# Flat binary image format

The simplest image format is a flat binary file. This contains nothing but the image data, so the necessary *metadata* (that is, data describing various attributes of the image such as its dimensions, georeferencing data and so on) are usually supplied as a separate text file. One important consideration for a flat binary file is the order in which the pixel values are stored. A pixel value has up to three coordinates: its row number, column number and (in the case of multi-band imagery) band number. Single-band images are normally stored row by row, i.e., all the pixels values of the first row are stored first, followed by all the pixels of the second row and so on. This is called *row-major* order, and we can represent it symbolically as

row > column

meaning that the row number increases more slowly than the column number as we proceed from one storage location to the next. With an analogues notation, we can represent three conventions for storing multi-band images as follows:

row > column > band

which is called *band sequential* (BSQ) format,

row > band > column

which is called *band-interleaved by line* (BIL) format, and

band > row > column

which is called *band-interleaved by pixel* (BIP) format.

Another important consideration is the number of bytes used to represent each pixel value. A single byte can represent integers from 0 to 255, and this is still very common for remotely sensed images but other possibilities exist, including two-byte integers that can represent integers from 0 to 65,536 (unsigned integers) or from −32,768 to 32,767 (signed integers). Floating-point numbers (i.e., those that can be expressed as some decimal fraction multiplied by 10 raised to the power of some positive or negative integer exponent) are generally stored in four or eight bytes. If the data are represented by more than one byte, it is also necessary to specify the order in which the bytes are stored. The two possibilities are *big-endian*, in which the more significant bytes precede the less significant bytes, and *little-endian*.

By default, Matlab reads and writes in *column-major* order (i.e., column > row) whereas many other systems are row-major, hence the need for the Matlab command `permute` after a call to `fread`, see also the Matlab function `freadenvit` which reads an ENVI or an ENVI-like header file `filename.hdr` which goes with the flat binary format file `filename`:

```
samples = 360
lines   = 180
bands   = 30
data type = 1
interleave = bsq
```

The number of blanks before the equals signs and before " bsq" are important.

W. G. Rees (2013), *Physical Principles of Remote Sensing*, third edition, Cambridge University Press.