected is properly focused. While it has been suggested that such mosaic images can be retained in place of the original specimen, the resolution required to prevent loss of data (with current cameras) would mean the minimum objective magnification used to create the mosaic would be 20× and preferably 40–60×, and for large specimens this would create exceptionally large image files. Some 'virtual slide' microscopes produce image sizes that are hundreds of megabytes in size.

IMAGE ANALYSIS SOFTWARE

Many commercial image software packages are available, for both the PC and Mac personal computers. All of these offerings include the variety of image analysis processes described above, although there is little standardization of terminology for specific types of process. The user should work with each algorithm of interest, and verify that it is doing the desired function, as the implementation of common algorithms does differ from one software program to another. In general, these software systems are organized as a way to display an image. The image may be either captured from a camera, or retrieved from storage. Once the image is available, the user can select, through menus or tool bars, a variety of image manipulation tools. When the tool is applied to the image, the results can be seen immediately. Most systems also provide a mechanism to back up or undo, in case the result was not satisfactory. Such an image system is an ideal learning tool, and a user should expect to spend some time becoming familiar with any new image analysis system.

In addition to commercial software packages, there are a number of 'freeware' or 'shareware' image analysis packages available. One of the best known of these is NIH Image, written for the Mac platform, and freely available worldwide. There is also a version of this software for the PC platform. Recently, the original author of NIH Image produced a new image software package, ImageJ. This software is written in the Java language, and offers the advantage of running on any computer. ImageJ is available at no cost, and is constantly being upgraded and expanded.

Most vendors of commercial image analysis software packages provide support through user groups, and these groups are a rich source of assistance with image analysis problems. There are also a number of generic image analysis groups that maintain discussions via the internet, and these groups can provide assistance with specific image analysis problems. The internet is also a rich source of images, and a number of histology and pathology image archives are available. Note that currently most of these images are available as JPEG images. The JPEG format is a file storage format used for many internet images, and is a compressed image format (with compression based on a discrete cosine transform). Such compression is a 'lossy' format, which means that image data are lost during compression and cannot be retrieved. This is a critical issue for some types of image analysis, but is not necessarily a problem for simply viewing the image. The newer form of JPEG image is JPEG 2000. This compression uses a wavelet transform, and produces somewhat better visual results. However, it is still a 'lossy' compression format, and therefore not suitable for many types of image analysis.

SPECIMEN ANALYSIS

The goal of most image analysis of microscope images is the generation of numerical data that describe some aspect of the specimen. If the specimen is stained for some specific constituent, and the mechanism with which the stain interacts with the constituent is known, then it may be possible to use photometric techniques to produce numbers corresponding to the actual amount of the constituent present in the specimen, or in selected portions of the specimen. Generally, there will be more than one area of interest within the specimen. In the case of a specimen stained for cell nuclei, one could expect several hundred nuclei in a single image. In such a case,