Chapter 2
Digital Images

2.1 Introduction

What is a digital image? Interestingly this question does not have a simple answer. Consider, for example, this image of a familiar Australian landmark (Fig. 2.1).

Fig. 2.1 Is this a digital image? No, it's an ink image. The intensity data was stored and manipulated in digital format between the time of capture and the time of printing of this page. Was a digital camera used? There is no way to tell from the ink in this image

What does it mean if we say this is a digital image? The image is printed on the page with ink so there is nothing 'digital' in what we see when we look at the image on the page. Even if the resolution were so poor that we could see pixelation we would not be seeing actual pixels (the smallest elements of image information) but a representation of them. There were many steps between the capture of the visible light image and the printing of the image on this page. It was originally captured with a digital camera, which means the continuous pattern of light being reflected off the Sydney opera house and the harbor bridge was initially recorded as an array of electric charges on a semiconductor light sensor. The amount of charge on each element of the sensor was then measured, converted into a binary number, copied into the memory of the camera, processed in some way, and then written onto a

R. Bourne, *Fundamentals of Digital Imaging in Medicine*,
DOI 10.1007/978-1-84882-087-6_2,

compact flash card. Later the image was downloaded from a card onto the hard disk in a computer, processed with some software, then stored again in a different format on a hard disk. It would be a very long and tedious story if we traced the path of the image all the way to this printed page. The point to consider is that, at almost every step of this process, the image data would have been stored on different electronic or optical media in different ways. Thus a single image data set can have many different physical forms and we only actually *see* the image when it is converted to a physical form that reflects, absorbs, or emits visible light.

We might broadly separate images into two categories – the measured, and the synthetic. A measured image is one acquired by using some device or apparatus to measure a signal coming from an object or a region of space. Obvious examples are photographs, X-ray images, magnetic resonance images, etc. In contrast, a synthetic image is one not based on a measured signal but constructed or drawn. Typical examples are diagrams, paintings, and drawings. Of course these two broad categories overlap to some extent. Many synthetic images are based on what we see, and many measured images are manipulated to change how we see them and to add further information – lines, arrows, labels, etc.

2.2 Defining a Digital Image

In a digital camera the subject light ‘pattern’ is focused by the lens onto a flat rectangular photosensor and recorded as a rectangular array of picture elements – *pixels*. In the sensor a matrix of photosites accumulate an amount of charge that (up to the saturation point) is proportional to the number of incident photons – the intensity of the light multiplied by the duration of the exposure.

Just how different is this ‘digital’ process from the so called ‘analog’ photochemical film process? Not very. With a film camera the subject light pattern is recorded as an *irregular* matrix of silver granules, the film grain, embedded in a thin layer of gelatin. Development of a film image is the chemical process of converting light-activated silver halide grains to an emulsion of silver metal with stable light reflection and transmission properties. By analogy, ‘development’ of a digital camera image is the process of converting the charge stored on the semiconductor light sensor to a binary array stored on stable electronic media. The stored digital image data is then equivalent to a film negative – it is the stable raw data from which a visible image can be repeatedly produced. Since this happens automatically inside the camera it is not something we pay much attention to.

Whether image contrast is stored as an irregular array, as in film, or a regular array, as in a digital recording, is of no significance in determining the information content (Fig. 2.2). However, it is *much, much* easier to copy, analyze, and process a digital data array.

One of the main operational differences between digital and film sensors is that digital sensors are relatively linear in their response to light over a wide range of exposures while films are generally linear only over a narrow range of exposures. This makes film harder to use because there is much more potential for exposure

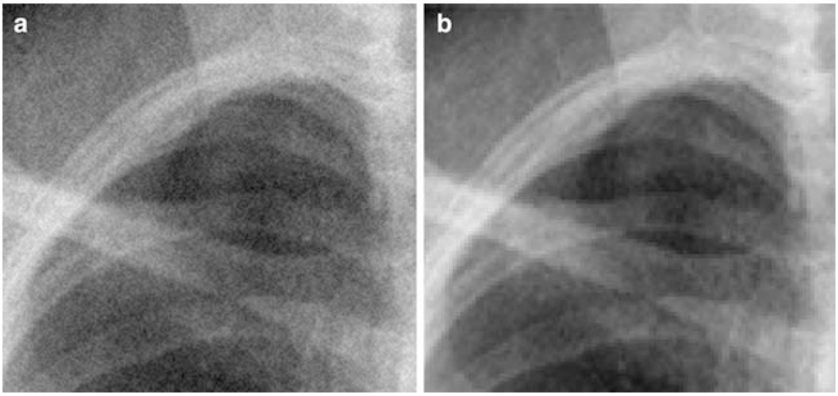


Fig. 2.2 Illustration of the *lack of* difference between the way film and a direct digital sensor record image information. Image **a** represents the *random* array of silver granules that provide optical contrast in a film recording of image data. Image **b** represents the *rectangular* array of pixel intensities (converted to some display medium) that provide optical contrast in a direct digital recording. There is no significant difference in the *information content* of the two images

errors that lead to either inadequate or excessive film density in the developed image. On the other hand the large dynamic range of digital X-ray detectors means that high exposures still give good quality images. This has led to ‘exposure creep’ – a gradual increase in routine exposures and unnecessarily high patient doses.

Another, less direct, analog of the chemical process of film development is the process of *image reconstruction*. Image reconstruction is the term used to describe the methods of formation of anatomical images from the raw data acquired in tomographic (cross-sectional) medical imaging devices. Since the raw data is not a cross-sectional image the process might be more appropriately named *image construction*, however, we will stick to the common usage in this text. Either way, image (re)construction depends on the processing of raw digital data to create a 2D or 3D image in which the position of objects in the image correspond to their positions in the subject – they are not superimposed as in a projection image.

Where does this leave us in defining a digital image? As a working definition we might simply say that *a digital image is an encoding of an image amenable to electronic storage, manipulation and transmission*. This is the huge advantage of digital images over film images. There are numerous ways to do the encoding, manipulation and transmission, each method having specific advantages and disadvantages depending on the intended use of the image. We will definitely not discuss these methods comprehensively, nor in detail, but important points of relevance to medical images will be covered.

No matter how a digital image is stored or handled inside a computer it is displayed as a rectangular array (or matrix) of independent pixels. Of course the objects we image are not rectangular arrays of homogeneous separate elements. The original *continuous* pattern of signal intensity coming from the imaged object is converted by the imaging system into a rectangular array of intensities by *discrete* sampling. Each element of the rectangular array represents the average signal intensity in a small region of the original continuous signal pattern. The size of each small region from which the signal is averaged is determined by the geometry of the imaging system and the physical size of each sensor element.

It is important to remember that the signals from separate regions of the imaged object are not perfectly separated and separately measured by an imaging system. All imaging devices ‘blur’ the input signal to a certain extent so that the signal recorded for each discrete pixel that nominally represents a specific region of sample space always contains some contribution from the adjacent regions of sample space. This inevitable uncertainty about the precise spatial origin of the measured signal can be described by the *Point Spread Function* (PSF) – an important tool in determining the spatial resolution of an imaging system. The PSF describes the shape and finite size of the small ‘blob’ we would see if we imaged an infinitely small point source of signal.

The raw image data has a specific size – m pixels high by n pixels wide. Put another way, the image matrix has m rows and n columns. In many image formats the pixel data is not actually *stored* as an $m \times n$ rectangular array. Because most images have large areas of identical or very similar pixels it is often more space and time efficient to store and transmit the pixel information in some compressed form rather than as the full $m \times n$ array. An image stored in this way must be converted back into an $m \times n$ matrix before display.

So far we have discussed only 2D images. In many imaging modalities it is common to construct 3D or *volume* images – effectively a stack of 2D images or slices. This does not change our conception of a digital image – 2D or 3D, it is still a discrete sampling where each pixel or *voxel* (volume element) represents a measurement of the average signal intensity from a region in space.

When we open a digital image file the computer creates a temporary $m \times n$ array of pixel data based on the information in the image file (if it is a color image then a series of $m \times n$ arrays are created – one for each base color, e.g. red, green, and blue in the case of an RGB image). This array is the one on which any image processing is performed, or it provides the input data for image processing that outputs a new ‘processed image’ array. If the image is to be displayed on a computer monitor then the rectangular array of pixel intensity and color information is converted into a new array that describes the intensity and color information for each pixel on the monitor. There will rarely be a one-to-one correspondence between the raw image pixels and the monitor pixels so the display array will have to be interpolated from the original array. Alternatively, if the image is to be printed on a solid medium such as paper or film, then the array of pixel information is converted into a new array that describes the intensity and color information for each printing element. On a sophisticated inkjet printer there may be ten different inks available and the print head may be capable of ejecting hundreds of separate ink droplets per centimeter of print medium. The data array that is required for printing is thus very much larger than the original image array. It contains a lot of information very specific to the particular image output device, but it need only exist for the duration of the printing process and need not be stored long term.

2.3 Image Information

It should now be quite clear that because digital images are so easily stored, transmitted, and displayed on different media the physical form of a specific digital image is highly context-dependent. Much more significant than the physical form of a digital image is its *information content*. The *maximum* amount of information that can be stored in an image depends on the number of pixels it contains and the number of possible different intensities or colors that each pixel can have. The *actual* information content of the image is invariably less than the maximum possible. As well as the uncertainty in the spatial origin of the signal due to the point spread function, there will be some uncertainty about the reliability of the intensity or color information due to a certain amount of *noise* in the measured signal.

When we perform image processing we are sorting and manipulating the information in an image. Often we are trying to separate certain parts of the 'true' signal from the noise. In doing this we must be careful not to accidentally destroy important information about the imaged subject, and also not to introduce new noise or artifacts that might be accidentally interpreted as information.

2.3.1 Pixels

You might say that the fundamental particle of digital imaging is the pixel – the smallest piece of discrete data in a digital image. The pixel represents discrete *data*, not necessarily discrete *information*. Due to the point spread function, subject movement, and several other effects, information from the imaged object will to some extent be distributed amongst adjacent pixels (or voxels). When discussing color images we could separate the individual color components of each pixel (e.g. the red, green, and blue data that describe a pixel in an RGB image) but since we are mainly dealing with gray scale images in medical imaging we need not worry about this refinement here. However, we do have to be careful about the way we use the term 'pixel' in digital imaging, even after defining it as a 'picture element'. *Pixel* can mean different things in different contexts and sometimes conflicting contexts are present simultaneously.

A pixel might be variously thought of as:

1. A single physical element of a sensor array. For example, the photosites on a semiconductor X-ray detector array or a digital camera sensor.
2. An element in an image matrix inside a computer. For an $m \times n$ gray scale image there will be one $m \times n$ matrix. For an $m \times n$ RGB color image there will be three $m \times n$ matrices, or one $m \times n \times 3$ matrix.
3. An element in the display on a monitor or data projector. As for the digital color sensor, each pixel of a color monitor display will comprise red, green and blue elements. There is rarely a one-to-one correspondence between the pixels in a

digital image and the pixels in the monitor that displays the image. The image data is rescaled by the computer's graphics card to display the image at a size and resolution that suits the viewer and the monitor hardware.

In this book we will try to be specific about what picture element we are referring to and only use the term pixel when there is minimal chance of confusion.

2.3.2 Image Size, Scale, and Resolution

Shrinking or enlarging a displayed image is a trivial process for a computer, and the ease of changing the displayed or stored size of images is one of the many advantages of digital imaging over older film and paper based technology. However, technology changes faster than language with the result that terminology, such as references to the *size*, *scale* and *resolution* of an image, can become confused. We may not be able to completely eliminate such confusion, but being aware of the possibility of it should make us communicate more carefully. We may need to be explicit when we refer to these characteristics of an image, and we may need to seek clarification when we encounter images which are described with potentially ambiguous terms.

What is the *size* of a digital image? Is it the image matrix dimensions, the size of the file used to store the image, or the size of the displayed or printed image? The most common usage defines *image size* as the rectangular pixel dimensions of the 2D image – for example 512×512 might describe a single slice CT image. For very large dimension images, such as digital camera images, it is common to describe the image size as the total number of pixels – 12 megapixels for example.

Image scale is less well-defined than image size. In medical imaging we generally define the *Field of View* (FOV) and the image matrix size. Together these define the *spatial resolution* of the raw image data. We discuss spatial resolution in detail in Chapter 3. Many file storage formats include a DPI (dots per inch) specification which is a somewhat arbitrary description of the *intended* display or print size of the image. Most software ignores the DPI specification when generating the screen display of an image, but may use it when printing.

2.3.3 Pixel Information

If, as in most cases, an image represents the state of a subject at some time in the past (e.g. A photograph, a CT scan, an MR image), then the image data represents a discrete sampling of some physical property of the subject. An MR image, for example, will have been acquired with a specific field of view and matrix size. A pixel in the raw MR image data represents the average MR signal intensity in a specific volume of space inside the MR scanner (together with a certain small amount of neighboring pixel information according to the point spread function).

The precision with which the signal intensity is measured and recorded, and the amount of noise, determine the maximum possible information content of the image data.

2.3.3.1 Bit Depth

An image must have adequate spatial resolution to show the spatial separation of important separate objects. It must also have adequate *intensity* resolution, or precision, to record any contrast difference between objects – assuming there is a measurable difference in the signals from the objects. In a measured image individual pixels represent discrete samples of the spatially continuous measurement signal. The digital encoding of the measured signal intensity for each pixel also has discrete rather than continuous values – the measured signal is *quantized*. The number of discrete levels, the maximum precision of the stored intensity data, is defined by the number of bits used to store the data – the *bit depth*. The *actual* precision of the data will be limited by the measurement hardware and system noise.

The choice of bit depth used to store raw image data is generally based on the precision of the measurement system. In a properly engineered imaging system we want the precision of the data recording system to be a little bit greater than the precision of the physical measurent apparatus. If the recording precision were too low then expensive measurement hardware would be inadequately utilized and potentially useful information would be lost in the data recording process. Alternatively, if the recording precision were excessively high, no extra information would be saved but data storage space would be wasted, and both data transmission and image processing would be slower.

By way of example, consider the data precision requirements of CT. In CT images each pixel stores a calculated integer CT number which can range from $+3{,}000$ for dense bone to $-1{,}000$ for air. We thus need a bit depth that will encode at least 4001 CT numbers. The bit depth required is 12 ($2^{12} = 4{,}096$). Typical bit depths for other imaging modalities are 10 or 12.

All imaging data is measured and stored with much higher precision than a human can actually see. Human visual perception has quite poor and non-linear discrimination of light intensity. By some estimates humans can reliably distinguish only about 32 different gray scale levels. This is clearly demonstrated in Fig. 2.3 where we see that even if we reduce the number of distinct gray scale levels from 256 to 16 the effect is barely noticeable. Most gray scale image display devices, for example monitors, have a bit depth of eight, with the result that $2^8 = 256$ different intensity levels can be displayed. There are two apparent paradoxes here. Firstly we acquire data with a precision of 2^8 or higher, secondly we display this data with a precision of 2^8, and yet we can only see with precision 2^5. Why do we bother to record and display images with such an apparent excess of intensity precision?

Remember that the raw data we acquire represents the variation in intensity of some measurable physical phenomenon. The information of interest, perhaps some anatomical details, will probably not be represented by intensity variations across

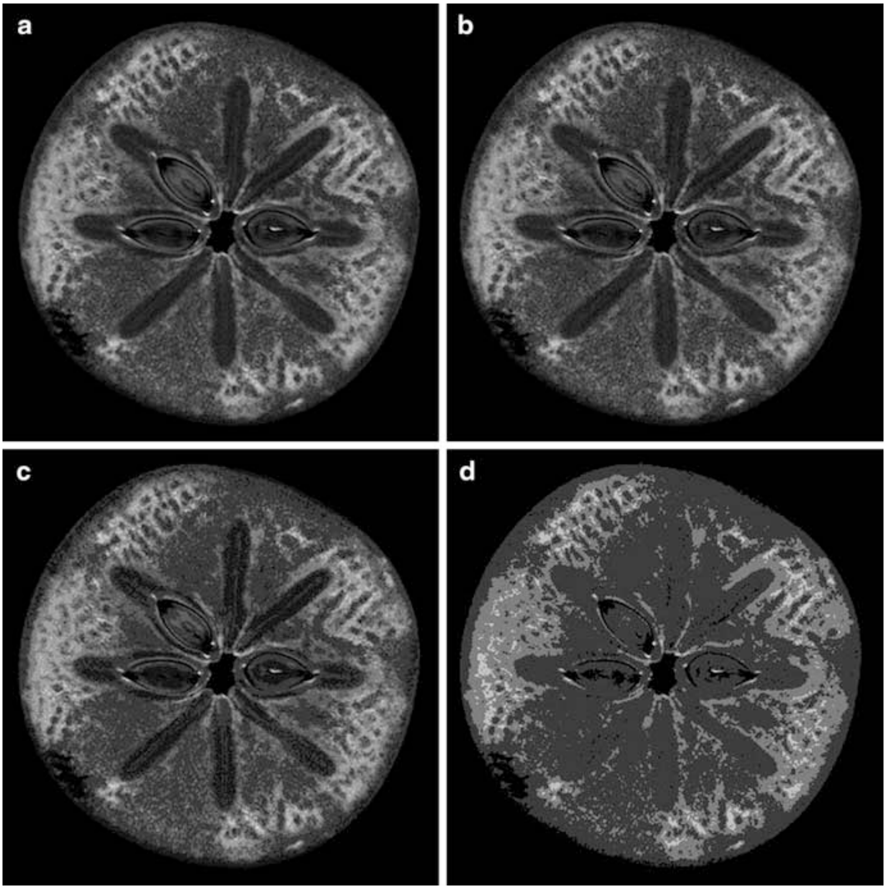


Fig. 2.3 The information content of a digital image depends on the number of pixels and the number of distinct intensities. This figure illustrates the effect of reducing the number of intensity levels on image information content. These MR images of an intact persimmon have 256, 16, 8, and 4 distinct gray scale intensities (**a–d** respectively). In this particular image the reduction in displayed intensity precision from 256 to 16 is barely noticeable. This may not be the case in all images, and in some cases important information could be lost by such a reduction in precision – particularly if we want to see the details in a small region or identify very subtle changes

the full range of measured intensities. It is often impossible or impractical to predict the intensity range of interest prior to acquisition. Thus the imaging technology must be able to measure a range of intensities that can be reliably predicted to include the information of interest, *and* it must record this range with sufficient precision (i.e. intensity resolution) to enable post-acquisition *expansion* of this range to create a display for human vision. That display must have sufficient intensity contrast detail to enable reliable interpretation

Because it is often difficult to predict the intensity range of the information of interest within the raw intensity data one of the most common image processing adjustments is the selective and interactive improvement of contrast performed while viewing an image. The raw information encoded in small differences of intensity may be imperceptible to the human viewer until these differences are exaggerated by contrast enhancement.

In CT data we usually can predict the range of CT numbers that will cover the information of interest for a particular investigation. In this case it is normal practice to use a standard *Window Function* to select a specific range of CT numbers to be displayed as an 8 bit gray scale image as illustrated in Figs. 2.4 and 2.5.

It is important to remember that no amount of post-acquisition contrast enhancement will be able to extract a difference that is not present and significant in the recorded physical phenomenon. As we shall see, there are a number of image processing 'tricks' we can perform to increase the *apparent* differences, and there are even some 'built in' to the human vision system. Whether such information is present and significant depends on the precision and noise level of the image acquisition and recording system.

Although we demonstrated in Fig. 2.3 that reduction of *displayed* intensity precision may be imperceptible it does not follow that we can reduce raw data precision with impunity. When we apply contrast enhancement to improve the visibility of displayed contrast the desired information must be available in the precision of the raw data.

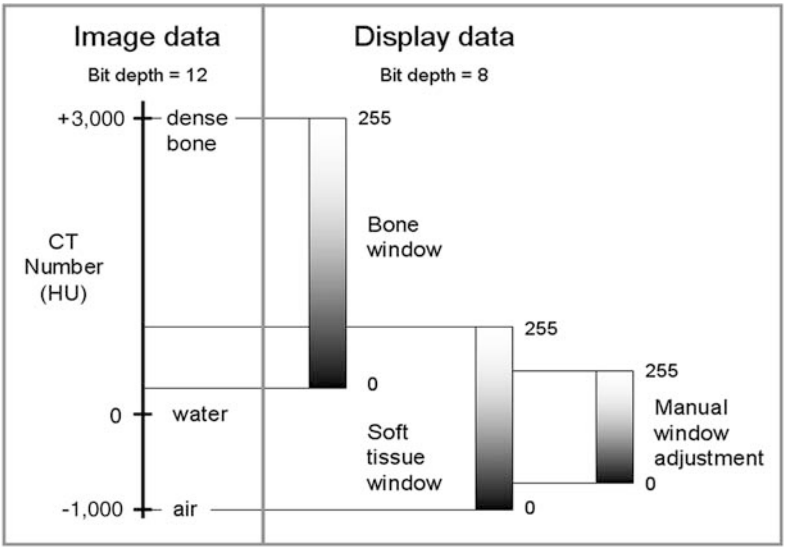


Fig. 2.4 Human perception cannot resolve the full precision of stored CT image data (typically 12 bits). According to the anatomy of interest, standard *Window Functions* are used to select a defined range of data for display. The precision of the display data is 8 bits. A subset of this data may be selected manually by the viewer to further enhance the visibility of specific anatomical features

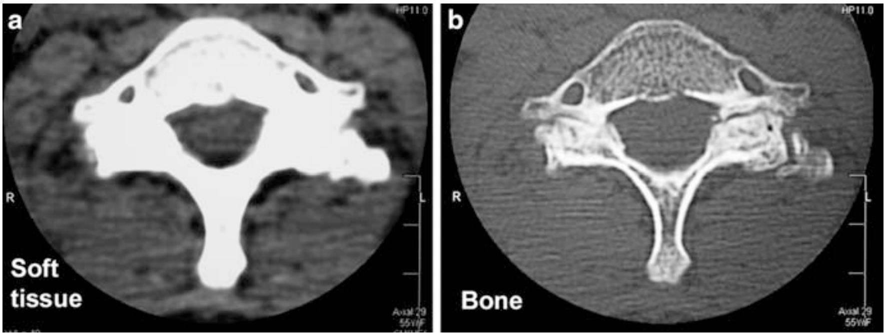


Fig. 2.5 Specific ranges (windows) of CT image data are used to display maximum contrast according the anatomy of interest. Here a single raw data set has been windowed for soft tissue (**a**) and bone (**b**)

2.3.4 Ways of Representing Numbers

So far we have described the binary representation of image data only in terms of positive integers. Using 8 bits we can represent (encode) all the integers from 0 to 255, with 14 bits all the integers from 0 to 16383, and so on. This is fine for image data that is naturally described by positive integers, such as pixel intensities, but often the raw data acquired by an imaging system and the results of image processing are *not* simple positive integers. They may include negative numbers (e.g. voltages), decimal fractions, and may range over many orders of magnitude – more than we can represent using positive integers within the bit depth available. There are several ways of addressing these needs using binary encoding.

2.3.4.1 Signed and Unsigned Integers

In *signed integer* encoding the first bit of the available bits indicates whether the encoded number is positive or negative. You might at first think that this will lead to two equivalent representations of zero (± 0) with the result that only $2^8 - 1 = 255$ numbers could be encoded by 8 bits. However, for 8 bit signed integers, the binary number that you might expect to represent -0 (1000 0000) in fact encodes -128 (this is because negative numbers are encoded differently from positive numbers and '-0' is not represented). In general an n bit signed integer can represent all the integers from -2^{n-1} up to $+2^{n-1} - 1$, a total of 2^n. *Unsigned integers*, the first type of binary encoding we discussed, can represent all the integers from 0 up to $2^n - 1$ with n bits.

2.3.4.2 Floating Point

In floating point encoding numbers are represented by a binary equivalent of the decimal ‘scientific notation’. For example, the decimal scientific notation for the number 123456 would be 1.23456×10^5. In floating point encoding this is changed to 0.123456×10^6. The *significand* (0.123456) and the *exponent* (6) are stored side by side as signed integers. Notice that the significand is actually *not* an integer – the decimal value of the binary number is always interpreted as a number between 1.0 and 0.1. There are many different floating point conventions that assign different bit depths to the significand and the exponent according to the need for precision (bit depth of significand) or dynamic range (bit depth of exponent).

2.3.5 Data Accuracy

What about data *accuracy*? Storing data in a large file with a high bit depth does not mean the recorded measurements are accurate. Nor does it *guarantee* that they are precise. A noisy or unstable imaging system will not be precise, and an uncalibrated system will not be accurate. Precision and accuracy are two distinct properties of measurement. Have a look at Appendix B if you are unsure of the difference.

By data accuracy we mean how well does recorded intensity information, whether relative contrast or an absolute measurement with specific units, reflect the actual physical properties of the imaged object. We would also like the intensity information to be spatially reliable, in other words, it can be attributed to a well-defined region of space.

Spatial accuracy is not strictly a property of individual pixels in the image data. A loss of spatial accuracy means that the image data attributed to a specific region of space in fact contains some contributions from adjacent regions. This could be a result of the Point Spread Function mentioned previously, or movement of the imaged object during the period of measurement. The measurement of the blurring aspect of spatial inaccuracy is discussed in terms of the Modulation Transfer Function (MTF) in Chapter 3.

In CT the calculated and stored CT numbers are directly related to the linear attenuation coefficient (μ) of the imaged tissue. A CT system needs regular calibration using a phantom containing regions of well defined attenuation coefficient to ensure that the calculated CT values are accurate. In other modalities, e.g. MRI and plain X-ray radiography, we are usually measuring relative intensities of signals rather than absolute physical properties. Such systems still require calibration to check the spatial accuracy of the data.

2.4 Image Metadata

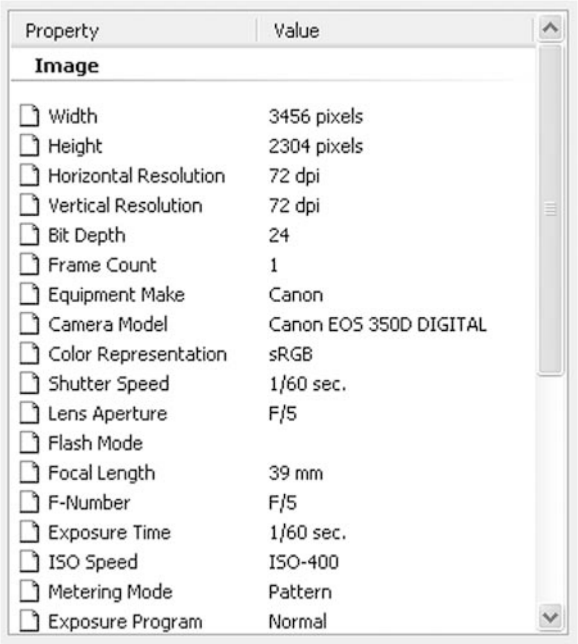
If the only data we stored in a digital image file was a long sequence of bits representing pixel intensities we would be missing a lot of essential information about the image. We would not even be able to display the image if we did not have a record of the pixel dimensions m and n. We would not know if the data represented a gray scale or color image. Other important information such as who or what was imaged, and how and when the imaging was performed would also have to be recorded somewhere and reliably connected with the pixel intensity data. It makes sense to store this sort of information, and a lot more, together with the pixel intensity data in a single image file. With a few rare exceptions this is the basic format of all digital image files. All the non-intensity data is called the *image metadata*, or image file *header*.

2.4.1 Metadata Content

An image file header is not necessarily a sequence of bytes with conventional text encoding. The header itself has a structure that is specific to the file type. How then does a piece of software know what kind of image file it is trying to read? Usually the first two bytes of the header itself are a ‘signature’ that defines the image file type. You can see this by opening an image file with a basic text file editor. Most of the displayed symbols will be meaningless because it is not text code, but the first few characters include text that indicates the file type (only try this with a very small image file or the text editing software may fall over).

The contents and format of the metadata depend on the particular image file type but *always* includes essential information including the size of the image matrix (m and n) and the precision (bit depth). An image file from a digital camera will usually include metadata that describes the camera settings for that particular image (Fig. 2.6). You can inspect some of an image file’s metadata without displaying the image (use File Properties in Microsoft Windows). Any image display software *must* read some of the metadata before it can work out how to display an image.

If the data is compressed then the metadata needs to describe the compression method and the parameters used. A medical image file will include information about the scanner on which the image was acquired, the acquisition parameters, and a way of identifying the patient. For privacy and efficiency, personal and clinical data are usually stored in a file separate from the image file. Figure 2.7 gives a schematic representation of the separate components of a simple digital image file. An example of some typical header information from a medical DICOM format image is shown in Fig. 2.8.



Property	Value
Image	
Width	3456 pixels
Height	2304 pixels
Horizontal Resolution	72 dpi
Vertical Resolution	72 dpi
Bit Depth	24
Frame Count	1
Equipment Make	Canon
Camera Model	Canon EOS 350D DIGITAL
Color Representation	sRGB
Shutter Speed	1/60 sec.
Lens Aperture	F/5
Flash Mode	
Focal Length	39 mm
F-Number	F/5
Exposure Time	1/60 sec.
ISO Speed	ISO-400
Metering Mode	Pattern
Exposure Program	Normal

Fig. 2.6 Image *metadata* is associated (and usually stored with) pixel intensity data. The metadata describes how to display the pixel data, and may include information about the method of data acquisition and the image subject. This table shows metadata retrieved from a digital camera image file by the Microsoft Windows *File Properties* command

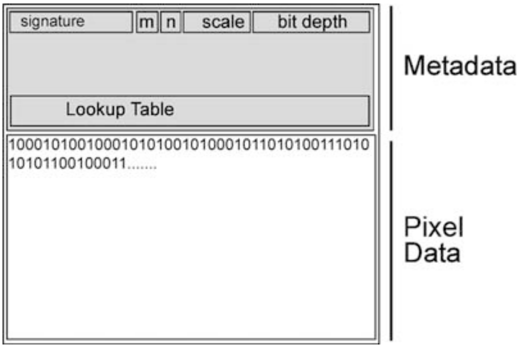
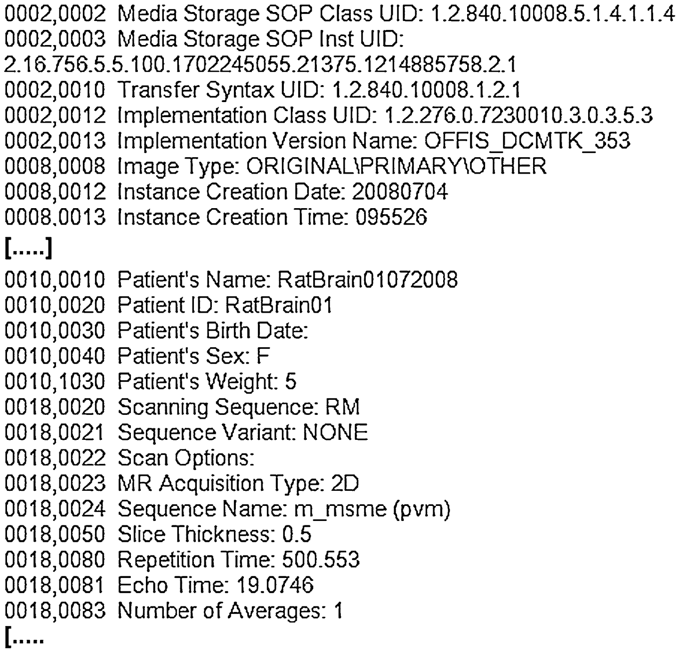


Fig. 2.7 Schematic representation of a digital image file. The metadata describes the image geometry, the source and acquisition parameters, details of compression if any, and may include a lookup table or color map describing the display intensity/color for the stored data. The data section contains the actual, often compressed and encoded, information about the pixel intensities and color



```
0002,0002  Media Storage SOP Class UID: 1.2.840.10008.5.1.4.1.1.4
0002,0003  Media Storage SOP Inst UID:
2.16.756.5.5.100.1702245055.21375.1214885758.2.1
0002,0010  Transfer Syntax UID: 1.2.840.10008.1.2.1
0002,0012  Implementation Class UID: 1.2.276.0.7230010.3.0.3.5.3
0002,0013  Implementation Version Name: OFFIS_DCMTK_353
0008,0008  Image Type: ORIGINAL\PRIMARY\OTHER
0008,0012  Instance Creation Date: 20080704
0008.0013  Instance Creation Time: 095526
[.....]
0010,0010  Patient's Name: RatBrain01072008
0010,0020  Patient ID: RatBrain01
0010,0030  Patient's Birth Date:
0010,0040  Patient's Sex: F
0010,1030  Patient's Weight: 5
0018,0020  Scanning Sequence: RM
0018,0021  Sequence Variant: NONE
0018,0022  Scan Options:
0018,0023  MR Acquisition Type: 2D
0018,0024  Sequence Name: m_msme (pvm)
0018,0050  Slice Thickness: 0.5
0018,0080  Repetition Time: 500.553
0018,0081  Echo Time: 19.0746
0018,0083  Number of Averages: 1
[.....
```

Fig. 2.8 Part of the metadata (header) of a DICOM format image file. This particular file is from a magnetic resonance microimaging system. This part of the header describes the file type, when and where the image was acquired, the acquisition parameters, and some details describing the sample or patient. The eight digit numbers on the left are standard DICOM labels for general and modality-specific image information

> In ImageJ use Menu: Image > Show Info... or simply press 'I' on the keyboard to show some or all of the metadata for an open image file The part of the metadata shown by this command depends on the file type. For DICOM files the full header is displayed.

2.4.2 Lookup Tables

If we measure a signal with 12 bit precision then the most obvious way to store the data would be as a list of pixel intensities, each using 12 bits of storage space. While this is the normal way to store raw image data it is often not the most efficient. Even in a high precision data set it is likely that there are far fewer measured intensities than possible intensities. Consider an 8 bit gray scale image that contains only 31 different measured signal intensities. We may still need 8 bit precision to accurately describe the *relative* differences between the intensities, but we actually only have to *store* 31 different intensities values. A good way to save storage space for such an image is to include a *Lookup Table* (or LUT) in the file header. In our example image the lookup table would list the 31 different intensites with 8 bit precision and each would have an associated 5 bit *index*. Instead of storing all the pixel intensities with the full 8 bit precision we could store a 5 bit index for each image pixel ($2^5 = 32$). This method would require only $\frac{5}{8}$ of the intensity data storage space.

Lookup tables are also referred to as *Color Maps* or *Color Palettes*. Color image files that use lookup tables are called *Indexed Color* images. Image files that do not use a lookup table and store individual pixel data with full precision are called *True Color* images.

Because the lookup table is distinct from the pixel intensity data the way image data is displayed can be easily and conveniently changed by manipulation of the lookup table without having to adjust the individual pixel intensity or color data. A color lookup table can also be used to display gray scale image data as a *'false color'* image (Fig. 2.9).

ImageJ. Use Menu: Image > Lookup Tables to change or invert the lookup table for an image.

Lookup tables are also used to adjust the output of display hardware. A typical computer graphics card (display adapter) includes a built-in lookup table that adjusts the raw display data to suit the specific monitor attached to the card. Monitor calibration systems adjust these lookup tables in order to produce a defined monitor light output (color and brightness) as measured by a photometer placed on the monitor face.

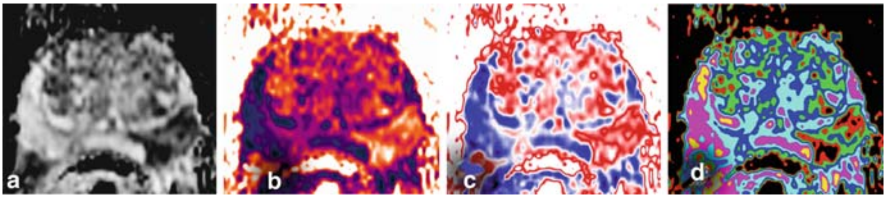


Fig. 2.9 A Lookup Table may be part of the image file metadata and specifies how to *display* the raw image data. In this example a (8 bit gray scale) diffusion weighted MR image of a human prostate (**a**) is displayed using three different color lookup tables. Creation of similar 'false color' images can sometimes increase the visibility of subtle diagnostic features present in medical images (see Fig. 6.8 for a graphical display of the 'Union Jack' lookup table data used for image **c**)

2.5 Image Storage

The storage and transmission of medical images is obviously of critical importance to medicine. Images must be stored safely to protect both the integrity of the data and the privacy of patients, but images also need to be easily available when and where they are needed by medical staff. The imaging software user will normally have the option to store processed images in a number of different standard formats. Although the methods of transmission of images are generally not of concern to image acquisition and processing it is important to be aware that the time required for transmission depends on the size of the image file.

2.5.1 Image File Formats

The choice of image file format has implications for:

1. The size of the stored image file
2. The type and amount of metadata that can be stored
3. The availability of multiple layers and transparent layers
4. The flexibility or 'customizability' of the content
5. The integrity of the data
6. The speed of image transmission
7. Software compatibility

All general-purpose file formats are designed to handle color images. The medicine-specific formats, e.g. DICOM, are primarily designed for gray scale images but are flexible enough to store color images when necessary. The following is a simplified overview of some common image file formats. Most of these formats utilize or enable image data compression.

Image data compression methods are categorized as either *lossless* (no intensity or color information is lost in compression), or *lossy* (some intensity or color information is lost). Section 2.5.2 below discusses compression methods in more detail.

2.5.1.1 Bitmaps and BMP Files

The simplest and most obvious way to store a digital image of size $m \times n$ pixels is as an $m \times n$ array of pixel intensities – commonly referred to as a *bitmap*. You can think of a bitmap as a table in which each entry represents the intensity of a pixel. For a gray scale image with 256 possible intensities we will need $m \times n \times 8$ bits to store the image data.

The term 'bitmap' has both a generic and a specific common usage. The generic term refers to all digital images that are represented as spatial maps of pixel intensities, in other words, as arrays in which each array element corresponds to a discrete position in space. The term *raster graphics* is also used for these images. The specific usage of ‘bitmap’ refers to a particular file format – *Windows Bitmap or BMP*.

2.5.1.2 Vector Graphics

An alternative method of encoding some types of digital images is *vector graphics*. A vector graphics image describes the line and tonal detail as a collection of vectors – lists of points that describe the geometry of objects in an image. Only when the image is displayed or printed is a raster graphics (bitmap) image generated from the vector graphics information – the vector data is *rasterized*. This method is efficient for storage of synthetic images created with graphic design tools as it provides a precise and easily scaleable description of geometrical image features. It is also good for animations as the changing composition of the image (objects, perspective, shadows, etc.) can be calculated geometrically from the virtual objects.

Vector graphics is unsuitable for representation of images with subtle tonal detail, such as anatomical medical images, but would be suitable for the masks and line diagrams used in medical treatment planning. In contrast to rasterization, the reverse process of converting a bitmap or raster graphics image to vector graphics (vectorization) is a relatively very difficult process that is likely to lead to significant loss of image information.

2.5.1.3 JFIF (JPG)

What we commonly call JPG or JPEG images (with file names ending in .JPG) are really JFIF (*JPEG File Interchange Format*) files. JPEG is a compression method, not a file format, and it may be used within file formats other than JFIF, such as TIFF. The JPEG algorithm (outlined below) provides efficient and controllable compression of images but it is most often implemented via a *lossy* method, meaning image information is discarded in the compression process. Any lossy compression process needs to be used with extreme caution on medical images in case important clinical information is lost.

2.5.1.4 GIF

The *Graphic Interchange Format* (GIF) is ideal for storage of simple images containing few distinct colors and very limited tonal detail. Only 256 different colors may be stored and these are encoded in a lookup table. GIF provides for multiple layers, including transparent layers. Transparency permits an image to be displayed on a background such that pixels designated as transparent in the image are displayed with the background color. The background could be a solid color or another image. The layers in a GIF image can be displayed in a timed sequence enabling simple animation. The very limited intensity precision of GIF makes it unsuitable for anatomical medical images.

2.5.1.5 PNG

The *Portable Network Graphics* (PNG) file format was developed as a lossless storage format that would still provide efficient compression. PNG provides for variable precision (8–16 bits) and variable transparency, but does not allow multiple layers. Medical examples of the use of variable transparency would be the overlaying of a color treatment plan on an anatomical image, and the superposition of two images of the same subject acquired from different imaging modalities – CT and PET, say. Because of its high precision and lossless compression PNG could safely be used for storage and transmission of individual medical images. The PNG file will, however, lack the extensive and standardized metadata capability of the DICOM format.

2.5.1.6 TIF

The *Tagged Image File Format* (TIFF or TIF) was designed by developers of color printers, monitors, and scanners. It focuses on the quality of the image rather than the size of the image file, however several different compression methods are supported. A useful feature of the TIF format is that it can store multiple images, or layers, in a single file. Such multiple layers might, for example, represent images of the same object acquired at different times or with different techniques, or an anatomical image and a separate set of annotations. In digital cameras the TIF format is commonly used to store uncompressed image data together with a small JPEG-compressed ‘thumbnail’ image bundled together in a single file (this is the EXIF file structure). The thumbnail image allows a preview of the main image without the need for decompression of the full image data.

The TIF format can be thought of as a package for one image or a collection of images. Depending on the software a range of compression methods, both lossless and lossy, may be available when saving an image in TIF format. A disadvantage of the flexibility of the TIF format means that TIF files created with one type of software may not be readable by some other software. This is the usual reason for the ‘Unsupported Tag’ error message which sometimes appears when unsuccessfully trying to open a TIF file.

2.5.1.7 DICOM

Most medical imaging systems archive and transmit image data in DICOM (*Digital Imaging and Communications in Medicine*) format. The DICOM standard (www.nema.org) is designed to enable efficient exchange of radiological information (images, patient information, scheduling information, treatment planning, etc.) independent of modality and device manufacturer. When we talk about a 'DICOM image' we mean an image file that conforms to Part 10 of the DICOM standard, which currently has 18 parts.

A DICOM image file comprises a header (Fig. 2.8) of image metadata and the raw image data within a single file. The header contains information about the imaging system, the acquisition parameters, and some information about the patient (or the object that was imaged). The DICOM standard provides for lossless and lossy JPEG compression, and other lossless compression formats. Multiple frames, such as the contiguous slice images of a 3D data set, can be stored in a single DICOM file. An important feature of the DICOM format is its ability to store pixel intensity data with precision of 8, 12, 16, or 32 bits according to the measurement precision of the imaging system.

A collection of DICOM files representing all the images acquired from a patient in a single examination usually includes a separate DICOMDIR file that acts as a stand alone 'superheader' describing the individual DICOM image files which have unhelpful file names that make no sense without the DICOMDIR file. Sometimes you can inspect the header information of a single DICOM file to work out what the image represents.

2.5.2 Image Data Compression Methods

Many image file storage formats compress the image data to reduce storage space requirements and speed image transmission. Most image processing software permits the user to specify whether or not to compress the data and what compression method to use. As mentioned above, the choice of method may be based on software compatibility but also on whether any loss of information can be tolerated. Lossy data compression methods discard the information that is considered to be least obvious to human perception and can often achieve an 80–90% reduction in file size. Lossless compression methods reduce the size of the stored data by methods that are perfectly reversible – they eliminate *only* redundant data. Images stored with lossless compression methods are identical in information content to their uncompressed counterparts.

Three different kinds of redundancy are possible in image data:

1. Coding redundancy. This is the type of redundancy described above where the data encoding method has more precision than is necessary for a particular image. This type of redundancy can be addressed by using a reduced bit depth and a lookup table.
2. Spatial redundancy. This occurs when there are large regions of identical pixels each containing identical information – for example the black background of an X-ray image. This redundancy can be reduced by a method that encodes the description of homogeneous regions.

3. Information redundancy. This is information that cannot be perceived – for example spatially small regions with very small differences in intensity and color cannot be seen by humans. This redundancy can be eliminated by making such regions homogeneous. Note that such newly homogenous regions will then be amenable to elimination of spatial and coding redundancy.

A bitmap file format stores an $m \times n$ image as an $m \times n$ rectangular array of pixel intensities. There are two reasons why this format is generally a very space-expensive way to store an image. Firstly, most images have significant areas of indentical, or nearly identical, pixel values. This is spatial redundancy. In a medical image, for example, most of the background is usually black, or contains only noise. The second inefficiency lies in the fact that there are often far fewer different pixel intensities present in the image than can be encoded with the nominal bit depth – there is more precision available than is necessary to encode the actual information in the image. This is coding redundancy.

We can drastically reduce the amount of media space required for image storage (and reduce the time required for image transmission) by reducing the redundancies just mentioned. If the first 100 rows of an $m \times n$ image matrix all represent black background then instead of using $100 \times n \times 8$ bits, all set to zero, to store this information we could simply use a code that says ‘pixels 1 to $100n$ have value zero’. Not only would this encoding save a huge amount of space but it results in no loss of image information. Alternatively, if we are prepared to lose some information considered to be unimportant, then we might decide to adjust very similar pixel values to make them identical and thus reduce the total number of different intensities we need to encode. When the image contains *fewer* discrete intensity values than the nominal bit depth can encode we can save space by encoding the intensities in a lookup table.

Image data compression methods take advantage of the spatial, intensity, and information redundancy just described. Statistical analysis of the image data can lead to further improvements in compression. If we make the assumption that the least common pixel intensities do not represent significant image information then we can omit them from the lookup table by changing them to the closest more common value. Similarly, we might decide that single pixels, or small groups of pixels, that do not fit some measured pattern or trend found in their neighborhood are not important and replace their original values in the stored encoding. The more assumptions of this kind we make the more space we save, but more original image information is lost.

2.5.2.1 JPEG

The JPEG (Joint Photographic Experts Group) compression method is ubiquitous in digital imaging. In fact it is so common that the name of the method is used more commonly than the name of the main file format (JFIF) that uses the JPEG compression method.

The JPEG compression algorithm includes both lossless and lossy steps. The lossy step exploits the limitations of human vision and reduces the precision of that part of the image information which is most weakly perceived by the eye. Specifically, this is small differences in intensities between closely spaced pixels (in technical terms: reduced precision of high spatial frequency components. We will have a lot more to say about spatial frequency in Chapter 4). The method breaks images down into blocks of 8 × 8 pixels and reduces the information content of each block. Because the blocks are processed independently, obvious discontinuities appear at the block edges in highly compressed images (Fig. 2.10). The appearance of the characteristic square pattern should not be confused with *pixelation* which results from simple duplication of pixels in images enlarged by the nearest neighbor method (Chapter 8).

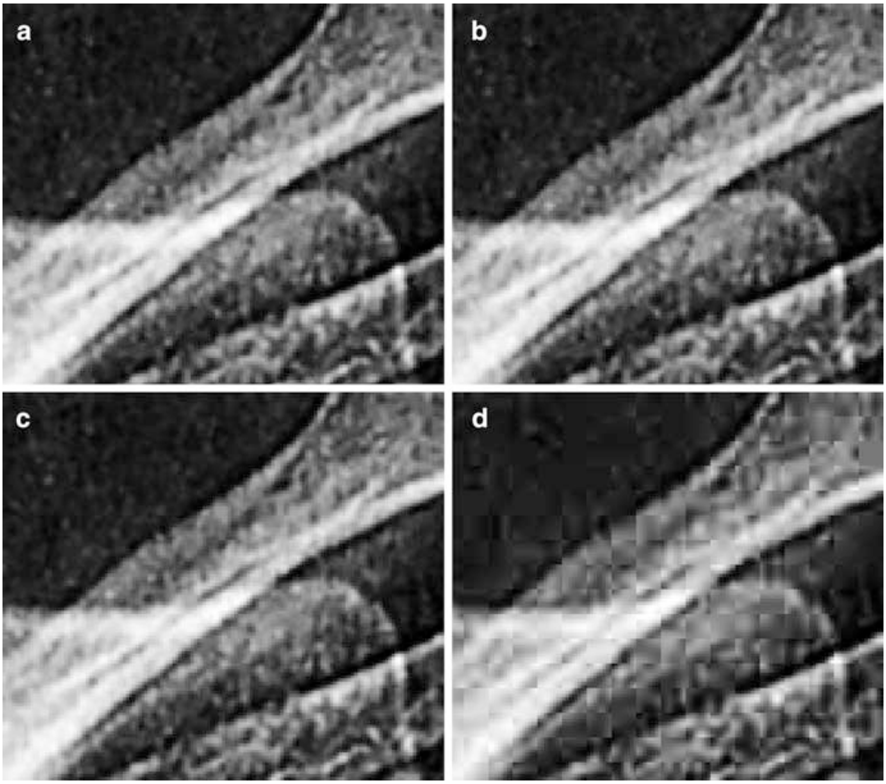


Fig. 2.10 Plain X-ray image illustrating the effect of different levels of JPEG compression. (**a**) Original image. (**b**) JPEG compression level 12 (minimum compression). (**c**) JPEG compression level 6 (medium compression). (**d**) JPEG compression level 0 (maximum compression). At high levels of compression the independently processed 8×8 pixel regions become distinctly visible and edge features are severely degraded. In less severely compressed images a more subtle speckled 'halo' artifact may be visible along edges

In the context of medical imaging we would not usually store raw image data using the lossy JPEG compression method. We might, however, choose to save a *copy* of an image in lossy JFIF format in order to send it as a reasonable-sized file over the internet. In this case both the sender and the recipient need to be conscious of the possibility that the JFIF version of the image may lack some information that is present in the raw data. ‘Lossless’ JPEG compression is available in some software.

A modified method, JPEG2000, has been developed to address some of the deficiencies of the original JPEG method. It is not yet in common usage and cannot be decoded by most image display and processing software, however, it is supported by the DICOM standard.

2.5.2.2 Packbits

Packbits is a lossless compression method that uses *run length encoding* (RLE). RLE takes advantage of the fact that most images have long lines of adjacent pixels of identical intensity or color. Rows of an image matrix are broken up into packets each with its own ‘mini-header’. Whenever three or more adjacent pixels are identical the mini-header is set to indicate that the packet describes how many pixels of a single specific color and intensity follow. If adjacent pixels are not identical the packet header is set to indicate that the packet describes a specific number of non-identical pixels.

2.5.2.3 ZIP, PKZIP

ZIP (derived from PKZIP) is a general purpose lossless data compression method. It can be used within an image file format (e.g. TIFF), as a method to compress a single file, or to convert a series of files and directories (folders) into a single compressed archive file. Most common software supports the ‘unzipping’ of ZIP compressed files but not all image processing software can open image files with internal ZIP compression.

2.5.2.4 LZW

LZW (Lempel-Ziv-Welch) is a lossless compression method used in GIF images and is available in some TIFF implementations. It reduces spatial redundancy by creating a ‘dictionary’ of common intensity or color patterns (a sophisticated kind of lookup table), and then encoding the image data as a sequence of dictionary references.

2.5.2.5 Which Method Is Best?

We have looked at some of the most common file formats and image data compression methods. Figure 2.11 compares the file sizes and image quality for several of these. There are a few important points to notice: (1) In the absence of compression there is little differences in file sizes. (2) Both the type of image data and the compression method have a major effect on the compressibility. (3) The lossy JPEG method does not always provide greater compression than lossless methods. (4) Lossy methods do not *always* produce visible degradation of image quality.

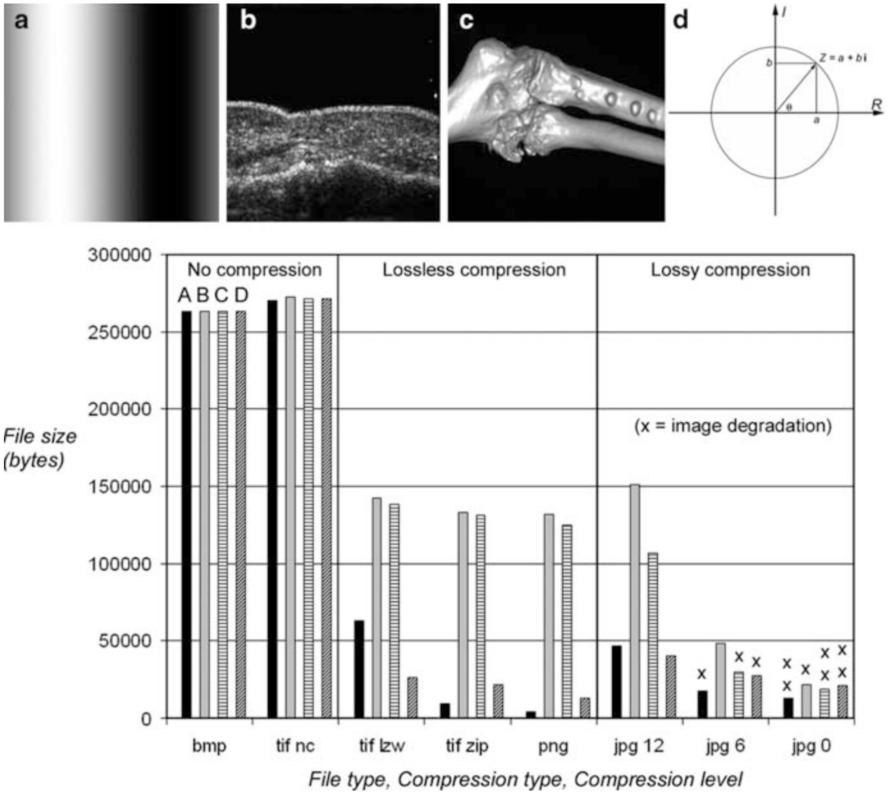


Fig. 2.11 Comparison of file sizes for different image types, file types, and compression methods. The 'compressibility' of an image depends on its content and the compression method. In the case of lossy compression the severity of image degradation depends on both the degree of compression and the image content. Points to note: Lossless PNG compression outperforms even the lossiest JPEG compression for images (**a**) and (**d**). The ultrasound image (**b**) is only slightly degraded even at the highest level of JPEG compression (All original images were 512 × 512 pixels, bit depth 8, gray scale. The compressed images are not shown.)

2.6 Summary

- A digital image is an *encoding* of an image amenable to electronic storage, processing, and compression. A digital image typically represents a discrete and regular rectangular sampling of some property of physical space.
- A *pixel* (picture element) can be either: an element in the raw image data; an element in an image data array created from raw data; or the smallest element in a display device such as a monitor. There may not be a one-to-one correspondence between these pixel types for a particular image at a particular time.
- The *information content* of a digital image is inherently limited by the method of acquisition and storage of the raw image data. Image processing can filter the information content but cannot increase it. The maximum information content that can be stored is limited by the pixel dimensions of the image and the intensity resolution or bit depth. Signal intensity information is *quantized* in digital image data. It can only have discrete values limited by the storage *bit-depth* and number format.
- Pixel intensity data is always accompaied by *metadata* that describes how the intensity data is stored and how it should be displayed. Metadata may also include details of the method of image acquisition and the subject of the image.
- Digital images can be stored in numerous formats, and interconverted between formats, according to the relative importance of data integrity, file size, speed of transmission, and the conventions of particular image processing software. Many digital image files store image data in a compressed format for space and transmission efficiency. Compression methods may be *lossless* (all original image information is retained) or, to a variable degree, *lossy* (some original image information is discarded to enable greater compression).