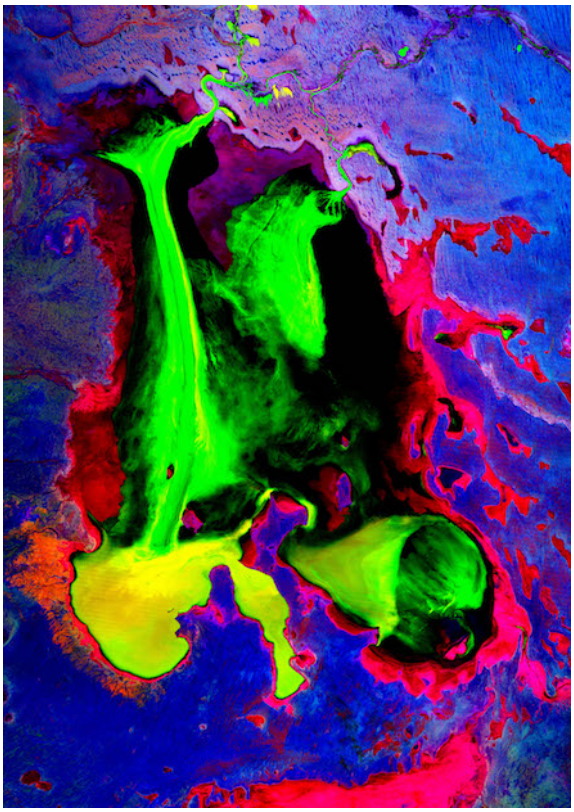
Volume 2C:

Processing—Image Transformations



Recommended Citation for Volume 2C:

CRCSI (2018) Earth Observation: Data, Processing and Applications. Volume 2C: Processing—Image Transformations. (Eds. Harrison, B.A., Jupp, D.L.B., Lewis, M.M., Sparks, T., Mueller, N., Phinn, S., Byrne, G.) CRCSI, Melbourne.

Background image on previous page: Principal Components Analysis of Landsat-5 image of Lake Eyre, South Australia, filling on 9 May, 2009, displayed using PCs 3, 4, 1 as RGB**. Source:** Norman Mueller, Geoscience Australia

Acknowledgements

Production of this series of texts would not have been possible without the financial support of CSIRO, CRCSI, GA and BNHCRC, input from members of the editorial panels and direction from members of the various advisory panels.

Volumes 1 and 2 of this series are based on text originally published in Harrison and Jupp (1989, 1990, 1992 and 1993)[1]. Many illustrations and some text from these publications have been reproduced with permission from CSIRO.

Other contributors are gratefully acknowledged:

- reviewers: ???
- illustrations: Norman Mueller kindly supplied most of the lovely images; other contributors of graphical material include: Tony Sparks, Megan Lewis???
- excursus: ???;
- Daniel Rawson (Accessible Publication & Template Design) for layout and formatting; and
- Carl Davies (CMDphotographics) for selected graphical illustrations.

We thank those owners of copyrighted illustrative material for permission to reproduce their work. Credits for individual illustrations are provided below the relevant graphic.

[1] Harrison, B.A., and Jupp, D.L.B. (1989) *Introduction to Remotely Sensed Data*: *Part ONE of the microBRIAN Resource Manual*. CSIRO, Melbourne. 156pp.

Harrison, B.A., and Jupp, D.L.B. (1990) *Introduction to Image Processing: Part TWO of the microBRIAN Resource Manual* CSIRO, Melbourne. 256pp.

Harrison, B.A., and Jupp, D.L.B. (1992) *Image Rectification and Registration: Part FOUR of the microBRIAN Resource Manual*. MPA, Melbourne.

Harrison, B.A., and Jupp, D.L.B. (1993) *Image Classification and Analysis: Part THREE of the microBRIAN Resource Manual*. MPA, Melbourne.

Table of Contents

List of Figures

List of Tables

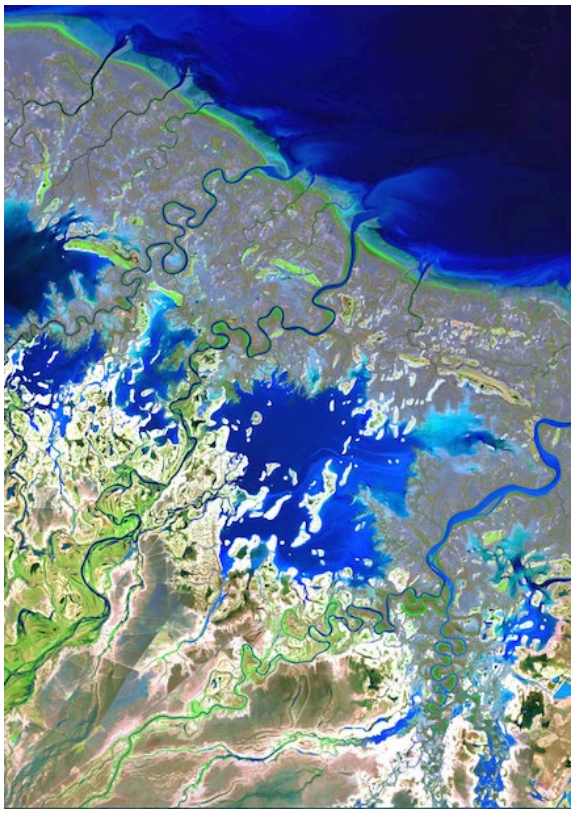
List of Excursus

About this Volume

Image transformations allow image data values to be changed by some consistent and defined process. Such processes are usually described in terms of modifying some or all of the existing pixel values in an image. While modern image processing systems allow transformations to be used without reference to the underlying algorithms, knowledge of their mathematical foundations does ensure that such transformations are applied appropriately. In the context of Earth Observation (EO), appropriate processing is more likely to result in consistent and repeatable results, and enables users to use transformed data with understanding and confidence.

Basic image operations are introduced in Volume 2A of this series and specific analytical tools and methods are covered in Volumes 2B, 2D and 2E. This volume details various image transformations, which may be used for interpretation, analysis or modelling. The modelling stage relates image values to data from other sources and accesses a wide variety of transformations to determine the relationships between different data types, and is further discussed in the context of specific application areas in Volume 3.

MATHEMATICAL FOUNDATIONS



Mention impact of transformations on image histogram – see DLBJ

Transformations may be applied to individual channels or multiple channels based on the input values in one or more channels. Single channel operations tend to involve either:

- rescaling of the spectral value range according to a predefined relationship as introduced in Volume 2A; or
- modifying each pixel value by considering the values of its neighbouring pixels, usually involving the use of filters. Image filtering is introduced in Section 1 below.

Multiple channel operations can be broadly grouped into two categories:

- combining input channels using addition and subtraction (linear operations) as detailed in Section 2; and
- combining channels using multiplication and division (non-linear operations; see Section 3).

Some image processing systems do not differentiate between these two types of multi-channel operation, but simply refer to them by a generic title such as 'band maths'. The following sections attempt to explain the arithmetic processes that are involved in these operations in the hope that they will be used appropriately when processing EO imagery.

Add definitions of four scales of measurement from 2E, p 31: nominal, ordinal, interval and ratio.

Background image on previous page: Landsat-8 OLI image of Burketown, northern Queensland, acquired in on 5 January 2016. **Source:** Craig Shephard, DSITI.

1 Filtering Operations

Image data can be represented in terms of the spatial arrangement of pixels (the spatial domain) or in terms of the frequencies of their spatial patterns (the frequency domain). The frequency domain represents image spatial patterns as sinusoidal waves of differing frequency, amplitude and direction. In EO imagery, large spatial patterns such as regional landscape patches correspond to low frequency waves in the frequency domain while noise or feature edges are represented as high frequencies.

When image data are represented in the frequency domain, specific frequencies can be modified using frequency filtering techniques such as Fourier analysis. Fuller (1966), Castleman (1998) and Moik (1980) discuss methods for designing filters to suppress, enhance or isolate specific ranges of frequencies. Discrete Fourier analysis allows image data to be transformed from the spatial to the frequency domain where they can be filtered in terms of specific frequencies then transformed back to the spatial domain using the inverse Fourier transform function. Fourier analysis is appropriate where data patterns are (semi-)periodic, such as in signal processing applications, but is less effective with data containing local variations, such as EO imagery.

Filtering can also be applied directly to spatial data by 'convolution'—a mathematical operation in which the value of an image pixel is weighted by the values of the surrounding pixels. Richards (2013) and Castleman (1998) discuss the relationship between spatial filter convolution and frequency filtering. Filters defined in the frequency domain can be represented as spatial filters and applied more efficiently (but in some cases less precisely) using convolution in the spatial domain. Conversely, simple and direct spatial filters are difficult to implement in the frequency domain. Spatial filtering is more applicable to EO data that exhibit local rather than periodic variations.

A wide range of filter-based transformations is available in most image processing systems. These can range from generic filtering to special purpose filters for particular data types, such as the computation of insolation or curvature from elevation data. The uses of specific filter-based transformations are detailed in Sections 5 to 7.1 below.

1.1 Spatial Filter Operation

Spatial filter transformations described in this Volume use those image values that are covered by a filter centred on a pixel to change the values of that pixel. The simplest form of a filter is a (square) grid of pixels, which usually has an odd number of pixels along each side to ensure symmetry about the central pixel. In this context, the filter is also commonly referred to as a moving window or box car filter, or a template or a kernel.

In a discrete dataset such as an image, the convolution process at an individual pixel (x,y) is defined as:

$$\sum_{i=-m}^{m}\sum_{j=-n}^{n} f(x+i,y+j)\,g(i,j)$$

where

$2m+1$ is the number of pixels across the filter

$2n+1$ is the number of lines down the filter

f is the image

g is the filter

x is the pixel position in the image

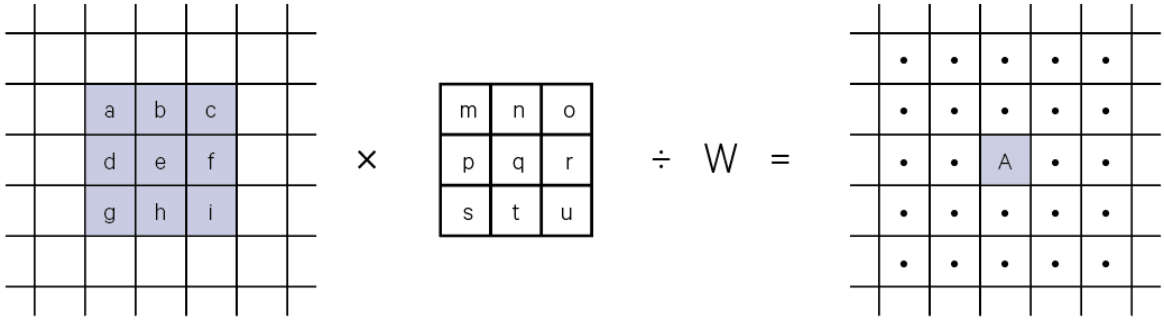
y is the line position in the image.

Weighting values may also be associated with each cell of the filter.

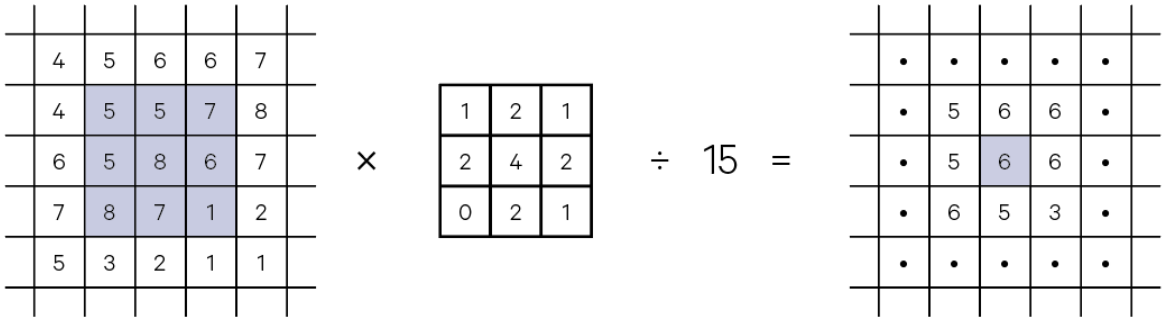
The operation of a 3×3 averaging filter is shown in Figure 1.1a. The filter is applied to each pixel in an image by multiplying it by the filter's central value and adding the products of the other filter elements and their corresponding neighbouring pixel values. Filters can operate by computing a (weighted) mean or average value as illustrated in Figure 1.1, or by determining the median or mode value of the pixels within the filter region. In an average filter, a cell weight of zero effectively excludes the cell from the filter region (see Figure 1.1b).

Figure 1.1 Operation of an average filter

a. This operation is applied to every pixel in an image, that is, every pixel is treated as the central value in the filtering operation to determine its output value. To do this, the filter is systematically passed over the image and moved one pixel after each operation. In this example, the filtered value for the central input pixel (e) will be:
A = (axm + bxn + cxo + dxp + exq + fxr + gxs + hxt +ixu) / W



b. Numeric example. Note that edge pixels have not been filtered in this example.



Source: Harrison and Jupp (1990) Figure 67

The effect of the spatial filtering process on an image largely depends on filter type (and the weighting values used in the case of an average filter) while the extent of this effect depends on filter size. Multiple applications of the same or different average filters can be computed as a larger single filter by convolution of the original filters (see Excursus 1.1).

Pixels at the edge of an image, that is, in the first or last row or column, are typically processed by 'reflecting' the pixel values of the adjacent rows or columns to fill the filter being used. This process is illustrated in Figure 1.3. The edge effects are not significant with smaller filters but should be considered with larger filters especially if image width is relatively small. When significant, they can result is anomalous pixel values around the edges of the image.

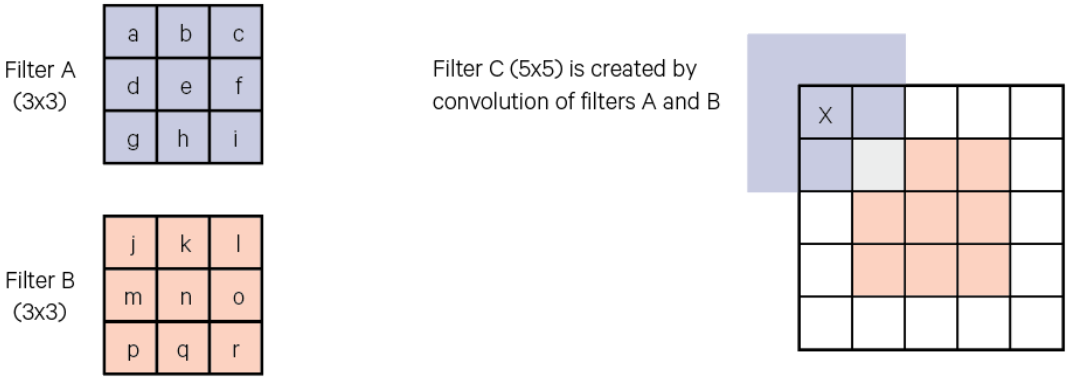
<mention what happens to null values as well>

Excursus 1.1 Filter convolution

Two filters A and B applied sequentially to an image would have the same effect as a single application of the filter C where C is the convolution product of A and B. This convolution process is shown in Figure 1.2. The value of a cell in the combined filter is computed as the product of the weights of the overlapping cells between the two filters when one filter is centred in the new filter region and the other filter is moved around it. This process is useful for determining higher order derivative filters for edge detection or surface shape delineation (see Sections 6 or 7).

Figure 1.2 Convolution of two filters

a. To compute the weights of a 5×5 filter, C = A×B, we sum the products of overlapping cells between A and B, when one filter is centred in the 5×5 region and the other filter is moved around it. The result is the weight of the cell at the centre of the moving filter. In this example, for the filter weight labelled X in Filter C, the overlap between Filters A and B includes only one cell, which is shown as grey.



b. Resulting values for Filter C

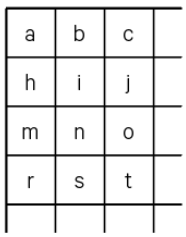
a×r	a×q + b×r	a×p + b×q + c×r	b×p + c×q	c×p
a×o + d×r	a×n + b×o + d×p + e×r	a×m + b×n + c×o + d×p + e×q + f×r	b×m + c×n + e×p + f×q	c×m + f×p
a×l + d×o + g×	a×k + b×l + d×n + e×o + g×q + h×r	a×j + b×k + c×l + d×m + e×n + f×o + g×p + h×q + i×r	b×j + c×k + e×m + f×n + h×p + i×q	c×j + f×m + i×q
d×l + g×o	d×k + e×l g×n + h×o	d×j + e×k + f×l + g×m + h×n + i×o	e×j + f×k + h×n + i×n	f×i + i×m
g×l	g×k + h×l	g×p + h×q + i×r	h×k + i×k	i×j

Source: Harrison and Jupp (1990) Figure 68

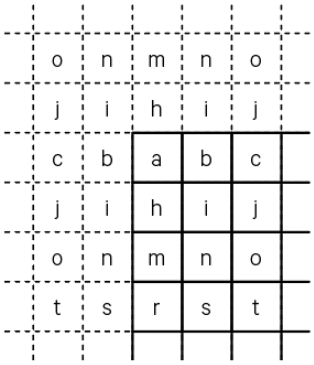
Figure 1.3 Treatment of edge pixels in an image during filtering operation

The values of pixels adjacent to an image edge are mirrored or reflected about the edge to provide a filtering neighbourhood for edge pixels.

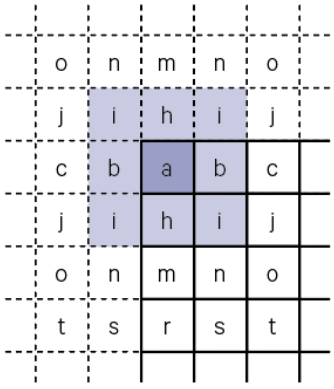
a. Input image values



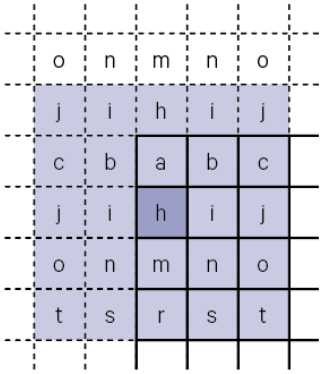
b. Reflected edge values



c. A 3x3 filter centred on pixel with value 'a' uses the image values highlighted below.



d. A 5x5 filter centred on pixel with value 'h' uses the image values highlighted below.



Source: Harrison and Jupp (1990) Figure 70

1.2 Filter Types

Filtering can be based on:

- mean, median or mode values;
- different filter sizes, typically ranging from 3×3 to 11×11;
- threshold values, which define the minimum difference between the input and filtered values of each pixel that are required before its input value is modified;
- variable filter weighting values; and/or
- computations performed in integer or real values.

Larger filters obviously involve more computations and therefore require longer processing time. While these differences may not be significant for small data volumes, they are a consideration with large datasets, even with ongoing improvements in processing speeds. Table 1.1 shows the relative processing times required for different filter sizes.

Table 1.1 Relative processing times for different filter sizes

Filter Size	Time Factor
3×3	1.0
5×5	1.7
7×7	2.7
9×9	4.2
11×11	6.0

Image filters can be defined to reduce spatial variation in the image, such as striping patterns or noise spikes (see Section 5.5), or to enhance selected spatial patterns, such as linear features or edges (see Section 6). The relationship between smoothing, edge-enhancing and edge-detecting filters is illustrated in Figure 1.4.

Figure 1.4 Relationship between different filter operations

a. Original image

<filter operation matrix>

<image>

b. Smoothed image = original image – edges

<filter operation matrix>

<image>

c. Edge-enhance image = original image + edges

<filter operation matrix>

<image>

d. Edge-detection image = original image – smoothed image

<filter operation matrix>

<image>

Source: Harrison and Jupp (1990) Plate 1

When applied to geophysical imagery (for example, magnetics or gravity data) high-pass or edge-enhancement filtering can allow regional trends to be removed, which simplifies analysis of small area variations (see Section 6.2). Alternatively, local variations may be removed to explore deeper or larger geological structures using low-pass or smoothing filters (see Section 5). Such filters are commonly applied to frequency data but may be defined in the spatial domain using an inverse Fourier transform on the desired frequency response and shortening to a practical filter size as detailed in Fuller (1966). Some filters defined in this way are detailed in Appendix 5.3.

Standard filtering techniques derive the filtered value of the central pixel using all (active) pixels in the filter neighbourhood. When the filter lies across the edge of a feature in the image however, smoothing operators tend to blur feature boundaries and de-emphasise small features (see Section 5). Alternative filtering approaches, such as adaptive filtering (see Section 1.4), minimise this effect.

1.3 Thresholding

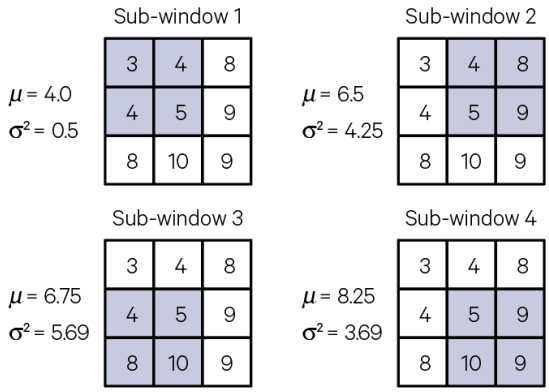
Filtering can also be conducted in conjunction with thresholds such that the centre pixel is only replaced by a filtered value if the difference between the original and filtered values is greater than some specified magnitude. Thresholding is useful for removal of noisy spikes in image data (see Section 5.5), without blurring the low frequency variations or edge features with lower frequency than the noise. Image histogram information may be useful when selecting appropriate levels for filter thresholds (see Volume 2A—Section 8).

1.4 Adaptive Filtering

Adaptive filtering defines sub-windows within the filter region and uses some criterion, such as minimum variance, to select the 'best' sub-window to use to compute for the central pixel. For example, four 2×2 sub-windows could be defined in a 3×3 smoothing filter as illustrated in Figure 1.5. In this case, sub-window 1 has minimum variance so the central value is replaced with the mean of that sub-window. This procedure tends to preserve, rather than blur, edges in smoothing filters and avoids emphasising edges in texture images (see Section 14). Adaptive filtering is further discussed in Section 5.4.

Figure 1.5 Adaptive filtering

A 3×3 filter can be divided into four sub-windows to allow selection of the filter region that is most compatible with the central pixel value. If the criterion of minimum variance were used to select a sub-window, the filtered value of the central pixel would be the mean of the top-left window, with value 4.0



Source: Harrison and Jupp (1990) Figure 69

Another implementation of adaptive filtering is known as the Wallis Filter (ref), which adjusts local image brightness to maintain or match specified local mean and standard deviation. This process results in more uniform tone, with improved local contrast in bright and dark regions (see Figure 1.6), so is particularly useful for highlighting local variations in imagery with uneven illumination.

Figure 1.6 Wallis filter

a. Original image

b. Processing image

Source: Tony Sparks, Icon Water

1.5 Homomorphic Filtering

The application of filtering techniques to non-linearly scaled image data can produce some interesting results. Moik (1980) describes 'homomorphic' filtering in which a linear filtering operation is applied to a logarithmically-scaled image then the result exponentially-transformed back to a linear scale. The additive operation of the filter on the log image has a multiplicative effect on the original image values. The transformation sequence reduces the overall image illumination component and enhances the feature specific reflectance. For example, this type of processing would enhance the subtle variations in image radiance values of shallow water features (see Figure 1.7). Homomorphic filtering is further discussed and exampled in Levine (1985).

Figure 1.7 Homomorphic filtering

a. Original image

b. Logarithm image

c. Filtered image

d. Exponent image (back to original)

1.6 Further Information

Gonzalez and Woods (2018)

Jensen (2016) Section 8

1.7 References

Castleman, K.R. (1998). Digital Image Processing. 2nd edn. Prentice-Hall, Inc. 667 pp.

Fuller, B.D. (1966). Two-dimensional frequency analysis and design of Grid Operators. In Society of Exploration Geophysicists Mining Geophysics. Vol II. Theory. Ed. D.A. Hansen.

Gonzalez, R.C., and Woods, R.E. (2018) *Digital Image Processing*. Pearson Educational Inc., New York.

Harrison, B.A., and Jupp, D.L.B. (1990). *Introduction to Image Processing: Part TWO of the microBRIAN Resource Manual.* CSIRO, Melbourne. 256pp.

Jensen, J.R. (2016). *Introductory Digital Image Processing: A Remote Sensing Perspective.* 4th edn. Pearson Education, Inc. ISBN 978-0-13-405816-0

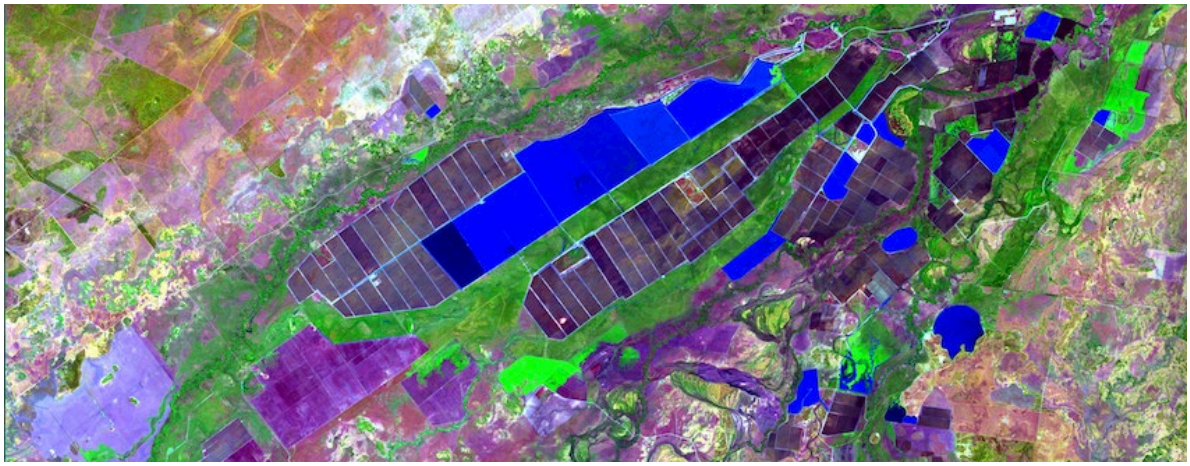
Levine, M.D. (1985). Vision in Man and Machine. McGraw-Hill Inc. USA.

Microimages, Inc (2018). Wallis Filter: http://www.microimages.com/documentation/TechGuides/55Wallis.pdf

Moik, J.G. (1980). Digital Processing of Remotely Sensed Images. NASA SP–431. Washington, DC. USA.

Richards, J.A. (2013). *Remote Sensing Digital Image Analysis: An Introduction.* 5th edn. Springer-Verlag, Berlin. ISBN 978-3-642-30061-5

2 Linear Operations



Background image: Landsat-8 OLI image of Cubbie Station in southern Queensland, which is the largest irrigation property in the southern hemisphere.

Source: Craig Shephard, DSITI.

Linear operations involve addition and subtraction. In the context of image processing, this applies to adding and subtracting pixel values from different channels, often using fractions or multiples of different channel values. In mathematical terms, the simplest way to represent linear combinations of input values is to use matrices (see also Volume 2X—Appendix 6).

Matrix operations allow new image channels to be defined as some linear combination of the original channels. For example, to compute the sum and difference of two channels, *X* and *Y*, we would use the equations:

$$X' = X + Y \quad (\text{or } X' = 1 \times X + 1 \times Y)$$

$$Y' = X - Y \quad (\text{or } Y' = 1 \times X - 1 \times Y)$$

In matrix notation, we assume that each new X' or Y' value can be computed as some weighted sum of the original X and Y values. This allows us to define a table of weighting values. For example, the general transformation equations:

$$X' = a \times X + b \times Y$$
$$Y' = c \times X + d \times Y$$

could be represented by the table:

Output Value	Coefficients for Input Values	
	X	Y
X'	a	b
Y'	c	d

which is more generally represented as the matrix:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

Similarly, the matrix required for the sum and difference equations given above would be:

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

The general matrix operation is usually described as:

$$\begin{bmatrix} X' \\ Y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \times \begin{bmatrix} X \\ Y \end{bmatrix}$$

where X, Y, X', Y', a, b, c, and d are defined above.

Such matrices can be used to transform imagery by computing new channels X' and/or Y' for each image pixel using its values in input channels X and Y. Image data spaces usually have more than two dimensions (channels) as used for the example above. For example, for a four channel image the general matrix operation is defined using a square four-dimensional matrix as shown in Figure 2.1.

Figure 2.1 General matrix operation

$$\begin{bmatrix} \text{output channel 1} \\ \text{output channel 2} \\ \text{output channel 3} \\ \text{output channel 4} \end{bmatrix} = \begin{bmatrix} A & B & C & D \\ E & F & G & H \\ I & J & K & L \\ M & N & O & P \end{bmatrix} \times \begin{bmatrix} \text{input channel 1} \\ \text{input channel 2} \\ \text{input channel 3} \\ \text{input channel 4} \end{bmatrix}$$

or

output channel 1 = A × input channel 1 + B × input channel 2 + C × input channel 3 + D × input channel 4
output channel 2 = E × input channel 1 + F × input channel 2 + G × input channel 3 + H × input channel 4
output channel 3 = I × input channel 1 + J × input channel 2 + K × input channel 3 + L × input channel 4
output channel 4 = M × input channel 1 + N × input channel 2 + O × input channel 3 + P × input channel 4

These linear algebraic operations allow very flexible manipulation of the image data space. In algebraic terms, the operation is simply a linear combination of the image channel values for each pixel. However in geometric terms, a wide range of sophisticated manipulations of the data space may be performed. Guidelines for defining matrix coefficients for general reflection, rescaling, skewing and rotation operators on the image data space are detailed in Volume 2B and Volume 2X—Appendix 6.

For example, a simple geometric operation may be to reflect the data space about the line $Y = X$ as shown in Figure 2.2. In terms of coordinates referenced by the X' and Y' axes, the geometric operation can be summarised algebraically by the two simple equations:

$X' = Y$ (or $X' = 0 \times X + 1 \times Y$)

$Y' = X$ (or $Y' = 1 \times X + 0 \times Y$)

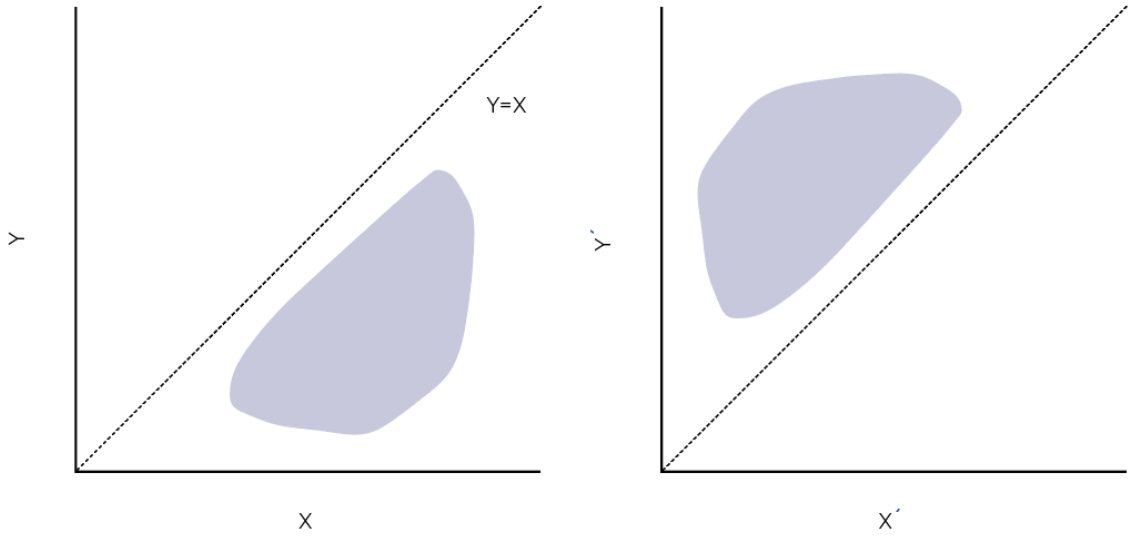
that can be represented by the matrix:

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Figure 2.2 Reflecting image axes about the line Y = X

Simple matrix operations can be used to modify the geometry of an image data value space.

a. Original axes b. Reflected axes



Source: Harrison and Jupp (1990) Figure 85

Most image processing operations allows linear transformations to be defined by multi-dimensional matrices plus an offset, or shift. A linear transformation followed by an offset factor is called an affine transformation (see Section 2.1). Another commonly available linear transformation is Principal Components Analysis (PCA; see Sections 2.2 and 9), possibly with the option to define selected transformation parameters and/or to rotate the transformed channels back to their original data space. A specific channel rotation operation that may be used to enhance vegetation greenness in EO imagery is described in Section 11.3.

2.1 General Affine Transformation

Derivation and application of the linear transformation are introduced above. This operation allows a new channel to be created as a linear combination (or weighted sum) of the original image channels. Affine transformations are commonly used to define vegetation or soil brightness indices in EO datasets.

An affine transformation adds an offset value to the result of a linear transformation to form the matrix equation:

$$\begin{bmatrix}\text{output}\\ \text{image}\end{bmatrix} = \begin{bmatrix}\text{transformation}\\ \text{matrix}\end{bmatrix} \times \begin{bmatrix}\text{input}\\ \text{image}\end{bmatrix} + \begin{bmatrix}\text{offset}\end{bmatrix}$$

In image processing, the offset allows the transformed values to be shifted to the image data range. If a transformation is defined where one channel has a coefficient of one and all other coefficients are zero, the offset can be used to add or subtract a constant value to or from each pixel value in an image channel.

Weighted averaging of multiple channels in an image can be easily implemented using an affine transformation (see also Volume 2B and Volume 2X—Appendix 6). For example, the equation:

$$\text{output channel} = a \times \text{input channel 1} + b \times \text{input channel 2} + c \times \text{input channel 3}$$

can be represented by the matrix equation:

$$\begin{bmatrix}\text{output}\\ \text{channel}\end{bmatrix} = \begin{bmatrix}a & b & c\end{bmatrix} \times \begin{bmatrix}\text{input channel 1}\\ \text{input channel 2}\\ \text{input channel 3}\end{bmatrix}$$

The combined use of affine transformation, logarithmic or exponential transformation (see Section 4.2) and channel ratios (see Sections 3 and 10) can produce any of the indices commonly encountered in EO literature.

2.1.1 Adding and subtracting channels

An affine transformation can be used to perform simple channel arithmetic, such as addition or subtraction. (Addition and differencing of all channels in pairs of images are discussed in Volume 2D—Section 1.1). These arithmetic operations are required to produce indices of particular image features. For example a simple vegetation greenness index could be based on the difference between pixel values in near infrared (NIR) and red channels of an image:

$$\text{Index channel value} = \text{NIR channel value} - \text{red channel value}$$

This index could be computed using the affine transformation:

$$Index = (1 \times NIR) + (-1 \times red)$$

or

$$[index] = [1 \quad -1] \times \begin{bmatrix} NIR \\ red \end{bmatrix}$$

with an offset equal to half the channel data range[2].

The popular normalised difference vegetation index (NDVI) involves ratioing this difference by the sum of the near infrared and red channels:

$$NDVI = \frac{NIR - red}{NIR + red}$$

or

$$NDVI = \frac{(1 \times NIR) + (-1 \times red)}{(1 \times NIR) + (1 \times red)}$$

This can be implemented by computing the sum and difference of the two channels by affine transformation then ratioing the results (see Section 11.2). The affine transformation matrix required to compute this sum and difference is:

$$\begin{bmatrix} difference \\ sum \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \times \begin{bmatrix} NIR \\ red \end{bmatrix}$$

with offsets equal to half the data range for *difference* and 0 for *sum*[3]. The NDVI could then be computed as a ratio of the output difference channel (as the numerator) and the sum channel (as the denominator; see Section 3). NDVI has demonstrated effective and usable correlations with plant greenness metrics, such as biomass or leaf area index (see Volume 3A). It can saturate, however, in vegetation with high greenness, such as irrigated sports fields (Tony Sparks, *pers. comm.*).

[2] In many image processing systems, one image value is reserved to indicate 'null' pixels (see Volume 2A—Section 1.2), which effectively reduces the data range. With byte data, for example, the null value is often value 0 or 255, and the data range reduces to 255 instead of 256 values. In this case the offset would equal 127 rather than 128.

[3] In many image processing systems, value 0 or 255 may be reserved in byte resolution imagery to indicate 'null' pixels, in which case the value 127 would be used as the offset.

Various channel combinations have been suggested to enhance and summarise image information relating to vegetation condition or soil brightness, such as the 'Tasselled Cap' transformations (see Section 11.3). Other specific transformations relating to vegetative cover and condition are considered in Section 11.

2.1.2 Inverting and rescaling channels

Channel data ranges can also be inverted and linearly rescaled using appropriate coefficients. For example, to invert the range of an image channel (byte resolution), we can compute:

$$inverted\ channel = \left(channel\ value \times -1\right) + channel\ data\ range$$

This can be implemented as an affine transformation using -1 as the channel coefficient plus a channel offset. For example, to invert channel a in a three channel image we would use the matrix equation:

$$\begin{bmatrix} inverted\ channel\ 1 \\ channel\ 2 \\ channel\ 3 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} channel\ a \\ channel\ b \\ channel\ c \end{bmatrix} + \begin{bmatrix} offset \\ 0 \\ 0 \end{bmatrix}$$

2.1.3 Linear transformations on log-transformed channels

When an affine transformation is applied to log-transformed imagery (see Section 4.2), channel addition and subtraction respectively represent multiplication and division of the original channels. For example,

$$channel\ 1 \times channel\ 2 = \log channel\ 1 + \log channel\ 2$$

$$\frac{channel\ 2}{channel\ 3} = \log channel\ 2 - \log channel\ 3$$

$$\left(channel 4\right)^3 = 3 \times channel 4$$

(In the case of channel ratios, dark values are implemented in this sequence during the logarithmic transformation; see Sections 4.2 and 10.2.)

A complex non-linear combination such as:

$$\frac{channel\ 1 \times channel\ 2}{channel\ 3 \times channel\ 4^3}$$

would then be a linear concatenation of the previous operations. In terms of matrix coefficients, these transformations (with offsets equal to 0) would be represented as:

$$\begin{bmatrix} \log\left(ch1 \times ch2\right) \\ \log\left(ch2 / ch3\right) \\ \log\left(ch4^3\right) \\ \log\left(\dfrac{ch1 \times ch2}{ch3 \times ch4^3}\right) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 3 \\ 1 & 1 & -1 & 3 \end{bmatrix} \times \begin{bmatrix} \log ch1 \\ \log ch2 \\ \log ch3 \\ \log ch3 \end{bmatrix}$$

The exponentiation option, which should be available in most image processing systems (see Section 4.2), could then be used to convert the resulting channels to a linear scaling.

2.1.4 Implementing regression models

Where some known linear relationship exists between image values and some ground parameter, such as crop yield, and can be expressed as a regression of the form:

$$parameter = a_i \times x_i + b_i$$

where

x_i is the image value in channel i

a_i is the model coefficient for channel i

b_i is the model constant for channel i

then a new channel, which represents a scaled value for the ground parameter, can be computed using the affine transformation:

$$parameter = \begin{bmatrix} a_i & 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} channel\ i \\ channel\ x \\ channel\ y \\ channel\ z \end{bmatrix} + \begin{bmatrix} b_i \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The regression technique described above can be used to determine the relationship between two image channels. One channel can then be rescaled to match the other using the regression coefficients and offsets as exampled above (see also Volume 2D—Section 1.4).

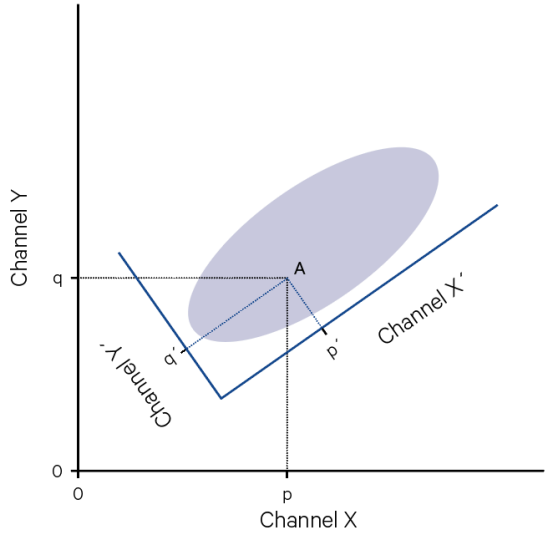
Other uses of the general affine transformation are for Principal Components Analysis (PCA; see Section 2.2) when a predefined matrix is to be applied rather than one specifically relating to the images being processed. The Canonical Variates Analysis (CVA) transformation matrix (see Volume 2E), which maximises the separation between sample image values, may also be implemented as an affine transformation. Most image processing systems allow both PCA and CVA transformation matrices to be computed for selected image values.

2.2 Principal Components Analysis

As discussed in Section 2.1, a multi-channel image can be considered as a multi-dimensional matrix, or multi-vector space. For a two channel image, the two-dimensional data space can be represented in a crossplot (see Volume 2A—Section 8.1.3). It is possible to redefine pixel values in an image by geometrically changing the positions of, or relationship between, the axes of this data space. For example, in Figure 2.3, by rotating and shifting the channel axes, the point A becomes defined as position p' and q'. Data space axes can similarly be rescaled, reflected or rotated as detailed in Volume 2X—Appendix 6. Such changes can allow spectral patterns and relationships in the data to be enhanced and analysed more directly.

Figure 2.3 Redefining image axes

Image data values can be redefined relative to modified position and scaling of axes.



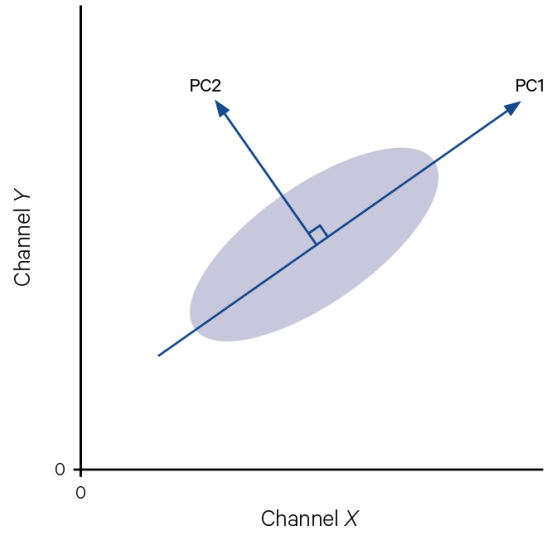
Source: Harrison and Jupp (1990) Figure 87

Mathematical matrices allow changes to the 'spectral' geometry of two and higher-dimensional data to be represented algebraically. For example, in Figure 2.4, channels X and Y show a high degree of correlation in the original data space. As detailed in Volume 2X—Appendix 6, the original axes can be rotated and each pixel assigned a new value relative to the new coordinate system. In that figure, the rotated channel X' is aligned with the direction of maximum variance so that the data channels are uncorrelated in the rotated space. This type of rotation is used in Principal Components Analysis for data reduction and enhancement.

The Principal Components transformation (PCA, also referred to as eigenvalue, Hotelling or discrete Karhunen-Loeve (KL) transforms) uses spectral statistics of the image to define a rotation of the original image such that the data are arranged along axes of decreasing variance. The coordinates for the new axes are computed by an affine transformation (that is, a linear combination) of the original data coordinates. The principle of a PCA transformation is illustrated in Figure 3.6 for a two channel image. The first principal component, PC1, is defined as the axis that passes through the mean data value and is aligned with maximum variance in the image. The second principal component, PC2, is defined as being orthogonal (perpendicular) to PC1.

Figure 2.4 Principle of PCA transformation

A new primary data axis (PC1) is defined in the direction of maximum image variance (over all image channels) then other axes are defined to be orthogonal to this direction.



Source: Harrison and Jupp (1990) Figure 88

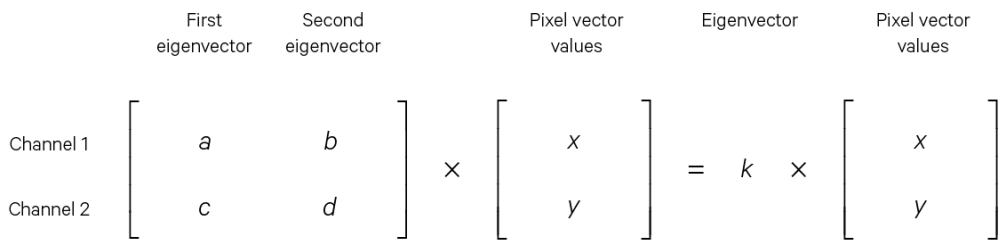
The same concept can be applied to images with more than two channels with the rotation transformation being computed from the covariance matrix. This matrix would typically be derived from the whole image, but may also be based on a training set (see Volume 2A—Section 9.1.2) or spectral theme (see Volume 2A—Section 9.1.3) if the first principal component was required to be aligned with the variation in a particular cover type to enhance a specific feature.

The covariance or correlation matrices indicate the degree to which image channels are correlated (see Volume 2A—Section 8.1.4). The PCA transformation produces an image with totally uncorrelated channels. This is done by mathematically defining a rotation that creates an image with all zero covariance values. In terms of the covariance matrix, this would mean that all off-diagonal elements were equal to zero. Since only the diagonal elements are non-zero, such a matrix is referred to as a diagonal matrix.

As detailed above, a general linear transformation can be represented as a matrix operator. This transformation redefines each image pixel relative to a new coordinate system. Special vectors exist between the two coordinate systems such that points on these vectors are simply redefined by the transformation as a constant multiple of their original coordinates. These vectors are called eigenvectors of the matrix and the constant values are called eigenvalues. Mathematically this is expressed as shown in Figure 2.5. The eigenvectors and eigenvalues of a matrix may be found using matrix algebra as detailed in Volume 2X—Appendix 6.

Figure 2.5 Eigenvectors of a matrix

Special vectors called eigenvectors exist between the original and transformed image coordinates, which can represent the transformation as a constant multiple (called the eigenvalue) of the original coordinates.



Source: Harrison and Jupp (1990) Figure 89

Include image example with correlation matrix, eigenvalues, eigenvectors and PC equations

Interpretation of principal component channels depends on the features of the imagery being transformed. The matrix coefficients specify the contribution each original image channel is making to the transformed channels so can be a useful indication to the 'meaning' of each PC. In Landsat TM imagery the first principal component (PC1) usually represents (at least) 90% of the data variation and is typically computed using all positive coefficients. This channel is a useful summary of image brightness, including topographic shading. As such it may be used for transformations which require a relief shading channel (see Section 7.1) or to split spectral classes derived from log-ratio channels into categories which indicate surface shading. In Landsat TM imagery containing extensive areas of vegetation, the second principal component (PC2) is usually related to vegetation greenness and the third principal component to vegetation 'yellowness' or soil colour (see also Section 11.3). In this case, the coefficients for PC2 would be positive in the infrared channels and negative in the visible channels or vice versa. These definitions vary widely depending on characteristics of the image channels being processed. Simpson (1990) reports that PCA of Landsat TM channels 1–4 clearly displays vegetation density changes (such as fire scars) in PC2. Low order PCs are generally accepted to represent the noise level in the image and typically contains (much) less than 5% of the data variation.

The image statistics used to define the transformation may be derived from one or more images or entered as predefined values. The latter, for example, may apply to a specific image feature (see Volume 2A—Section 9) or an image classification (see Volume 2E). Once defined, the channels of the matrix may be weighted in different ways such as:

- equal—unweighted; this option would be applicable when channel scaling is correct and equal in all channels of the image;
- inversely proportional to channel standard deviation—this converts the covariance matrix to a correlation matrix so each channel is considered to have the same variance; this option may be useful when the image channels are from different sources (in terms of sensor or date) and/or different transformations, and have totally different scaling;
- inversely proportional to channel data range—this can be used for nominal data or when variance is not related to image distribution, for example in an image with a bi-modal histogram; or
- defined by the user—you may already know the relationship between channels and want to specify which one(s) make the greatest contribution to the PCA; this option is also useful to calibrate the data channels to radiance or some appropriate units before processing.

User-defined channel weighting may be required for nominal data where the scaling in the original channels has been specifically adjusted for category definition, and the range and variance statistics are not comparable between channels. In most EO applications, channel weighting is not required since the scaling between channels of a single image is relatively consistent in current image data sources. It cannot be assumed, however, that the scaling will be consistent between different images (from multiple dates or sources).

Singh and Harrison (1985) claim that a significant improvement in the signal-to-noise ratio of PC images can be achieved using the correlation rather than covariance matrix with PCA of EO data. They refer to the results of a transformation based on the correlation matrix as 'standardised PCs'. This standardisation is suggested to be especially useful in multi-temporal analyses to minimise differences due to atmospheric conditions or sun angle in the images (Fung and LeDrew, 1987).

In general terms, unweighted PCA assumes the error component in the image data is uncorrelated and has the same variance in all channels whereas weighted PCA assumes that the error in all channels is uncorrelated and has variance proportional to the total image variance. Local variance statistics (see Volume 2A—Section 8.2.1) can be considered as an estimate of relative error levels between channels so, for each channel, weights for the PCA could be determined as:

$$\frac{1}{\text{local standard deviation}}$$

The scaling of the PC channels can also be modified by a 'balance factor', which could range from a minimum value (often 0), where all PC scales are equal, to a maximum value (often 1), which scales each PC in proportion to variance. This parameter is most relevant to back rotation from the PCA space to the original

image data space as it effectively allows contrast enhancement of the transformed data before conversion to the original data space. This produces similar results to a decorrelation stretch as described in Section 9.2.

A principal component may also be removed in the back transformation (from the PCA axes to the original image space). This is particularly relevant to noisy imagery where the 'noise feature' can be isolated as one component. Back transformation without that component then effectively removes the noise from the original data space. In four channel Landsat MSS imagery, for example, the fourth PC usually contains (much) less than 5% of the total image variance and visually appears to only summarise image noise.

The PCA transformation matrix is generally computed using the specified image statistics (channel means and covariance or correlation matrix) with adjustment for the selected channel weights and balance factor. This matrix is then applied to selected images as a linear transformation. The resulting values are also rescaled to the full image data range using user-defined expected minimum and maximum values. These values are often best determined iteratively.

Specific uses of PCA are discussed in Section 9. Volume 2X—Appendix 6.3 explains the mathematical derivation of the PCA transformation and interpretation of typical output statistics. Other image transformations, which may be implemented as matrix operations, are detailed in Volume 2X—Appendix 6.1.

2.3 Further Information

Gonzalez and Woods (2018)

Jensen (2016) Section 8

Sawyer (1955)

2.4 References

Fung, T., and LeDrew, E. (1987). Application of Principal Components Analysis to change detection. Photogramm. Eng. and Remote Sensing 53, 1649–58.

Gonzalez, R.C., and Woods, R.E. (2018) *Digital Image Processing*. Pearson Educational Inc., New York.

Harrison, B.A., and Jupp, D.L.B. (1990). *Introduction to Image Processing: Part TWO of the microBRIAN Resource Manual.* CSIRO, Melbourne. 256pp.

Jensen, J.R. (2016). *Introductory Digital Image Processing: A Remote Sensing Perspective.* 4th edn. Pearson Education, Inc. ISBN 978-0-13-405816-0

Sawyer, W.W. (1955). *Prelude to Mathematics.* Penguin Books Ltd. London.

Simpson, C.J. (1990). Deep weathering, vegetation and fireburn. Significant obstacles for geoscience remote sensing in Australia. Int. J. Remote Sensing 11 (11), 2019–2034.

Singh, A., and Harrison, A. (1985). Standardised Principal Components. Int. J. Remote Sensing 6, 883–96.

3 Non-linear Operations

Background image: Landsat-8 OLI image of coastal waters just north of Brisbane, Queensland, acquired on 7 October 2014, displayed using bands 4, 3, 2 as RGB.

Source: Craig Shephard, DSITI.

Section 2 discussed some linear operations that are commonly applied to one to more image channels. In this context, addition and subtraction of channels are examples of simple linear operations. This Section is concerned with non-linear operations, namely multiplication and division, applied to pairs of channels in an image.

3.1 Simple Products and Ratios

Most image processing systems can construct multiple ratio channels from pairs of input channels. This ratio is computed as:

$$\frac{x-a}{y-b}$$

where

x and y are the input channels, and

a and b are their respective dark values.

As mentioned above, the ratio value resulting from this calculation is rescaled according to user-defined parameters to fill the full image data range.

The dark value for an image channel represents the minimum 'data' response in that channel and corresponds to the darkest feature in the image. As detailed in Section 3.3, it provides an origin for computing the ratio of two channels such that values below the dark value are assumed to be due to atmospheric effects and/or sensor 'noise' (see Volume 1A—Section 13).

<mention that shadow divided by shadow =1 – another way of looking at ratios>

The simple ratio transformation is computed quite rapidly and may be used to quickly determine ratio results of multiple channel pairs. The smoothed ratio, described in Section 3.3 is recommended for computation of the final ratio channel for feature analysis as the smoothed ratio process does not accentuate localised differences between channels as typically occurs in ratio channels.

Simple ratios are required however for processes which are attempting to reduce spatial 'noise' in an image. This noise may be due to uneven illumination effects, such as topographic shading. Since the latter has a similar effect on all image channels, shading differences are greatly reduced in ratioed channels. The ratio effectively enhances differences between the channels and reduces similarities. Holben and Justice (1981) examined band ratioing as a technique to reduce topographic shading in Landsat MSS imagery. They reported a reduction in topographic effect by an average of 83% with variations in the reduction effect being due to sun angle and surface slope.

Uneven illumination can be a problem with scan-digitised imagery. This may be compensated for in the same way as haze correction by using the scan of a blank target as the denominator or 'reference' channel. Since the range of the reference channel is usually quite small, to preserve the data range of the image channels they should not be rescaled before ratioing and dark values should be given as zero.

Image channels may also be multiplied. For example, a channel that modulates image brightness, such as relief shading, may be multiplied with other channels to produce a fully modulated image. Such operations may also be done linearly in certain cases as detailed in Section 2.1.4.

3.2 Computing Products and Ratios using Log Channels

Ratioing of values in a linear scale is equivalent to the differencing of the corresponding values in a logarithmic scale (ref). Thus ratioing of image channels may be computed as the difference between log-transformed channels (see Section 4.2), with the differencing operation being computed using an affine transformation to produce log-ratio channels (see Section 2.1.3). When channel ratios are computed this way, dark values are specified as negative scaling (offset) values in the logarithmic transformation. Products of image channels can be derived in a similar way by adding logarithmically-transformed channels.

Log-ratio channels provide a useful enhancement for reducing topographic shading effects, since the log transformation reduces variation in the brightest areas of the image and provides greater contrast in the darker features. Ahmad (1987) used log-ratio channels to reduce the effect of topographic shading in Landsat MSS imagery of a mountainous area in northeast Tasmania.

3.3 Smoothed Ratios

As well as computing the simple ratio result of individual pixel values in a pair of channels, it is possible to apply a smoothing process during the ratio calculation. This operation reduces the 'noise-enhancement' effects, which can occur in ratio channels due to small variations between channels. Gordon and Philipson (1986) suggest that ratioing selected texture channels would reduce their characteristic edge effects. In this case, a smoothed ratio would avoid increasing within-feature texture.

Some image processing systems can compute 'smoothed ratio' channels for a single pair of image channels. For example, the 3×3 neighbourhood of each pixel may be used to estimate the ratio as the solution to the linear equation:

$$c_i - d_i = r_{ij} \times \left(c_j - d_j \right)$$

where

c_i and c_j are the channels i and j being ratioed

d_i and d_j are dark values in channels i and j

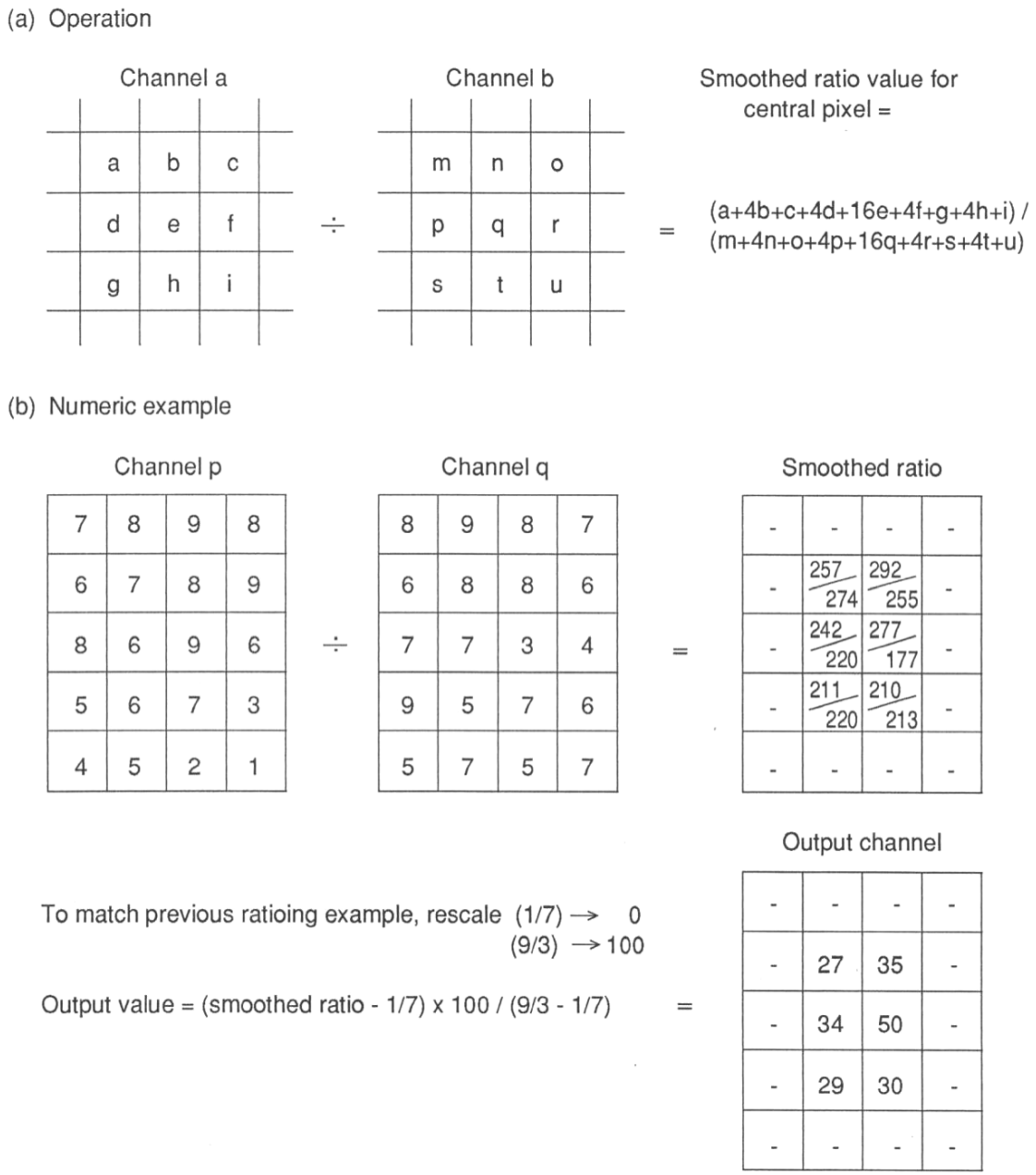
r_{ij} is the estimated ratio value.

This operation is illustrated in Figure 3.1. The processing order, the non-linear nature of ratioing and the rescaling differences between processes, mean that this transformation produces slightly different results to either computing simple ratios of smoothed data channels or the smoothing of simple ratio channels using the same filter weights.

Figure 3.1 Smoothed ratio operation

a. Process

b. Numeric Example. The resulting smoothed ratio values are rescaled to integers in the output range using the same minimum and maximum values as Figure 14.1 for comparison of results. Edge pixels have not been processed in this example.



Source: Harrison and Jupp (1990) Figure 97

Some implementations of this process may also allow themes to be used for the channels being ratioed. This option is a useful way of pre-segmenting the image into feature and non-feature areas and then computing a ratio value only for those pixels that are deemed to represent the feature. Pixels outside the specified theme are set to the default null value in the output image. These may be subsequently 'filled' using filter-based interpolation (see Section 5.6) if appropriate. However, it is usually recommended that the themes are defined to be too broad rather than too narrow to ensure all feature pixels are processed. With the vegetation index ratio of near infrared divided by red for example, a theme from zero to the maximum vegetation radiance value in the red channel may be used for the denominator and the range of values from the minimum vegetation near infrared radiance to the maximum channel value would form a theme for the numerator.

The additional computation required by the smoothing process means that this transformation is likely to be slower than the simple ratios described in Section 3.1. However, the results of this process provide a more 'stable' set of values, which highlight the major rather than minor variations between the channels. Thus, when ratio channels are being computed for density slicing or other interpretation purposes, it is highly recommended that they be produced as smoothed rather than simple ratios.

3.4 Further Information

Gonzalez and Woods (2018)

Jensen (2016) Section 8

3.5 References

Ahmad, W. (1987). A forestry inventory using Landsat MSS data in NE Tasmania Australia. PhD. Thesis. University of Tasmania, Hobart.

Gonzalez, R.C., and Woods, R.E. (2018) *Digital Image Processing*. Pearson Educational Inc., New York.

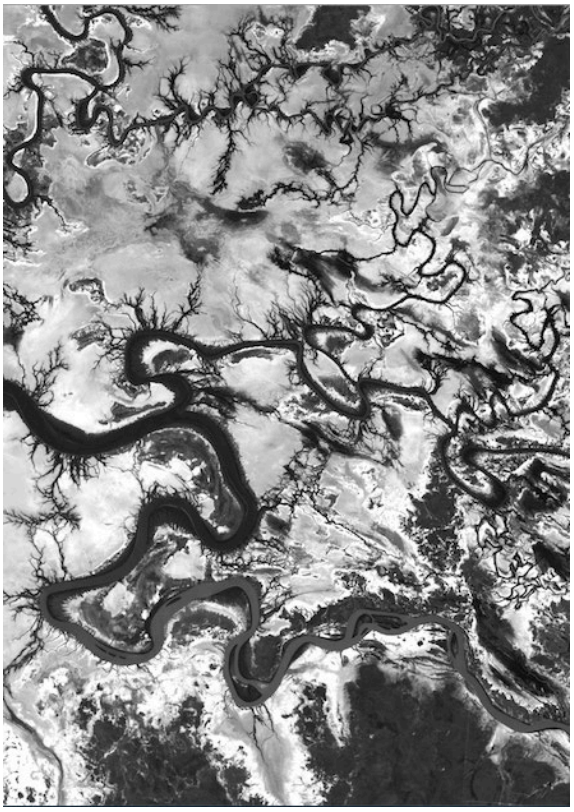
Gordon, D.K., and Philipson, W.R. (1986). A texture-enhancement procedure for separating orchard from forest in Thematic Mapper data. Int. J. Remote Sensing 7, 301–4.

Harrison, B.A., and Jupp, D.L.B. (1990). *Introduction to Image Processing: Part TWO of the microBRIAN Resource Manual.* CSIRO, Melbourne. 256pp.

Holben, B., and Justice, C. (1981). An examination of spectral band ratioing to reduce the topographic effect on remotely sensed data. Int. J. Remote Sensing 2,115–33.

Jensen, J.R. (2016). *Introductory Digital Image Processing: A Remote Sensing Perspective.* 4th edn. Pearson Education, Inc. ISBN 978-0-13-405816-0

SINGLE CHANNEL OPERATIONS



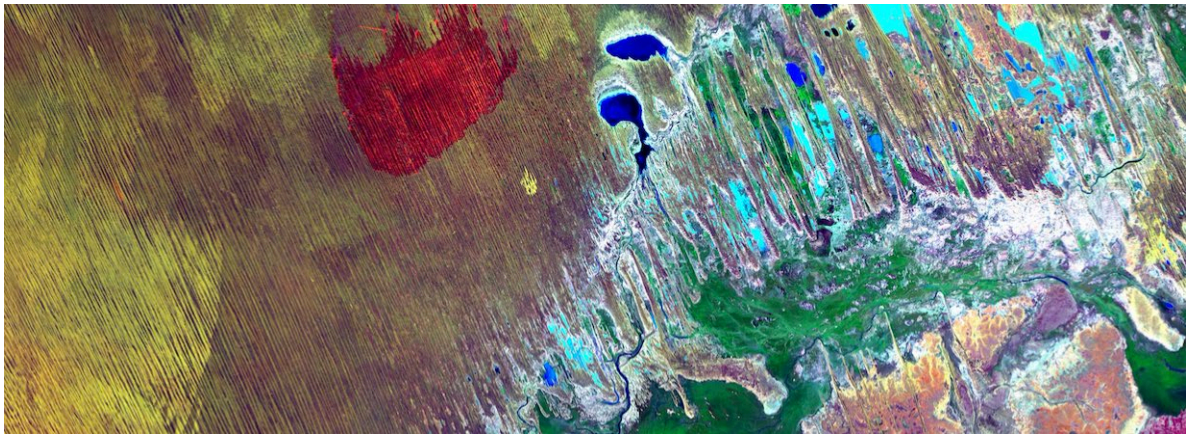
Volume 2A—Section 4 describes some simple methods for rescaling image data values to improve contrast in a displayed image. Rescaled image values can also be written to an image file to:

- match data values between different channels in an image;
- equalise values in imagery from different sources and thus allow them to be subsequently processed using similar parameters; or
- model the relationship between ground measurements and EO radiance.

Background image on previous page: Panchromatic image acquired by Landsat-8 on 14 April 2014 over Norman River in northwest Queensland, which feeds into the Gulf of Carpentaria.

Source: Craig Shephard, DSITI.

4 Rescaling



Background image: Landsat-5 TM image, acquired on 6 October 2011 over the arid landscape of Munga-Thirri National Park in western Queensland. The image is displayed using bands 6, 4, 1 as RGB, contrasting active fires (red) and fire scars (dark red) with temporarily-filled waterways and parallel sand dunes. Note that the red linear streaks, oriented roughly east-west near the large active fire body, are image artefacts resulting from oversaturation.

Source: Craig Shephard, DSITI.

A variety of rescaling operations may be applied to image data for a variety of purposes. These operations are described below in terms of process and application.

4.1 Linear rescaling

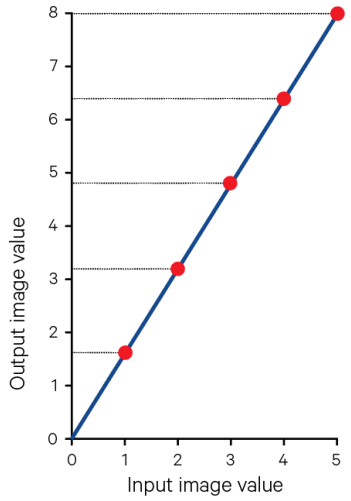
Linear rescaling evenly matches the image data range to an output range, such as the available contrast in an image display device, while non-linear rescaling will selectively increase separation between low or high values in the image (see Volume 2A—Section 4.2). Linear rescaling is also useful for matching the contrast between images from different sources. Balancing of the data values between multi-source imagery, such as multiple images of a given ground location that were acquired on different overpass dates or by different sensors, is required before mosaicking or differencing such image data (see Volume 2D—Section 1.4). Having the same minimum and maximum values in all channels of a single image is convenient for some operations, but such rescaling should be used with caution.

As discussed in Volume 2A—Sections 4 and 8, rescaling can produce discontinuities in the image data values. Given the discrete nature of image values, this discontinuity pattern can be exaggerated by rounding 'errors' during the rescaling process, which lead to a false representation of the differences between values in the output image. For example, in the rescaling illustrated in Figure 4.1, the input values 0, 1 and 2 become output values 0, 2 and 3 so that in the rescaled image the difference between the original values 0 and 1 is exaggerated relative to the original values 1 and 2. These potential problems are worth considering when selecting rescaling ranges, although in most cases the large potential range of data values in EO images means these problems do not seriously degrade image data.

<note from Tony 22/6/16: Same discontinuation occurs when converting to reflectance>

Figure 4.1 Effect of rescaling operation on image data range

Original image values 0–5 are rescaled to the range 0–8. The rescaling falsely represents differences between values in the original image and the values 1, 4 and 7 do not occur in the rescaled image.



Input Value	Output Value
0	0
1	2
2	3
3	5
4	6
5	8

Source: Harrison and Jupp (1990) Figure 64

Rescaling can use linear or non-linear contrast stretching. The rescaling range is generally defined in terms of the minimum and maximum values in the input image, which are to be respectively mapped to the minimum and maximum values in the output image data range. Non-linear stretching can be specified using a gamma value (as used for image display and hardcopy; see Volume 2A—Section 6) with a gamma of 1 producing a linear stretch, a gamma value less than 1 enhancing low image values and a gamma greater than 1 enhancing high image values (see Volume 2A—Sections 4 and 5). Thus, for byte images with a null value of 255, entering minimum values of 0 and maximum values of 254 with a gamma of 1 does not rescale the image channel, that is, it is a null operation.

When the selected rescaling range is within the absolute minimum and maximum values of the image, any differences in values greater than the rescaling maximum or less than the rescaling minimum are effectively lost in the output image and cannot be derived in any subsequent processing. The effect of this situation is also relevant to painted imagery as discussed in Volume 2A—Sections 9.2 and 9.4.

<note from Tony 22/6/16: Issue can be removed if the output data type is changed in the output image???>

Inversion of the data range in an image channel (that is reversing the relative order of pixel values to increase rather than decrease between the channel maximum and minimum values) can be implemented as a linear operation and is described in Section 2.1.2.

4.2 Non-linear Rescaling

Simple non-linear re-scaling of image data values using logarithmic and exponential functions are introduced in Volume 2A—Section 4. These functions are often used to model the relationship between image radiance values and quantifiable ground parameters such as water depth. Transformations based on these functions then allow image values to be converted to or from a scaling which can be linearly related to ground measurements.

Some properties of logarithms make this transformation particularly useful as a preprocessing step for other transformations. For example, the product of two values in a linear data scale may be computed as the sum of their logarithmic values, that is:

$$x \times y = \exp\left(\log(x) + \log(y)\right)$$

Similarly, the quotient of two linear values can be determined from the difference of their logarithmic values, that is:

$$\frac{x}{y} = \exp\left(\log(x) - \log(y)\right)$$

Another calculation that can be simply implemented in a logarithmic scale is exponentiation, since:

$$x^n = \exp\left(n \times \log(x)\right)$$

These relationships allow computation of complicated, non-linear combinations of channels such as:

$$\frac{ch1 \times ch2}{ch3^3 \times ch4}$$

If the logarithms of the four channels are computed, this combination can be produced from the expression:

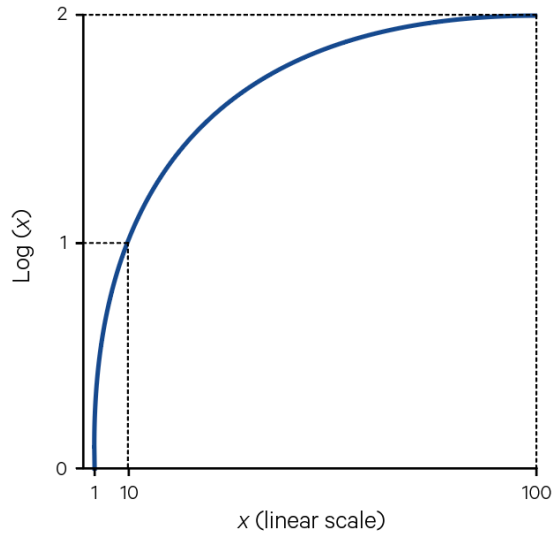
$$\exp\left(\left(\log(ch1) + \log(ch2)\right) - \left(3 \times \log(ch3) + \log(ch4)\right)\right)$$

This computation can be easily implemented in an affine transformation as detailed in Section 2.1.

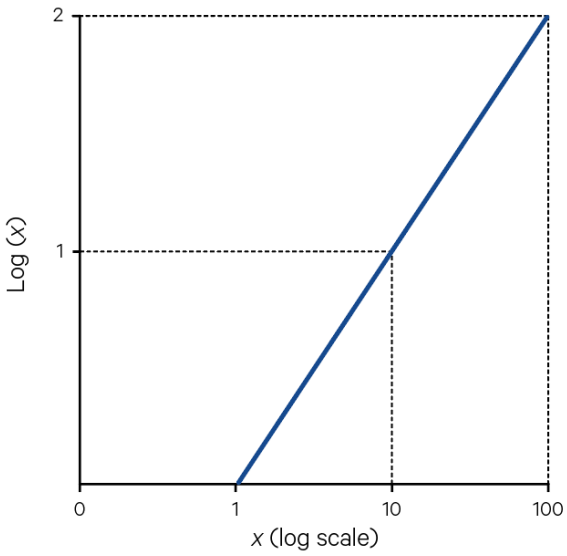
Filtering of log-transformed images effectively multiplies pixel neighbourhood values and can produce some useful enhancement effects. Extensive datasets are often plotted on a logarithmic scale. This scaling is effective when the full data range is large but has significant variation only near one end of the range. Logarithmic scaling does, however, affect the apparent relationship between the variables being plotted. For example, if a logarithmic function (see Figure 4.2a) is plotted using a logarithmic scale for the X-axis, the graph becomes linear as shown in Figure 4.2b.

Figure 4.2 Logarithmic scaling

a. The function $y = \log_{10}(x)$ plotted using a linear scale



b. The function $y = \log_{10}(x)$ plotted here using a logarithmic scale for the X axis



Source: Harrison and Jupp (1990) Figures 65a and 66

The logarithmic transformation has particular relevance to models relating EO image radiance to measurable ground features such as crop yield or water depth. For example, the radiance of clear water in visible wavelengths is modelled as a non-linear combination of substrate brightness and water depth in the relationship:

$$\text{pixel radiance} = \text{deep water radiance} + (\text{substrate brightness} - \text{deep water radiance}) \times \exp(-2 \times \text{attenuation coefficient} \times \text{waterdepth})$$

(Jupp *et al.* 1985). If this model is expressed as a logarithmic relationship we have the parameters of substrate brightness and water depth related linearly as:

$$\log(\text{pixel radiance}) = \log(\text{substrate brightness} - \text{deep water radiance}) - 2 \times \text{attenuation coefficient} \times \text{water depth}$$

If *water depth* and the *deep water radiance* are known and the *attenuation coefficient* can be estimated, this relationship allows us to derive the logarithmic value of *substrate brightness* for an image pixel. An exponential transformation of log (*substrate brightness*) finally gives us *substrate brightness* as an image channel. A description of the implementation of this derivation in terms of image processing steps is given in Jupp (1988c).

In image processing systems, logarithmic and exponential transformations can be implemented by computing the functions:

$$\log(x+a)$$

and

$$\exp\left(\frac{x}{a}\right)$$

where

x is an image pixel value, and

a is the scaling factor.

As indicated in the graph of Figure 4.2a, the logarithm function for value 0 is undefined (said to be negative infinity). The scaling factor '*a*' allows an offset to be added to, or subtracted from, the image values during processing, that is:

$$a=\frac{\beta}{\alpha}$$

where

$$\log(\alpha\times x+\beta)=\log(\alpha)+\log\left(\frac{x+\beta}{\alpha}\right)$$

Since the $\log(\alpha)$ term is a constant offset it can be ignored when rescaling image values. The term would become important for model development if image values were to be related to physical parameter measurements. When log-ratio images are being computed, dark values for the ratio can be specified as negative scaling factors during the logarithmic transformation (see Section 10.3).

Similarly a suitable divisor, *a*, for the exponential function avoids extremely large output values which would be clipped during the rescaling process:

$$\exp(\alpha\times x+\beta)=\exp(\alpha\times x)+\exp(\beta)$$

In this case, the offset can again be ignored in image rescaling and

$$a=\frac{1}{\alpha}$$

In addition to the scaling factor, a data range is specified for each image channel so that all other values are clipped to this range and values within the range are adjusted to increment from a minimum value of zero. This sub-range has relevance when the log channels are to be used for other transformations, such as to form channel ratios (see Section 10.3). Generally the minimum range value should be one less than the absolute minimum of a channel. This value is scaled to the minimum value in the image data range of the output channel. Similarly, the maximum value specified in the processing range is scaled to maximum image data range value in the transformed channel.

The actual transformed minimum and maximum values are often reported during processing and are useful for subsequent rescaling operations and model fitting. The resulting image channels contain data ranges that have been redistributed using logarithmic and/or exponential functions as defined. However, in byte images, the pixel values are usually integer representations of these function values, which are subsequently rescaled to fill the maximum image data range. The final rescaling operation needs to be taken into account when using image pixels values in log (*radiance*) models or when computing the inverse transformation.

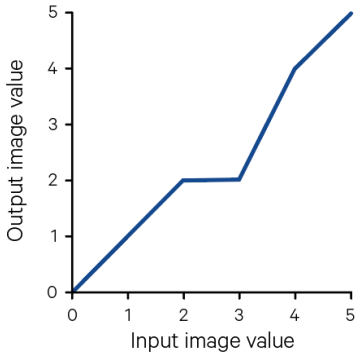
4.3 Lookup Tables (LUT)

Image data values may be redefined using a Lookup Table (LUT). The use of LUTs for image display enhancements is discussed in Volume 2A—Sections 4 and 8. The use of LUTs for density slicing, or for aggregating basic feature classes into labelled categories in the mosaic model for image classification, is also discussed in Volume 2E.

In the context of image rescaling, a LUT performs a similar function by defining an output value for each possible input value in the image. A LUT is simply a single column table; the rows of the table represent input data values while the entries in the table indicate the output values to assign to each input value. The definitions may represent a wide range of contrast enhancements such as piecewise linear or sawtooth ('overlapping contrast enhancement') stretches as illustrated in Figure 4.3.

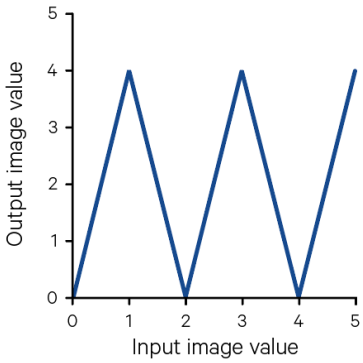
Figure 4.3 LUT operation

a. Piecewise linear rescaling



LUT entry	Output Value
0	0
1	1
2	2
3	2
4	4
5	5

b. Sawtooth stretch



LUT entry	Output Value
0	0
1	4
2	0
3	4
4	0
5	4

Source: Harrison and Jupp (1990) Figure 10

Other more sophisticated rescaling transformations, such as histogram equalisation, may be represented as a LUT (although this is more commonly done as an 'on the fly' statistical transformation). Indeed, for rapid computation, many other image processing functions such as non-linear functions are implemented using LUTs in image processing programs. LUTs are also a convenient means for representing hierarchies and other many-to-one mappings.

4.4 Destriping

EO imagery may contain spatial striping patterns due to the miscalibration between multiple detectors in the scanning system (see Volumes 1 and 2A). In scanners that record adjacent image lines using different detectors, a line striping is frequently observed in which the periodicity of the stripes matches the number of detectors in the instrument. Landsat TM senses 16 lines per scan, but records during both forward and back scans across the swath so its imagery can have 16 or 32 line striping. When such data are geometrically corrected, the striping pattern can be observed to have 17 or 34 line periodicity. Additionally, in linear array or 'pushbroom' scanners that operate banks of detectors across an image line, different pixel values in the line will be determined by different detectors. Imagery produced this way can show vertical striping, such as was visible in some early SPOT data.

<add new figure like 9.7 to illustrate this?>

<note from Tony 22/6/16: need to stress importance of using raw data – now supplied data is mostly rectified so rotates the effect>

Various preprocessing operations are performed by ground receiving stations before the image data are distributed in an attempt to correct for miscalibration effects. However, such variations are data-dependent so are difficult to remove effectively using whole scene statistics or instrument calibration readings. Consequently, striping patterns commonly occur in image data from remote sensing sources and usually need to be reduced in some way before applying other image processing operations for enhancement, interpretation or classification.

Striping patterns are most evident in areas with relatively uniform image values, such as deep water, with different land covers showing different striping effects. In the scanning device, individual detectors are calibrated relative to a standard lamp, which radiates a standard range of wavelengths. For this calibration target, all detectors may record, or can be 'calibrated' to record, the same radiance; however differences in response functions between the detectors can still cause miscalibration for other wavelengths. Thus, the striping pattern varies with the land cover being imaged. In destriping then, these patterns are best identified if the image is segmented into broad land cover regions (such as water, vegetation, bare soil and cloud) before applying destriping methods (see Volume 2A—Section 10 regarding image segmentation methods).

A number of destriping algorithms have been developed to try to minimise the effects of detector imbalance. These algorithms basically accumulate statistics relating to each detector. For example, for an image with six-detector striping, one set of statistics are gathered separately for lines 1, 7, 13, 19, etc., then another for lines 2, 8, 14, 20, etc. and so on. The destriping process then attempts to remove the differences by producing imagery in which selected statistics match between all detectors.

Destriping algorithms typically accumulate mean and variance statistics for a user-defined periodicity in each channel in a (stack of) image(s). Divergence and centrality values are generally reported, which indicate the extent to which each detector is similar to the other detectors in each channel (with low values indicating a close match between detectors). This algorithm is applied to the image(s) by matching all detectors to the mean detector values or to a selected reference detector. The detector reporting the smallest sum of centralities for each channel is recommended as the reference detector. This process should be applied to imagery before resampling as the original striping pattern is likely to be modified to some form of diagonal artefact in the geo-corrected image, which would be more difficult to correct.

<add image from Tony?>

These detector statistics are used to compute a correction equation for each detector and channel combination. This equation can be expressed as:

$$x_{ij} + a_{ij} + b_{ij} \times x_{ij}$$

where

x_{ij} is pixel i in channel j, and

a_{ij} and b_{ij} are adjustment factors computed from the detector statistics.

Some destriping algorithms allow a theme to be defined for the destriping operation—that is, the process is only applied to pixels that satisfy the theme. This option allows the image to be spectrally segmented into different land cover categories without previously separating the image data into different files. As detailed above, it is highly recommended that appropriate theme values be used in conjunction with image destriping to account for the data-dependent patterns.

The effects of destriping may be difficult to discern visually in some imagery but can be quantified by computing the difference between the original and destriped images (see Volume 2D—Section 1.1). In some cases, a narrow data range may make the destriping operation very difficult since the striping effectively occurs about a single grey level value. Linear rescaling of the data channel before destriping may improve the performance of the destriping algorithm.

If a significant striping pattern cannot be removed adequately by using a destriping algorithm, image filtering may be used to reduce the visual impact of the pattern. This process is described in Section 1. In this context, average filtering using the default filter weightings with a filter size that is at least as wide as the detector striping should be selected.

Very noisy imagery should be despiked before destriping (see Section 5.5). For example, isolated noise spikes (pixels with unrealistically high values) are commonly observed in imagery of deep water. The despiking process replaces these values with more reasonable ones and thus avoids biasing any statistics generated for the destriping algorithm.

4.5 Further Information

Jensen (2016) Section 8

Gonzalez and Woods (2018)

4.6 References

Gonzalez, R.C., and Woods, R.E. (2018) *Digital Image Processing*. Pearson Educational Inc., New York.

Harrison, B.A., and Jupp, D.L.B. (1990). *Introduction to Image Processing: Part TWO of the microBRIAN Resource Manual.* CSIRO, Melbourne. 256pp.

Jensen, J.R. (2016). *Introductory Digital Image Processing: A Remote Sensing Perspective.* 4th edn. Pearson Education, Inc. ISBN 978-0-13-405816-0

Jupp, D.L.B., Mayo, K.K., Kuchler, D.A., Van R. Classen, D., Kenchington, R.A., and Guerin, P.R. (1985). Remote sensing for planning and managing the Great Barrier Reef of Australia. Photogrammetria 40, 21–42.

5 Smoothing (Low Pass)

The operation of spatial filters is introduced in Section 1.1. Various forms of smoothing filters are available, which replace the value of the central pixel in a pre-defined filter region with another value determined from the values of pixels within that filter region. Commonly available filters for image smoothing include:

- average or mean, which compute the (weighted) average of all pixels in the filter region (see Section 5.1);
- median, which select the median value of pixels within the filter region (see Section 5.2);
- mode, which select the most common pixel value in the filter region (see Section 5.3).

The effects of these three types of filtering operation are illustrated for different filter sizes in Figure 5.1.

Figure 5.1 Effect of different filter sizes and operation

The original image is shown in Figure 2.4a

		Filter operation	
Filter size	**Full average**	**Median**	**Mode**
3x3	<image>	<image>	<image>
7x7	<image>	<image>	<image>
11x11	<image>	<image>	<image>

Other forms of smoothing filters include:

- edge-preserving filters, based on adaptive windows (see Sections 1.4 and 5.4); and
- despiking, which replaces noise 'spikes' (pixels with markedly different values to their neighbours) with values derived from neighbouring pixels (see Section 5.5).

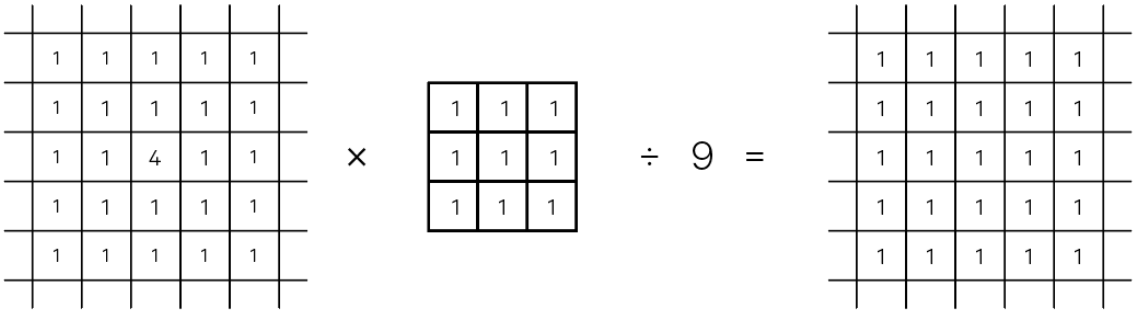
5.1 Average Filters

Image smoothing is best applied to EO data using a mean or average filter. In most image processing systems, the calculation of this filter may be optionally performed in integer or real arithmetic, the latter being slower but more precise. Similarly, average filters can often be defined as integer or real weights with optionally a divisor and offset being applied to the filter product. The use of divisors simplifies the specification of the filter and enables scaling into the image data range. The filtering operation can generally be applied to a selection of, or all, channels in an image.

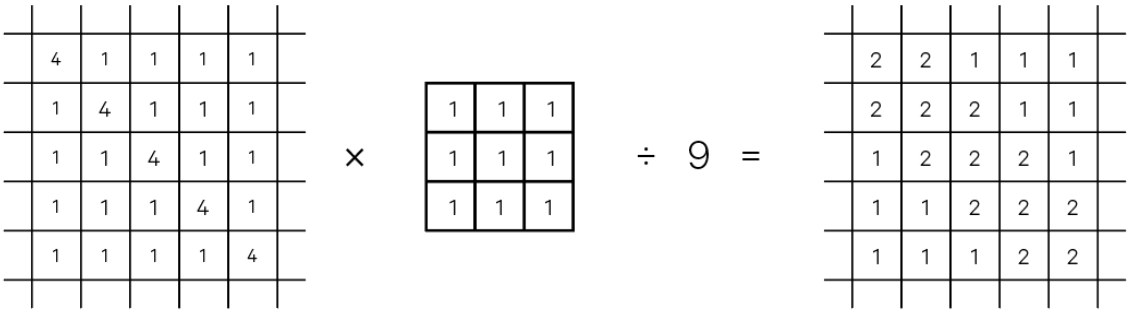
The operation of a simple average filter is introduced in Section 1.1 (see Figure 2.1). When all filter weights are defined using positive values, this process has a blurring or smoothing effect on image features with abrupt changes in value in the input image being softened into more gradual transitions in the filtered image (see Figure 5.2). Such filters are also referred to as low pass or regional variation filters as they effectively enhance the low frequency or regional information in an image by reducing higher frequencies.

Figure 5.2 Effect of smoothing filter on image data

a. Image noise is reduced with a smoothing filter.



b. Linear features are broadened and reduced in intensity by smoothing. Note: in this example, the filtered values for edge pixels have been computed assuming that the pattern of the sub-image is continuous.



Source: Harrison and Jupp (1990) Figure 71

Both the weighting values used in an average filter and the filter size combine to determine the type and extent of effect the filtering process will produce on an image. Smoothing average filters usually contain all positive weights with the central value being greater than or equal to other weights. The effect of a filter can be reduced by increasing its central value relative to the other values in the filter region. The filter size obviously determines the number of pixels that are involved in calculating the transformed value for each pixel, with larger filters producing a greater smoothing effect.

Most implementations of image filtering allow both filter size and weighting values to be varied (see Figures 8.1 and 8.3). The size and weights of the selected filter will depend on the application of the operation. In many image processing systems the default filter is a 3×3 average filter with the weights similar to those below:

1/36	4/36	1/36
4/36	16/36	4/36
1/36	4/36	1/36

or

1	4	1
4	16	4
1	4	1

with a divisor 16. This filter is formed from the outer product of the two one-dimensional smoothing filters:

1
4
1

and

1	4	1

This is a relatively 'gentle' filter, that is, the value of the central pixel is largely retained, with the surrounding pixels contributing to the filtered value in proportion to their distance from the filter centre (the centres of pixels in adjacent rows and columns being closer to the centre of the filter than the centres of the four diagonal pixels). The default filter weights for larger filter sizes are derived by convolution of two smaller filters as detailed in Appendix 5. The impact of filter weights on image values is illustrated in Figure 5.3 using the default filter defined above.

Figure 5.3 Effect of different filter sizes and weights

The original image is shown in Figure 2.4a

	Filter operation	
Filter size	**Full average**	**Weighted average**
3x3	<image>	<image>
7x7	<image>	<image>
11x11	<image>	<image>

Filtering operations such as smoothing generally do not rescale the data range of an image channel. Non-rescaling is ensured in an average filter if the sum of filter weights divided by the divisor equals 1. Figure 5.4 shows two simple full average filters with (b) being a shaped window version of (a). Such filters would have a strong blurring effect on the image data values.

Figure 5.4 Full average filters with differing weights

a. A square filter with all cells active produces heavily smoothed image values as illustrated in Figures 8.2 and 12.7.

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

or

1	1	1
1	1	1
1	1	1

with divisor = 9

b. A filter with corner cells inactive does not consider the diagonally adjacent pixels when computing the filtered value. Smoothing filter weights are typically graduated to have higher values for those pixels that are closest to the central value. This graduation results in a 'softer' smoothing effect than can be achieved using a filter with equal weights in all cells.

0	1/5	0
1/5	1/5	1/5
0	1/5	0

or

0	1	0
1	1	1
0	1	0

with divisor = 5

Source: Harrison and Jupp (1990) Figure 72

In some implementation of average filtering, the effect of the overall filtering operation may also be varied by specifying a weighting value (between zero and one), which is used to compute a weighted average of the filtered result and the original value for each pixel. The use of this value is defined as:

$$\text{transformed value} = \lambda \times \text{original value} + (1-\lambda) \times \text{filtered value}$$

so that a λ value of zero results in full smoothing and a value of one retains the original pixel value.

In some image processing systems, threshold values may also be specified for each image channel to limit the smoothing operation to pixels where the percentage difference between the smoothed and original values (that is, $difference / original\ value \times 100$) is greater than the threshold percentage value (see Section 1.3). Specification of the threshold as a percentage, rather than as a digital value, results in a greater filtering effect on low image values. This is usually desirable for EO data given its typically skewed distribution. Image histogram information may be useful when selecting appropriate levels for filter thresholds (see Volume 2A—Section 8.1.1).

The average filter operation can be used to reduce image noise but also blurs edge features. Within image patches such as vegetated paddocks, this process has an equalising effect so that value variations are reduced. However, definition of linear features such as edges or roads is also reduced by average filtering, although this can be advantageous for image destriping as discussed in Section 4.4. Noisy imagery often benefits from a smoothing operation before edge detection or enhancement filters are used since these operations will enhance both noise and edges.

An unbalanced filter design may be selected in an average filter to destripe an image. For example, to destripe imagery with 6 line (horizontal) striping, we could use a 7×7 filter (that is, greater than the six line striping pattern) with all values except those in the central column of the filter being zero so that only horizontal patterns in the image are modified:

0	0	0	1/106	0	0	0
0	0	0	4/106	0	0	0
0	0	0	16/106	0	0	0
0	0	0	64/106	0	0	0
0	0	0	16/106	0	0	0
0	0	0	4/106	0	0	0
0	0	0	1/106	0	0	0

Such weights should minimise the blurring effect of an average filter.

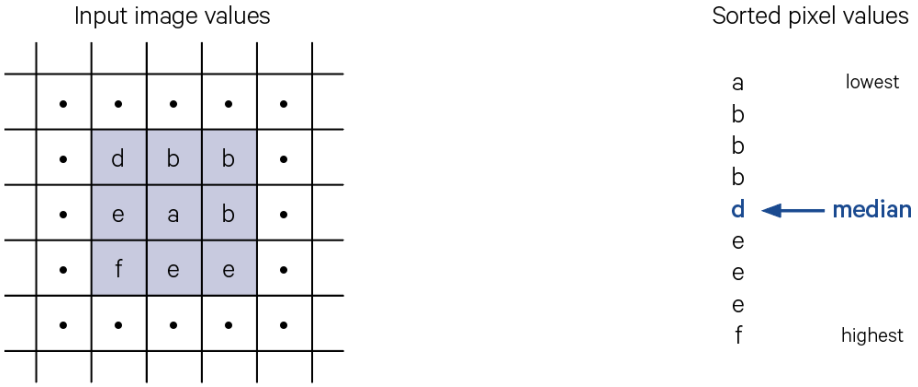
Image average smoothing effectively represents a process of removing edges from features within an image. The impact of smoothing can be seen by differencing the original and filtered images as shown in Figure 2.4 (see Volume 2D—Section 1.1 regarding image differencing). As such, this transformation performs a complementary process to edge enhancement (see Section 6.2).

5.2 Median Filters

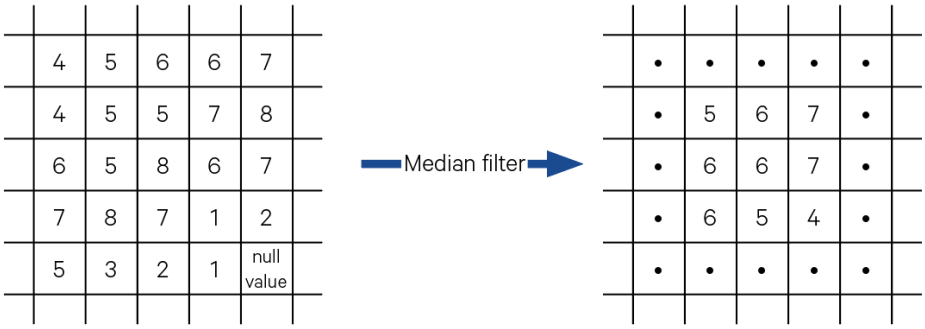
Image smoothing can also be effected by selecting the median of the image values within the filter region as the replacement value of the centre pixel. The median of a set of values is the middle value by count when the values are ordered (see Figure 5.5). Smoothing on this basis allows single pixels with anomalous values to be totally disregarded. By definition the actual value of the median for the filter region already occurs in the image so can be considered a 'reasonable' replacement value for the central pixel.

Figure 5.5 Operation of a median filter

a. The filtered value for the central pixel with value a in the input image is the median (or middle ranking value) of the values in the filter region—in this case, value d.



b. Numeric example. If an odd number of null values exist in the filter region, then two median values can occur. In this case, the two medians are averaged to generate the output value.



Source: Harrison and Jupp (1990) Figure 73

The difference between average and median filtering is demonstrated for a one-dimensional function (such as a line of image values in one channel) in Figure 5.6. The mean or average, a_m, of a set of points, x_i, would minimise the expression:

$$\sum (x_i - a_m)^2$$

whereas the median, a_n, minimises the expression:

$$\sum |x_i - a_n|$$

For the dataset

1.0	0.5	1.0	1.5	9.0	1.0	1.0

the mean value is 15 / 6 = 2.5 while the median values is 1.0. The moving average values (that is the average of each triplet of values) would be:

•	0.83	1.0	3.8	3.8	3.7	•

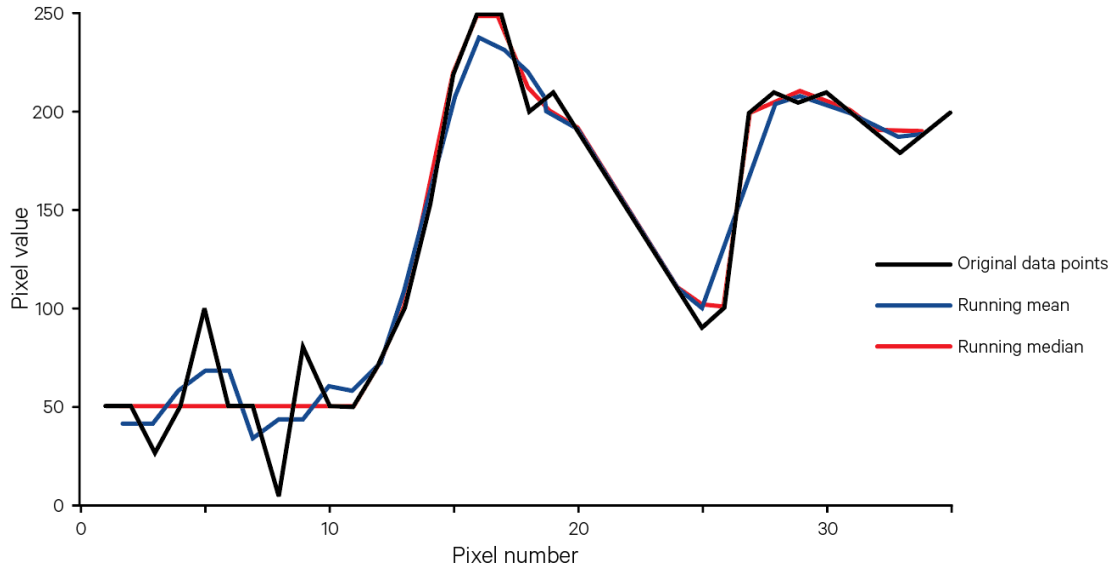
and the moving medians are:

•	1.0	1.0	1.5	1.5	1.0	•

This example and Figure 5.6 demonstrate the way a median filter removes outlier points whereas averaging merely spreads their effect into adjacent data points. Sudden changes in data values however, such as feature boundaries, are also better preserved by median filtering as illustrated in Figure 5.1.

Figure 5.6 Average versus median smoothing

The effect of mean and median filtering is shown for a one-dimensional function. The median filter removes outlier values and preserves edges more effectively than the mean filter.



Source: Harrison and Jupp (1990) Figure 74

Repeated median filter operations tend to quickly reach a stable end point (Tukey, 1977) whereas average smoothing can cycle through a series of minor variations in pixel values. This form of smoothing provides a powerful tool for noise reduction in image processing with less loss of edge information. This filtering mode is commonly used in conjunction with thresholds for despiking an image as described in Section 5.5.

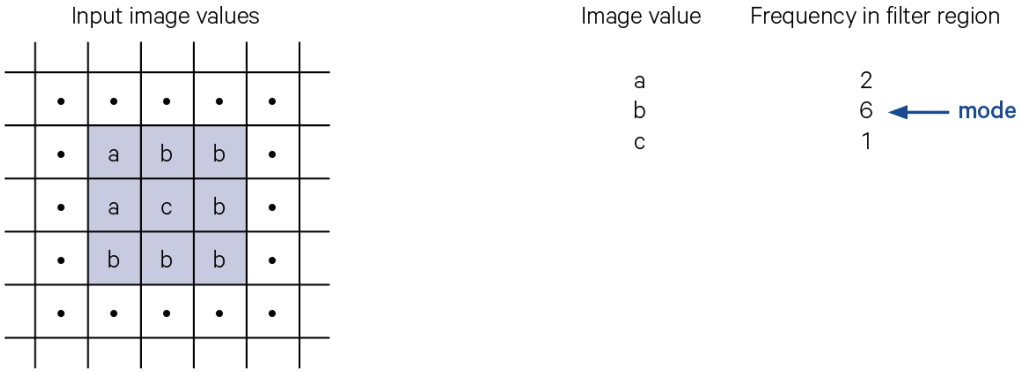
Median filtering can be applied in most image processing systems. The effect of filter size with a median filtering is illustrated in Figure 5.2. If an even number of non-null pixels occurs in a filter region (that is, some pixels may be equal to the null value so are excluded from the filtering calculation) the average of the two mid-range values is generally used as the output median value.

5.3 Modal Filters

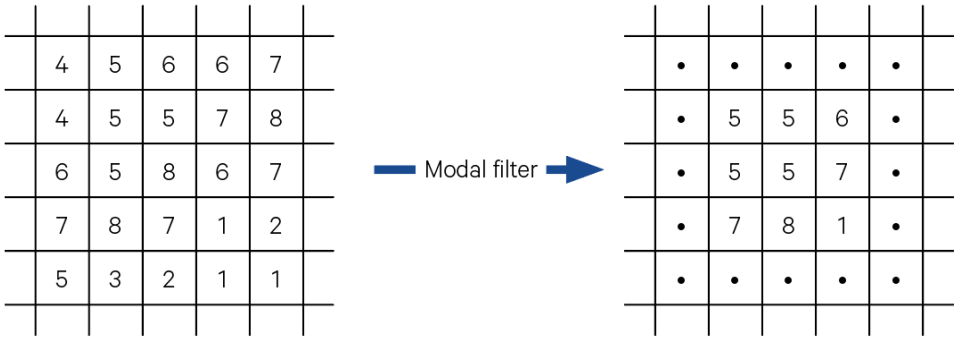
Imagery can also be smoothed by selecting the mode, that is, the most ‘popular’ value, within the filter region as shown in Figure 5.7. This approach is particularly useful for nominal data where the ordering of values does not imply a relationship between them since their average or median would be a meaningless result. The effect of variations in filter size in the modal filtering operation is illustrated in Figure 5.1.

Figure 5.7 Operation of a modal filter

a. The filtered value for the central pixel (with value c in the input image) becomes the most commonly occurring value, or mode, in the filter region—in this case, value b.



b. Numeric example. Where two or more values are equally 'popular' in the filter region, a predefined rule determines which value is used as the model. In this example, the lowest value is selected for the output pixel value.



Source: Harrison and Jupp (1990) Figure 75

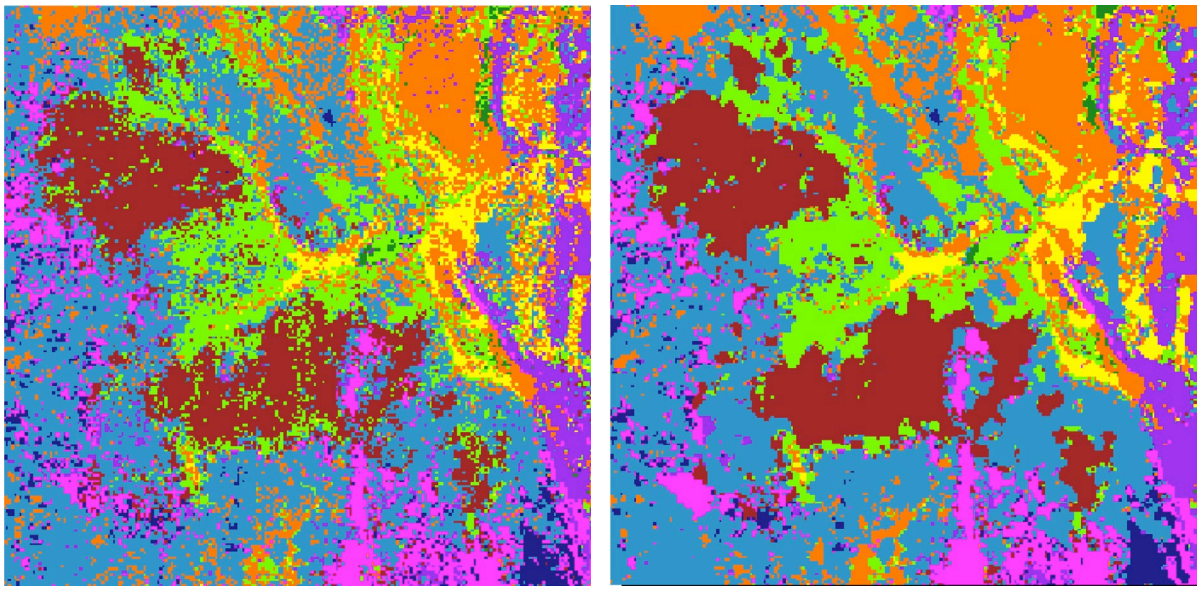
This operation performs a type of 'logical' smoothing by assuming that the central pixel should be related to its surrounding pixels. Typically, in an image classification of EO data for example, patch size will be larger than a single pixel so that patches consisting of only one pixel are likely to be noise. When the classification is represented by a single image channel, modal filtering can be used to replace isolated pixel patches with the most commonly occurring class in the filter region. Modal filtering can often be conducted in conjunction with thresholds, which only permit the centre pixel to be replaced by a filtered value if the mode occurs in a minimum number of pixels within the filter region (see Section 1.3).

This filtering operation is particularly useful for nominal data, such as an image classification channel, where the original image values can be reorganised but not interpolated (see Figure 5.8). When a filter region is bi-modal, that is there are two modes, a predefined rule determines whether the one with the lowest or highest image value is used. The automatic selection of the lowest or highest value in bi-modal filter regions assumes that an implicit priority order exists in the nominal data, which should be considered when the data are coded.

Figure 5.8 Modal filtering of classified image

a. Original classified image

b. Classified image after three passes of 3x3 modal filter



Source: Megan Lewis, University of Adelaide

Nominal data may be derived from EO imagery, such as a classification channel, or created to represent ancillary data from map sources. The modal filter is a very effective image 'cleaning' tool after classification (see Volume 2E).

5.4 Edge-preserving filters

Edge-preserving smoothing filters can be designed using adaptive windows within the filter (see Section 1.3). This technique involves selecting the best sub-window to use for the filtering process. The selection is based on some relevant criterion such as minimum variance as illustrated in Figure 2.5. This process removes patches smaller than the filter size and results in an image in which the boundaries of major patches are preserved rather than blurred as would be typical for smoothing operations. Cheng (1989) considers the use of an adaptive noise smoothing filter for restoring airborne scanner images with microphonic noise. The effect of edge-preserving filtering is illustrated in Figure 5.9.

Figure 5.9 Effect of filter size on edge-preserving and edge-enhancement filters

The original image is shown in Figure 2.4a (or repeat here?)

	Filter operation	
Filter size	**Edge-preserving**	**Edge-enhancing**
3x3	<image>	<image>
7x7	<image>	<image>
11x11	<image>	<image>

Source: Harrison and Jupp (1990) Plate 4

5.5 Despiking (Noise Reduction)

A specialised use of image smoothing, called despiking, may be applied to imagery to remove outlier pixels, or noise 'spikes'. These pixels can occur in features of relatively uniform 'colour' and have unrealistically high or low value. For example, image spikes commonly occur in water features, so despiking is a standard pre-processing operation in various shallow water mapping analyses.

Despiking can be simply applied to imagery in two ways:

- removing the spike pixels with specialised algorithms then filling them with the neighbourhood mode value; or
- using a median filter with 70% thresholds.

Some specialised algorithms identify spikes using a median filter with selected thresholds then set those pixels to the null value in the output image. Appropriate thresholds for filtering can be determined either by trial and error, or statistically (such as three standard deviation units; see Volume 2A—see Section 8.1.2). The null pixels can be subsequently filled using the mean, median or mode value of the surrounding pixels. This sequence is the preferred method for despiking since it uses the median to detect outliers then allows them to be replaced using some other criteria. The mode value is recommended in this sequence for classified channels.

Using the median value to replace noisy pixels enforces the use of an existing image value, which is more reliable than the neighbourhood mean value for spikes covering more than one pixel. The use of threshold values during filtering ensures that a pixel is only altered if its value differs from the median of the filter by 70% or more. The effect of this filtering operation is shown in Figure 5.10.

Figure 5.10 Despiking

a. Original image showing significant noise spikes

b. Despiked image (describe process – identified using 70% threshold then replaced with median filter value)

Source: ????

Add text from 2E Section 3.4.3.2

5.6 Filling Missing Values (Interpolation)

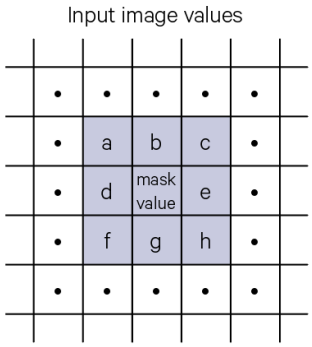
Filtering techniques may be used to interpolate missing values in image data. Such pixels are usually given the image null value. Scanner or reception malfunctions can occasionally cause lines or partial lines in EO images to be 'lost'. During processing, pixels may be deemed 'missing' after certain transformations (such as smoothed ratios with themes—see Section 3.3) or left unclassified after image analysis (see Volume 2E). The boundaries of segmented imagery may also be filled lightly to reduce jaggedness (see Volume 2A—Section 9). Alternatively, data from non-EO sources, such as water depth transects or terrain elevation, may be registered with an image base but form a sparse dataset relative to the image grid. Provided such data does not contain large contiguous areas of missing values, filter interpolation techniques may be used to fill the missing values.

Such techniques operate in a similar fashion to image smoothing with only the values of the pixels surrounding the central 'missing' pixel being used to compute the filled value. This process is illustrated in Figure 5.11.

Figure 5.11 Filter-based interpolation

Whenever the central pixel in a 3×3 neighbourhood is equal to a specified mask value (eg. null), it can be replaced by the median, mode or weighted mean (as selected) of the surrounding 'active' pixels (that is, values not equal to the specified value). A minimum number of active pixels can be specified before the central pixel is filled. In this case, the output value for the central mask value can be:

- median of (a, b, c, d, e, f, g) or
- mode of (a, b, c, d, e, f, g) or
- weighted mean of $\frac{a+4b+c+4d+4e+f+4g+h}{20}$



Source: Harrison and Jupp (1990) Figure 76

Filter interpolation is generally based on a 3×3 mean, median or modal filter to fill missing pixels. An average filter would use predefined weights, such as:

1/20	4/20	1/10
4/20	•	4/20
1/20	4/20	1/20

The missing pixel value may generally be selected as any valid pixel value in the image data range (for example 0–255 for byte format images). To control the effect of this transformation, a minimum number of neighbouring non-missing pixels could be specified to ensure that the operation is only applied to isolated missing pixels. If more than one pixel is missing in the neighbourhood, the filter weights would need to be adjusted accordingly to preserve image scaling.

With median filtering, if an even number of 'active' (non-missing) pixels occur in the filter region, a decision needs to be made as to which value is used in the output image. In some image processing systems, the lower of the two mid-range values is used as the output median value. Modal interpolation filtering of a labelled classification channel provides a convenient method for improving spatial continuity or assigning spatial labels to unclassified pixels (see Volume 2E).

While large targets may be filled iteratively using this process, care must be taken to ensure that the results represent a reasonable interpolation of the original data. Filter interpolation is not appropriate for large area interpolation; it simply provides a mechanism for tidying spotty image data or creating a continuous data channel from thickly sampled ancillary data.

Add image example?

5.7 Further Information

Gonzalez and Woods (2018) Section 3

Jensen (2016) Section 8

5.8 References

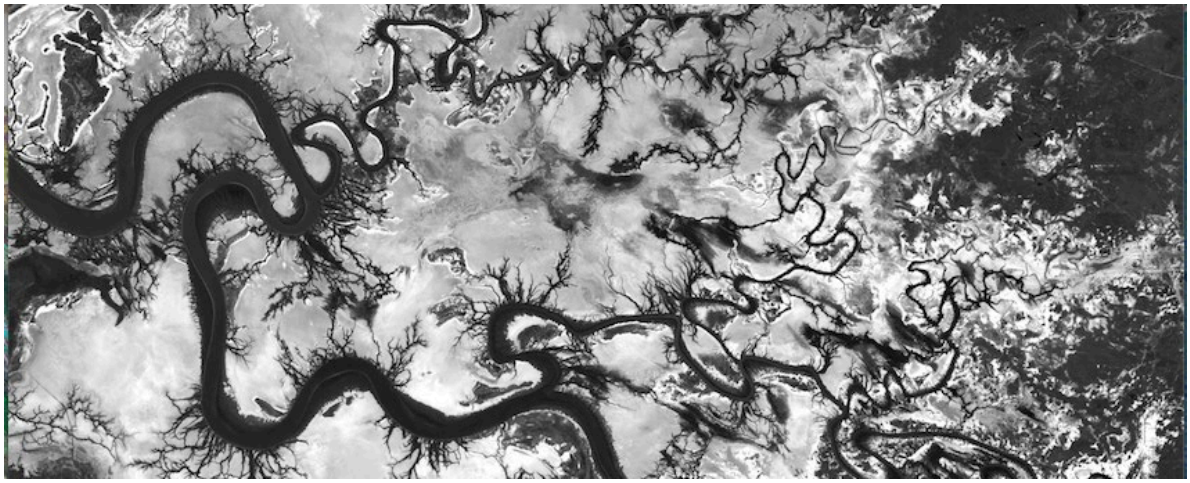
Cheng, S. (1989). An adaptive noise smoothing filter for remotely sensed images with microphonic noise. Int. J. Remote Sensing 10, 1015–34.

Gonzalez, R.C., and Woods, R.E. (2018) *Digital Image Processing*. Pearson Educational Inc., New York.

Harrison, B.A., and Jupp, D.L.B. (1990). *Introduction to Image Processing: Part TWO of the microBRIAN Resource Manual.* CSIRO, Melbourne. 256pp.

Jensen, J.R. (2016). *Introductory Digital Image Processing: A Remote Sensing Perspective.* 4th edn. Pearson Education, Inc. ISBN 978-0-13-405816-0

6 Highlighting Edges



Background image: Panchromatic image acquired by Landsat-8 on 14 April 2014 over Norman River in northwest Queensland, which feeds into the Gulf of Carpentaria.

Source: Craig Shephard, DSITI.

Edge features in EO imagery can be highlighted using differential filters. The following sub-sections consider filters that:

- detect edges (see Section 6.1); and
- enhance edges (see Section 6.2).

6.1 Edge Detection (Differential)

The filters described for image smoothing manipulate image values using balanced filter weights around the central pixel value (see Section 5). The effect of smoothing operations is to effectively reduce the strength of the central pixel value relative to those other pixels surrounding it.

In a 'balanced' filter, the same weighting values are used for all adjacent pixels at the same linear distance from the central pixel. Filters can also be designed as 'unbalanced' to highlight or reduce the effect of image features that have a directional component such as roads, lineaments or detector striping. Such filters basically rely on differential calculus, or derivatives, to achieve their result.

Before considering the design of differential filters, we will review the meaning of the derivative in Excursus 6.1.

Excursus 6.1 Differential calculus review

From high school mathematics you will remember that the first derivative of a one-dimensional function, say

$$f(x)$$

is written as

$$f'(x) \text{ or } \frac{df}{dx}$$

In terms of image values, *x* could represent a row or column of pixels in an image that have the values $f(x)$ where these values can somehow be modelled in terms of pixel position. For example if the image values across

an image line are as shown in Figure 6.1a, we could represent these as a graph (or a 'spectral transect') as shown in Figure 6.1b then plot a continuous function line to describe the changes in image value in terms of position along the line. If we know the equation of this line we could use differential calculus to determine the derivative equation and then compute a derivative value for each pixel along the line. For non-continuous functions, such as image data, these calculations can be approximated using filter-based processes.

Figure 6.1 First and second derivative for spectral transect

a. Pixel values across image line

1	4	6	7	8	9	11	9	8	7	6	4	1	0	1	2	4	7

b. The *X* axis represents pixel positions along an image line and the *Y* axis shows the pixel values

c. Central difference values

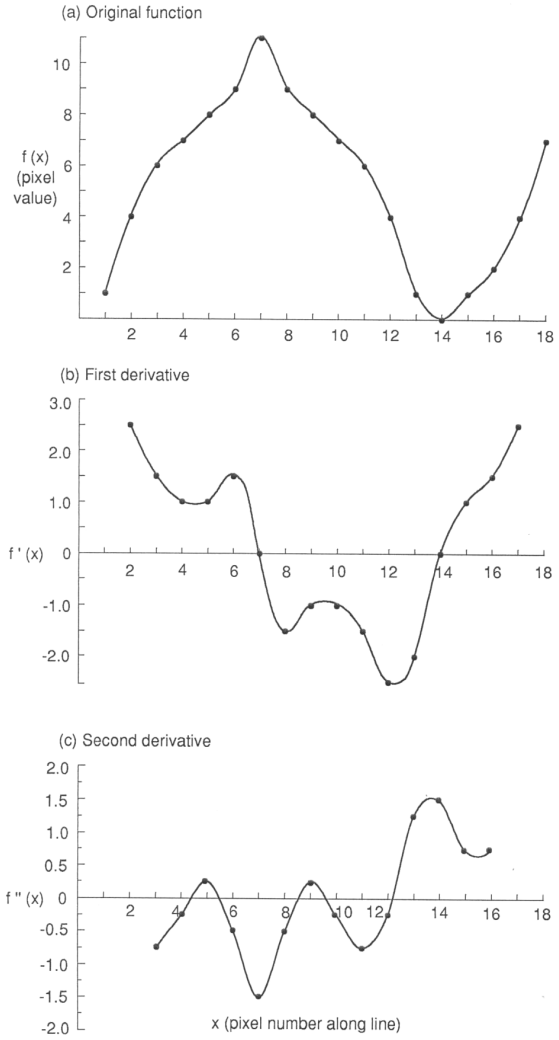
•	2.5	1.5	1.0	1.0	1.5	0	-1.5	-1.0	-1.0	-1.5	-2.5	-2.0	0	1.0	1.5	2.5	•

d. The first derivative indicates the changes in slope of the original function

e. Second derivative values

•	•	-0.75	-0.25	-0.25	-0.5	-1.5	-0.5	0.25	-0.25	-0.75	-0.25	1.25	1.5	0.75	0.75	•	•

f. The second derivative indicates the direction and extent of curvature in the original function



Source: Harrison and Jupp (1990) Figure 77

The initial function value, $f(x)$, describes the relationship between two variables—in the case of Figure 6.1a these were the pixel value and its position along a line. To determine the way values change along the line, we can look at the slope of the function line and identify where the values are maximum, minimum, increasing or decreasing. What we are really extracting from the graph is a visual measure of the change in the $f(x)$ value caused by a change in the *x* values. These observations may also be performed analytically using differential calculus.

Mathematically these changes can be expressed as the function:

$$\frac{df}{dx} = f'(x) = \lim(\delta x \to 0)\frac{\delta f}{\delta x}$$

where

$\frac{df}{dx}$ is referred to as the derivative of *x*

δf is the change in $f(x)$ corresponding to δx

δx is the change in x values

When the function is defined at points which are a constant distance apart, a number of operators may be use to measure its rate of change. For a forward difference, that is, the difference between values as *x* increases:

$$\Delta_F f(x) = f(x+h) - f(x)$$

where *h* is the δx, that is the number of pixel positions, or lag, between values. For a backward difference:

$$\Delta_B f(x) = f(x) - f(x-h)$$

or for the central difference (the average of the forward and backward differences):

$$\begin{aligned}\Delta_C f(x) &= \frac{f(x+h) - f(x)}{2} + \frac{f(x) - f(x-h)}{2} \\ &= \frac{f(x+h) - f(x-h)}{2}\end{aligned}$$

These difference formulae give different results where *h* is not equal to, or close to, 0. The derivative function defines the way the function changes as *h* approaches 0. Based on forward difference, the derivative function becomes:

$$f'(x) = \lim(h \to 0)\Delta_F \frac{f(x)}{h}$$

Similarly, the 'backward derivative' is:

$$f'(x) = \lim(h \to 0)\Delta_B \frac{f(x)}{h}$$

while the derivative computed from the central difference is defined as:

$$\begin{aligned}f'(x) &= \lim(h \to 0)\Delta_C \frac{f(x)}{h} \\ &= \lim(h \to 0)\frac{f(x+h) - f(x-h)}{2h}\end{aligned}$$

For a smooth (or 'differentiable') function all these operators give the same result.

Using the example in Figure 6.1a, we can compute the (central difference) derivative values for *h* = 1 as illustrated in Figure 6.1e, which can be plotted as shown in Figure 6.1b. Where we have increasing values in the original function, its derivative values are positive, while decreasing values in the original function give negative derivatives and the minimum and maximum values in the original function have zero values in the derivative. Thus the derivative function has quantified the change in slope of the original function.

The slope of the derivative function itself contains information about the curvature of, or the change of slope in, the original function. Line sections of constant slope in the original function have derivative values with no slope, while sections in the original function with gradually increasing slope, such as pixels 14–18 have a positive slope in the derivative, and conversely sections of gradually decreasing slope in the original function have a negative derivative slope. Thus where the rate of change in the original function varies (such as around pixel 6) we get a local peak or trough in the derivative graph, a peak corresponding to a decrease and a trough to an increase.

These observations about the derivative function are also important for interpreting our original function. Since they all relate to the slope of the derivative function, they may be simply quantified by computing the derivative of the derivative function—which is commonly called the second derivative or $f''(x)$.

For our example, the second derivative values would be as shown in Figure 6.1f, and could be graphed as illustrated in Figure 6.1c. This function clearly identifies the direction and extent of curvature of the original function: $f''(x)$ values with no slope indicate constant slope in $f'(x)$ or constant rate of change in $f(x)$;

positive values in $f''(x)$ show constantly increasing slope in $f'(x)$ or increasing rate of change in $f(x)$ and negative values show constantly decreasing slope in $f'(x)$ or decreasing rate of change in $f(x)$. The points of inflection in $f(x)$, that is, where a change in slope occurs, are 0 in $f''(x)$ and in this example occur between pixel values.

6.1.1 Image-based derivatives

In image data, derivative functions may be simply derived along rows or columns in the image or along defined transects such as diagonals. These derivatives can then be used to indicate the rate of spatial variation in the defined direction. Where a dataset has more than one spatial dimension, the derivatives are computed with respect to one dimension at a time and are termed partial derivatives. For image data with two spatial dimensions, the terminology frequently used is X for across lines and Y down pixel columns in the image, with f_x or $\frac{\partial f}{\partial X}$ being the first partial derivative function computed in the X direction for a given Y value (that is, an image line) and f_y or $\frac{\partial f}{\partial Y}$ being the first partial derivative in the Y direction for a given X value (that is, an image pixel column).

When the image data represent samples from a smooth underlying function (such as geophysical data), the derivative function may be approximated using filters. For example, the difference (or error) between the value for the central difference used as an approximation to the derivative and $f'(x_i)$ can be shown (Conte and de Boor 1965) to be:

$$\left| f'(x_i) - \frac{\left(f'(x_{i+1}) - f'(x_{i-1})\right)}{2h} \right| = \frac{h^2}{6} \times \left| f'''(y) \right|$$

where $x_i - 1 \leq y \leq x_i + 1$. That is, when the step h is small, and the underlying function is smooth (hence the third derivative is also small), the central difference can be a very close approximation to the actual derivative.

To calculate the derivative values across a line in an image for $h = 1$ pixels, we can compute the difference between pairs of pixels which are separated by one pixel. This spatial difference can be easily calculated using a filter with the weights:

-1/2	0	1/2

Similarly, to compute derivative values down columns in the image, we could use the filter:

-1/2
0
1/2

The results of applying some simple derivative filters are given in Figure 6.2.

Figure 6.2 Effect of filter direction on differential filters

The original image is shown in Figure 2.4a. The striping patterns visible in a. and c. are due to slight scanner miscalibration which is not discernible in the orginal image. Differential filters accentuate this type of image 'noise' as well as feature boundaries.

Differential filter direction	3x3 filter
a. Horizontal	<image>
b. Vertical	<image>

c. Diagonal 

Source: Harrison and Jupp (1990) Plate 5

A filter with the weights:

-1/6	-1/6	-1/6
0	0	0
1/6	1/6	1/6

also uses the spatial variation of the two adjacent image columns to compute the central value. This filter is formed from two one-dimensional filters:

-1/2
0
1/2

×

1/3	1/3	1/3

so is equivalent to carrying out the two operations of smoothing and differentiating. Since the differential highlights the differences between pixel values, the use of adjacent pixel values to smooth the image values before differencing usually produces better results.

We can effectively vary the h value used for the derivative by varying the filter size. For example, for h=2 we could use the filter:

-1/4	0	0	0	1/4

Similarly, the direction for the filter may be expressed in filter weights. For example, a 'diagonal' filter could be:

-1/2a	0	0
0	0	0
0	0	1/2a

where a is √2. The 'a' term accounts for the longer distance to the centres of diagonally adjacent pixels from the centre pixel as compared with those that are in the same row or column. The differential filter weights are usually chosen so that the result is 'exact' for some function, such as a linear function, which implies that the weights sum to zero. This means that the filter would produce constant values when applied to a linear function (that is, a constant slope) or zero when applied to constant data values. In practice though, if the 'a' value equals 1 in this filter, the image scaling would be preserved satisfactorily.

Second derivative values may be computed by re-applying the same filter to a previously filtered channel (or by convolving the two filters as described in Section 1.1). Mixed derivatives may also be computed at this stage by applying the Y filter to the first partial derivative in X or the X filter to the first partial derivative in Y to incorporate two-dimensional information. The second order partial derivatives are:

$$f_{xx} = \frac{\partial}{\partial X}(f_x) = \frac{\partial}{\partial X}\left(\frac{\partial f}{\partial X}\right) = \frac{\partial^2 f}{\partial X^2}$$

$$f_{xy} = \frac{\partial}{\partial Y}(f_x) = \frac{\partial}{\partial X}\left(\frac{\partial f}{\partial Y}\right) = \frac{\partial^2 f}{\partial X \partial Y}$$

$$f_{yx} = \frac{\partial}{\partial X}(f_y) = \frac{\partial}{\partial Y}\left(\frac{\partial f}{\partial X}\right) = \frac{\partial^2 f}{\partial Y \partial X}$$

$$f_{yy} = \frac{\partial}{\partial Y}(f_y) = \frac{\partial}{\partial Y}\left(\frac{\partial f}{\partial Y}\right) = \frac{\partial^2 f}{\partial Y^2}$$

This set of second partial derivative values can also be treated as a matrix and manipulated to determine the direction and magnitude of the maximum curvature within the filter region. A larger filter size, such as 7×7 should be used to compute surface curvature (see) Section 7.4).

A simple estimation of the derivative function is also commonly computed from the difference between a central pixel and a selected neighbour. This first difference effectively implements a forward or backward first derivative and can be computed as:

$$diff(x) = f(x_i) - f(x_{i-1})$$

or using a simple filter with the weights:

1	-1

The first differences in X and Y directions, $diff(x)$ and $diff(y)$, approximate the first partial derivatives, f_x and f_y). Various distance functions, or 'norms' ($||.||$), can be used to estimate the 'gradient' or the magnitude of the maximum directional difference (see Figure 6.3):

$$\|\cdot\|_2 = \sqrt{diff(x)^2 + diff(y)^2}$$

$$\|\cdot\|_1 = |diff(x)| + |diff(y)|$$

□□

$$\|\cdot\|_\infty = \max\left[|diff(x)|, |diff(y)|\right]$$

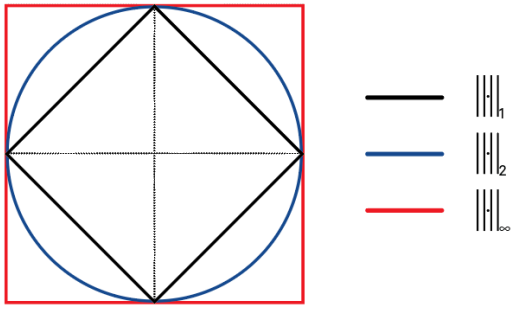
Moik 1980). The first differences in X and Y can also be used to compute the difference in a particular direction θ (where θ is measured anti-clockwise from the X axis) as:

$$diff(\theta) = diff(x) \times \cos\theta + diff(y) \times \sin\theta$$

(A similar approach is used for directional filtering—see Section 7.1).

Figure 6.3 Norm functions

These can be used to estimate the maximum directional difference from two orthogonal distance measurements.



Source: Harrison and Jupp (1990) Figure 78

Some commonly used differential filters are detailed in Excursus 6.2.

Excursus 6.2 Commonly used differential filters

The Roberts cross operator (Roberts 1965) uses diagonal first differencing in two directions with the filters:

0	1
-1	0

and

1	0
0	-1

and computes the magnitude and direction of the gradient using the equations given above.

The Sobel operator (Levine 1985) uses a similar approach to the Roberts operator with the two differential filters:

-1	0	1
-2	0	2
-1	0	1

and

-1	-2	-1
0	0	0
1	2	1

The Prewitt operator (Prewitt 1970) varies these weights slightly as:

-1	0	1
-1	0	1
-1	0	1

and

-1	-1	-1
0	0	0
1	1	1

These filters incorporate local averaging to reduce noise effects so tend to produce superior results to the Roberts operator for 'sharp' edges. The use of zero weights along the 'edge' is reported to reduce jitter (Levine 1985).

A second difference, which approximates the second derivative, can also be computed as:

$$diff_2(x) = \left(f(x_{i+1}) - f(x_i)\right) - \left(f(x_i) - f(x_{i-1})\right)$$
$$= f(x_{i+1}) + f(x_{i-1}) - 2f(x_i)$$

or with the filter:

1	-2	1

As illustrated in Figure 6.4, the first difference measures the change between two adjacent pixels whereas the second difference indicates the extent of change across a pixel. Images produced by second difference filters highlight edges as adjacent light and dark lines, whereas first difference filters result in single pixel edges.

Figure 6.4 First and second differences

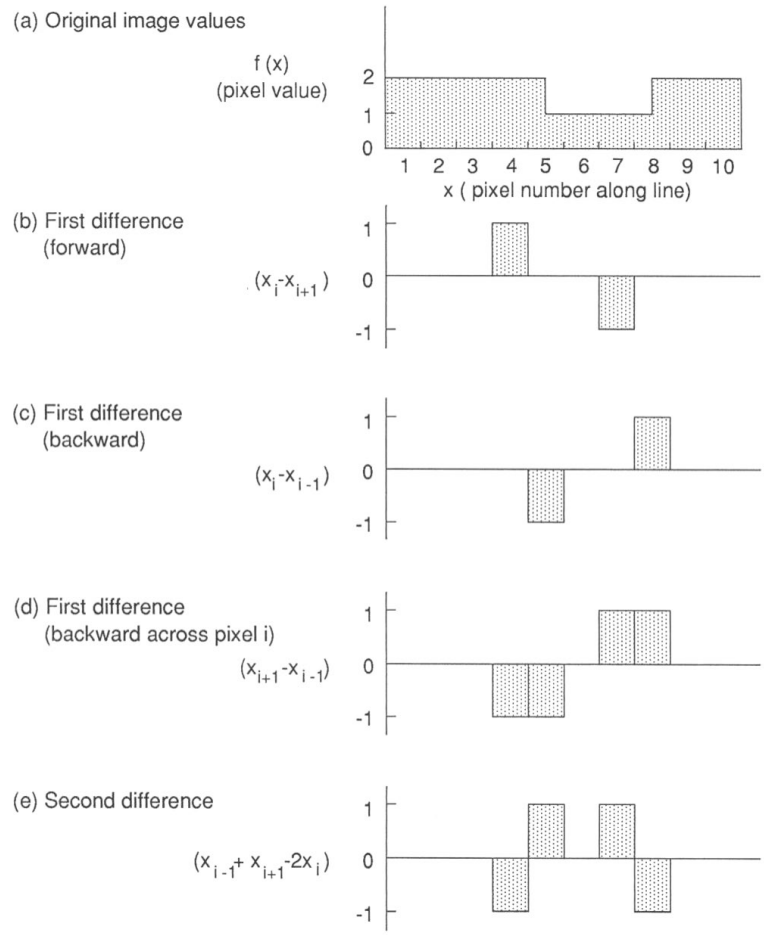
a. Original image values

b. First forward difference

c. First backward difference

d. First difference backward across pixel i

e. Second difference



Source: Harrison and Jupp (1990) Figure 79

An effective edge detector is the Laplacian operator (an approximation of the mathematical Laplacian: $f_{xx} + f_{yy}$) which combines second differences in *X* and *Y* directions. This omni-directional operator can be implemented with the filter weights:

-1	-1	-1
-1	8	-1
-1	-1	-1

This operator is sometimes referred to as the centre-point 2-ring (or nine point) filter in geophysics. Another commonly used bi-directional filter, which is also referred to as a Laplacian operator (or the centre-point 1-ring filter), is:

< should operator below be + or x?>

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & -2 & 1 \end{bmatrix} \times \begin{bmatrix} -1 \\ -2 \\ 1 \end{bmatrix}$$

This basic filter can also be rotated to highlight diagonal edges with the weights:

$$\begin{bmatrix} a & 0 & a \\ 0 & -4a & 0 \\ a & 0 & a \end{bmatrix}$$

where $a = \sqrt{2}$, or convolved to operate on a larger image neighbourhood (see Section 1.1). Some specific derivative filters which were developed for geophysical data and other directional filters are detailed in Volume 2X—Appendix 5.

6.1.2 Uses of edge detection

Edge detection techniques are useful for a number of remote sensing applications. For example, the effects of using horizontal and vertical filters on rock surfaces are illustrated in Figure 6.5. Ton *et al.* (1989) outline a method for automatic road identification, which involves the use of edge detection filters to locate roads and categorise their 'status'. Paine and Lodwick (1989) report a multi-stage approach for extracting edges in an image to identify region boundaries which involves smoothing, edge detection, thresholding, thinning and linking. The ultimate result from this type of processing would be a set of polygons that are suitable for inputting to a Geographic Information System (GIS).

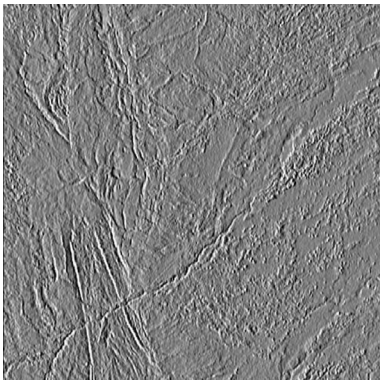
Figure 6.5 Horizontal and vertical filters

Landsat TM image of ???, date, band 3

a. Original image

b. After filtering with horizontal filter

c. After filtering with vertical filter



Source: Megan Lewis, University of Adelaide

Jacobberger (1988) reports an interesting application of directional filtering to map abandoned river channels in Landsat Thematic Data. Linear dunes in the study area obscured the morphology and trend of the channel system so directional filtering was used to selectively suppress edges parallel to the dune system. In this case the filter was based on a Laplacian operator that was scaled to the width and orientation of the dune edges in the image. The first principal component (PC1—see Sections 2.2 and 9) was used for filtering then the residual between PC1 and the filtered channel was used for mapping. This residual image enhances the high frequency detail that is not related to the dune system at the expense of detail that is parallel to it.

Filters which effectively compute higher partial derivatives also exponentially enhance image noise. Noisy images should be appropriately pre-processed to reduce noise effects (as discussed in Sections 8.2 and 8.6) before edge detection filtering.

Differential or derivative filters may be applied in most image processing systems. This typically involves selected weightings (which may be defined as integer or real values) in an average filter (see for a selected filter size (see Figure 8.1 and Section 5.1). The effects of varying filter sizes for differential filters are illustrated in Figure 6.6.

Figure 6.6 Effect of filter size with differential filters

Landsat MSS band 1 of Kennedy Ranges, Western Australia

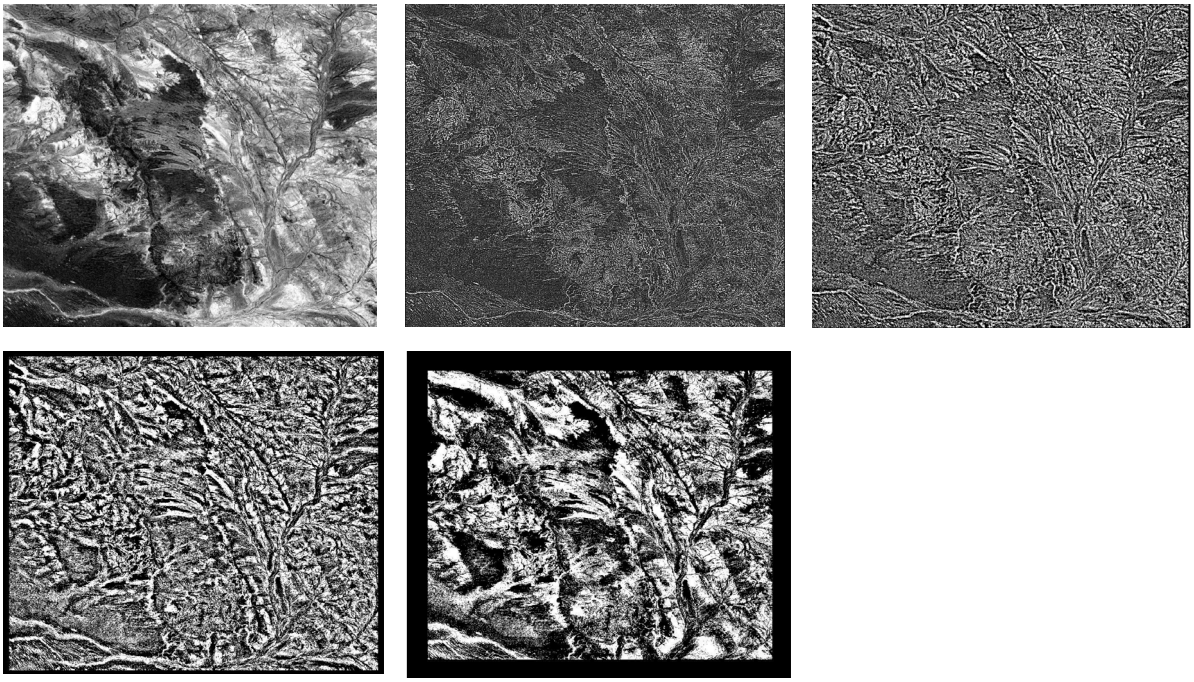
a. Original image

b. 5x5 filter

c. 11x11 filter

d. 31x31 filter

e. 101x101 filter



Source: Megan Lewis, University of Adelaide

Thresholding can be useful in clearly identifying edges in noisy images. If the threshold is set too high, low-amplitude edges may not be detected but a low threshold will result in noise being highlighted as edges. Edges in a multi-channel imagery may be summarised into a single channel by averaging (see Section 5.1) or Principal Components Analysis (see Section 9). Density slicing of edge channels can also be used to separate edge information from other image features (see Volume 2A—Section 9.2.1).

The interpretation of derivative results depends on the image data being processed. For example, with elevation data the first derivative represents slope (or gradient) and the second derivative indicates surface curvature (see Section 7.4). Some image processing packages can also compute slope and aspect for an image channel containing elevation data. Typically a 5×5 filter is used to estimate the local derivatives and produces an image with a linearly scaled slope (or 'gradient') channel in which the minimum image value indicates a slope of 0° and the maximum value represents a slope of 90°. Aspect is usually linearly scaled from north at the minimum value through the mid-range value for an orientation of 180° to maximum value corresponding to 359°. These results are further discussed in Section 7.4.

6.2 Edge Enhancement (High Pass)

Many image analysis tasks require that the features in an image be clearly defined in terms of spatial pattern and boundaries. For example, accurate positioning tasks requires that linear features, such as roads, rivers or various boundaries be clearly delineated in the image (see Volume 2B). Geological applications, in particular, frequently use edge enhancement methods to highlight lineaments. Image filtering transformations can be used to enhance this edge-related detail. Such filters are also referred to as high pass or residual filters since they enhance the high frequency variations in imagery.

Section 1.1 and Figure 2.4 introduced the idea of image smoothing and showed that a smoothed image could effectively be defined as:

$$\textit{smoothed image} = \textit{original image} - \textit{edges}$$

or

$$\textit{edges} = \textit{original image} - \textit{smoothed image}$$

Conversely, an edge enhanced image could be described as:

$$\textit{edge enhanced image} = \textit{original image} + \textit{edges}$$

or

$$\begin{aligned}\textit{edge enhanced image} &= \textit{original image} + (\textit{original image} - \textit{smoothed image}) \\ &= 2 \times (\textit{original image}) - \textit{smoothed image}\end{aligned}$$

An example of edge enhancement filtering is shown in Excursus 6.3.

Excursus 6.3 Edge Enhancement Filtering

We can represent an original image and its smoothed image as the filters shown in Figure 6.7.

Figure 6.7 Example images

Original image values

-1	-1	-1
0	0	0
1	1	1

Smoothed image values

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Using some elementary arithmetic we can then derive the weights for a simple edge enhancement filter as shown in Figure 6.8.

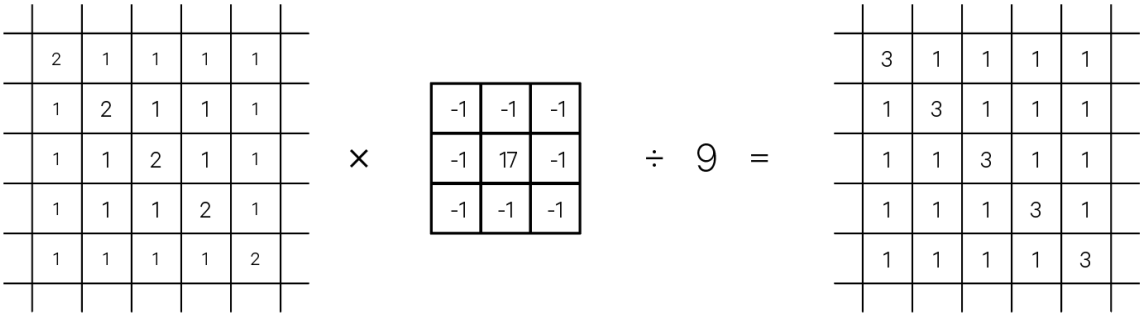
Figure 6.8 Edge enhancement filter derivation

0	0	0	+	0	0	0	–	1/9	1/9	1/9	=	-1/9	-1/9	-1/9	or	-1	-1	-1	with divisor = 9
0	1	0		0	1	0		1/9	1/9	1/9		-1/9	17/9	-1/9		-1	17	-1	
0	0	0		0	0	0		1/9	1/9	1/9		-1/9	-1/9	-1/9		-1	-1	-1	

The effect of edge enhancement using this filter can be seen in Figure 1.4. This example filter has a typical pattern of weighting values for edge enhancement, with the central value being large and positive and surrounding values being small and negative. The effect of the filter is to exaggerate any difference between the central pixel and those pixels adjacent to it. The operation of this filter may be best understood using an example image as shown in Figure 6.9 below.

Figure 6.9 Operation of edge enhancement filter

Differences between the central pixel and its neighbours are exaggerated using a filter that has a large central value with small edge values.



Source: Harrison and Jupp (1990) Figure 80

Various edge detection filters such as the Laplacian operator are described in Section 6.1. Following the opening discussion in this Section, an edge enhancement filter can be simply developed by adding an edge detection filter to the original image filter, such as:

0	0	0	+	-1/9	-1/9	-1/9	=	-1/9	-1/9	-1/9				
0	1	0		-1/9	8/9	-1/9		-1/9	17/9	-1/9				
0	0	0		-1/9	-1/9	-1/9		-1/9	-1/9	-1/9				

An effective edge enhancement filter involves subtraction of the first difference Laplacian from the original (the weightings of this Laplacian required subtraction rather than addition to produce a 'positive' image):

0	0	0	+	0	1	0	=	0	-1	0				
0	1	0		1	-4	1		-1	5	-1				
0	0	0		0	1	0		0	-1	0				

Niblack (1986) refers to this filter type as being similar to the photographic process of 'unsharp masking'. Edge enhancement filters can similarly be derived from other edge detection filters, such as the directional templates.

Edge enhancement transformations may be applied in most image processing systems with user-defined weights in an average filter (see Figure 1.4 and Figure 10.2). As with edge detection filters, thresholding or pre-processing to remove noise is often beneficial to avoid highlighting noise as well as edges.

6.3 Further Information

Jensen (2016) Section 8

Gonzalez and Woods (2018) Section 3

6.4 References

Conte, S.D., and de Boor, C. (1965). Elementary Numerical Analysis. McGraw-Hill Kogakusha Ltd, Japan.

Gonzalez, R.C., and Woods, R.E. (2018) *Digital Image Processing*. Pearson Educational Inc., New York.

Harrison, B.A., and Jupp, D.L.B. (1990). *Introduction to Image Processing: Part TWO of the microBRIAN Resource Manual.* CSIRO, Melbourne. 256pp.

Jaccobberger, P.A. (1988). Mapping abandoned river channels in Mali through directional filtering of TM data. Remote Sensing of Environment 26, 161–70.

Jensen, J.R. (2016). *Introductory Digital Image Processing: A Remote Sensing Perspective.* 4th edn. Pearson Education, Inc. ISBN 978-0-13-405816-0

Levine, M.D. (1985). Vision in Man and Machine. McGraw-Hill Inc. USA.

Moik, J.G. (1980). Digital Processing of Remotely Sensed Images. NASA SP–431. Washington, DC. USA.

Niblack, W. (1986). An Introduction to Digital Image Processing. Prentice-Hall International. New Jersey.

Paine, S.H., and Lodwick, G.D. (1989). Edge detection and processing of remotely sensed digital images. Photogrammetria 43, 323–36.

Prewitt, J.M.S. (1970). Object Enhancement and Extraction. In Picture Processing and Psychopictorics. Eds B.S. Lipkin and A. Rosenfeld. Academic Press, New York. pp 75–149.

Roberts, L.G. (1965). Machine perception of three-dimensional solids. In Optical and Electro-optical Information Processing Ed. J.T. Tippett. MIT Press, Cambridge, Mass. pp 159–97.

Ton, J., Jain, A.K., Enslin, W.R., and Hudson, W.D. (1989). Automatic road identification and labelling in Landsat TM images. Photogrammetria 43, 257–76.

7 Highlighting Surface Variation

Background image: Astronaut photograph of Gosses Bluff crater, in southern Northern Territory, acquired from the International Space Station on 18 March 2005 from an altitude of 328 km.

Source: NASA ISS Photo retrieved from Wikimedia Commons: https://commons.wikimedia.org/wiki/File:ISS007_Gosses_Bluff.jpg

Local surface variations are called image 'texture' and can be characterised using a range of statistics (see Section 7.1). A variety of differential filters can also be adapted to accentuate surface variation. Such filters can be used to transform imagery to highlight:

- shaded relief (or insolation)—see Section 7.2;
- exposure to surface winds—see Section 7.3; or
- surface curvature—see Section 7.4.

7.1 Texture (Local Variance)

Image texture has been measured and characterised by a wide variety of methods (Haralick 1979; Levine 1985). Texture transformations use spatial statistics to compute the local standard deviation at each pixel. In this context, the square root of the local variance, is called 'texture' (see Volume 2A—Section 8.2.1), which statistic can be normalised by the mean of the pixel neighbourhood values to produce the 'coefficient of variation'.

A 3×3 filter is used to define the neighbourhood of each pixel. Within this filter region, the texture (T_{ij}) is computed by:

$$T_{ij} = \sqrt{\frac{1}{2n} \times \sum \left(x_{ij} - x_{kj} \right)^2}$$

where

x_{ij} is the central value of a 3×3 window centred at column i and row j

x_{kl} is the pixel value at column k and row l where $k = i \pm 1$ and $l = j \pm 1$.

n is the number of pixels involved in the calculation—usually 8 but sometimes less if there are null pixels within the 3×3 window.

Where $n = 8$, this equation becomes:

$$T_{ij} = \sqrt{\frac{1}{16} \times \sum \left(x_{ij} - x_{kj} \right)^2}$$

This texture value can be normalised by x_{ij}, the mean of the pixels in the filter neighbourhood, to give the Coefficient of Variation (C_{ij}) as:

$$C_{ij} = \frac{T_{ij}}{x_{ij}}$$

The maximum texture value is generally difficult to estimate so is usually determined iteratively. By default, this can be estimated theoretically, using the minimum and maximum values for a given image channel, by computing the texture value of a 3×3 neighbourhood where the central pixel has the minimum value and the surrounding pixels are equal to the maximum value.

The texture transformation highlights areas of rapid spatial change, such as edges. In this sense, it is equivalent to the average slope of the image channel 'surface' over the filter region. It has also been used to categorise differences in the spatial structure of image features. For example, at the Landsat MSS pixel level, different forest canopies frequently radiate similar spectral values but can be separated by the difference in their spatial variability or 'texture'. To enhance vegetation texture, this transformation would be best applied to a near infrared image channel or to a vegetation index channel (see Section 11 and Volume 3A).

This spatial analysis is similar to visually considering feature texture in aerial photograph interpretation. The coefficient of variation effectively reduces the typically high texture values in bright areas of an image. Pickup and Foran (1987) observed that the coefficient of variation in multi-temporal imagery provides a measure of the effect of plant cover changes on spectral variability within a landscape. As with other spatial statistics, texture results are dependent on both the radiometric and spatial resolutions of an image (see Volume 1B—Section 1).

As an edge enhancement transformation, texture analysis accentuates boundaries between any spectrally coherent patches (see Figure 7.1). The effect of this enhancement can highlight noise as well as edges. In most cases texture analysis on a smoothed image will reduce this effect.

Figure 7.1 Effect of filter direction on differential filters

The original image is shown in Figure 2.4a. The striping patterns visible in a. and c. are due to slight scanner miscalibration which is not discernible in the orginal image.

Texture filter type	3x3 filter
d. Texture	<image>
e. Coefficient of variation	<image>
f. Edge-preserved texture	<image>

Source: Harrison and Jupp (1990) Plate 5d to 5f

Edge-enhancement effects can be reduced with texture filtering by using adaptive windows (Woodcock and Ryherd, 1989—see Section 1.4). As illustrated in Figure 2.5, the adaptive window technique selects the 'best' sub-window to use within a filter to derive the filtered value. In texture filtering, if the sub-window is selected on the basis of minimum variance, the resulting texture value is more likely to relate to a feature patch in the image than to an edge between two features.

Adaptive windows can be used to compute either smoothing or texture measures for an image. The texture result are generally selected from either the:

- minimum variance (based on the mean);
- deviation (associated with the median); or
- entropy (related to the mode)

of values within a sub-window of each filter region. The minimum variance result is illustrated in Figure 12.1.

The texture transformed values could be used to create an image in which local variance (or standard deviation) is 'constant'. This type of enhancement has been variously referred to as 'constant variance' (Harris 1977), 'adaptive contrast' (Levine 1985) or 'statistical differencing' (Niblack 1986) and results in even contrast over the whole image. The general transformation is defined as:

$$\frac{x_{ij} - \overline{x_{ij}}}{\sigma_{ij} + \overline{x_{ij}}}$$

where

x_{ij} is the image value at pixel i and line j

$\overline{x_{ij}}$ is the mean value of the neighbourhood of x_{ij}

σ_{ij} is the standard deviation of the neighbourhood of x_{ij}

The enhancement is particularly effective for grey-scale imagery with a high dynamic range but little local variation. Medical imagery such as X-rays can be clarified using this technique.

In conjunction with image classification, texture results are best used in a post-classification stage to split up specific, spectrally-defined classes (see Volume 2E). If texture channels are used to define the classification, a large number of classes are likely to be generated, as the boundaries themselves will tend to form separate classes. Gordon and Philipson (1986) suggest the use of ratioing texture-enhanced channels to reduce this characteristic edge brightness (see Section 10). Unlike most classification algorithms, which are pixel specific, texture analysis considers the neighbouring pixel values so can add new information to the classification process.

7.2 Insolation (Relief Shading)

As introduced in Volume 1, insolation describes the relief shading produced on the Earth's surface by a particular sun position. Relief shading is a useful transformation for elevation data as it highlights geomorphological patterns. When elevation data are registered with EO data, the insolation corresponding to the time of image acquisition can be computed to help reduce topographic shading effects in the image. A number of procedures have been suggested that utilise insolation information to reduce the effect of topographic shading in EO imagery.

The insolation can be computed from relief data such as a Digital Elevation Model (DEM) if a reflectance model is assumed for the terrain and the sun position can be defined in terms of altitude (or elevation, that is degrees above the horizon) and azimuth (degrees clockwise from north).

A range of surface reflectance models have been devised for various purposes. For example, the simple cosine reflectance model defines the radiance (R) of a Lambertian surface (see Volume 1A—Section 5 or Volume 1B—Section 3) as:

$$R = R_n(\cos i)$$

where

R is the radiance of a target surface;

R_n is radiance viewed in a direction perpendicular to the target surface

i is the angle between upwelling radiance and the normal to the surface as illustrated in Figure 7.2. This effectively models surface radiance for a Sun position, which is normal to the surface and a sensor directly overhead. The radiance can then be modified for other Sun positions using the topographic modulation factor, which is defined as:

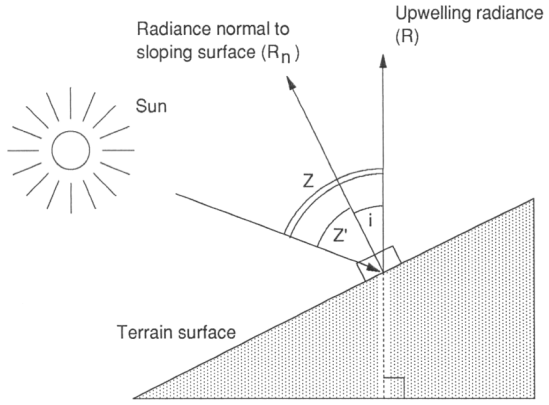
$$\frac{\max(\cos Z')}{\cos(Z,0)}$$

where

Z is the zenith angle of the Sun (the angle from the directly vertical position), and

Z' is the angle between the Sun vector and the normal to the slope (Ahmad 1987).

Figure 7.2 Cosine reflectance model



Source: Harrison and Jupp (1990) Figure 83

As discussed in Section 6.1, the first partial derivatives of an elevation surface can be used to define the direction and magnitude of the slope. The 'downhill' direction of a slope vector is defined as the two-dimensional vector:

$$\begin{bmatrix} -f_x \\ -f_y \end{bmatrix}$$

In a three-dimensional space, a vector can be described in terms of the angles it makes with each of the X, Y and Z axes. This is commonly represented as the direction cosines (that is, the cosines of each of these angles). The vector which is normal (vertically orthogonal) to the slope is described by the direction cosines l, m and n where:

$$l = \frac{p}{\sqrt{1 + p^2 + q^2}}$$

$$m = \frac{q}{\sqrt{1 + p^2 + q^2}}$$

$$n = \frac{1}{\sqrt{1 + p^2 + q^2}}$$

and

$$p = f_x$$

$$q = f_y$$

The Sun vector can similarly be defined as:

$$l_s = \frac{p_s}{\sqrt{1 + p_s^{\,2} + q_s^{\,2}}}$$

$$m_s = \frac{q_s}{\sqrt{1 + p_s^{\,2} + q_s^{\,2}}}$$

$$n_s = \frac{1}{\sqrt{1 + p_s^{\,2} + q_s^{\,2}}}$$

where

$$p_s = \tan Z \times \sin\theta$$

$$q_s = \tan Z \times \cos\theta$$

and

Z is the Sun zenith angle; and

θ is the Sun azimuth angle.

The cosine of the angle between two vectors can be computed as the scalar product of their direction cosines, that is:

$$\begin{bmatrix} l & m & n \end{bmatrix} \times \begin{bmatrix} l_s \\ m_s \\ n_s \end{bmatrix}$$

The cosine of the angle between the sun and the 'normal to the slope' vectors (that is Z') can thus be derived as:

$$\frac{p \times p_s + q \times q_s + 1}{\sqrt{1 + p^2 + q^2} \times \sqrt{1 + p_s^{\,2} + q_s^{\,2}}}$$

Insolation is then computed as:

$$\frac{R \times \cos Z'}{\cos Z}$$

$$R \times \cos Z' / \cos Z$$

which gives a relative value between 0 and 1 for 0°≤ Z ≤ 90° and Z' > 0, or 0 for any other angles. <check with DLBJ – was 90º in original text>

Some image processing systems compute insolation, slope and aspect for an elevation channel using the cosine reflectance model for the terrain. A 5×5 filter (detailed in Volume 2X—Appendix 5) may be used to estimate local derivatives (see Sections 6.1). The first partial derivative results can be used to define the principal gradient for the filter region at each pixel; that is, the vector which is perpendicular to the dominant direction of contour lines within the region. The slope of the gradient corresponds to the length of this vector and may be computed as:

$$\sqrt{f_x \times f_x + f_y \times f_y}$$

The direction of the gradient vector, or aspect, can also be computed from the partial derivatives as:

$$\tan^{-1} \frac{-f_x}{-f_y}$$

Relative insolation, or relief shading, can be computed as detailed above with f_x and f_y of the specified image channel defining the slope vector and user-selected sun azimuth and zenith angles[4] defining the sun vector.

Input image geometry needs to be defined in terms of scaling in X (pixel width), Y (pixel depth) and Z (height scaling—the relationship between actual height and the values coded in the image channel) dimensions, as well as orientation relative to north. An atmospheric correction factor may also be specified to account for absorption and backscatter, which would reduce the measurement of surface radiance in conditions of haze or cloud.

Additonal properties related to insolation include:

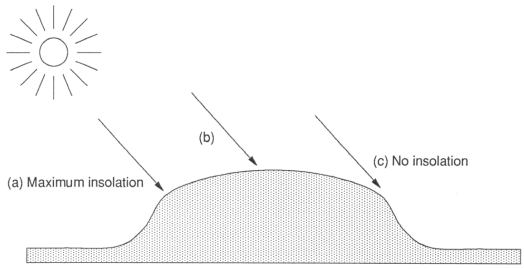
- a smoothed elevation value (see Section 5.4)
- an insolation value coded such that the minimum value indicates deep shadow and the maximum value would occur for a surface orientation that is orthogonal to the sun (see Figure 7.3). The channel value corresponding to a flat surface for the given sun position is generally reported after processing. The resulting insolation channel can be considered as the radiance that would be measured from a surface with uniform land cover for the given sun position and atmospheric transmittance conditions.
- a slope value which represents approximate mean slope (in an unspecified direction) within the image region defined by the 5×5 filter. This channel would be linearly scaled from the minimum value, corresponding to 0°, to the maximum image value for a slope of 90° relative to a horizontal surface.
- an aspect or surface orientation value which, typically, would also be linearly coded from the minimum image at north through the mid-range value at south, or 180° from north, to the maximum value corresponding to 360°. Since a flat surface has no aspect, any pixels in a 5×5 neighbourhood with constant elevation are given the null value in this channel. Such pixels can be subsequently 'filled' if required (see Section 5.6).
- surface curvature or convexity (profile, plan, longitudinal or cross; see also Section 7.4).

Figure 7.3 Insolation values

a. Maximum insolation occurs when the surface is perpendicular to incoming radiation

b. Insolation depends on sun position and atmospheric transmittance for a flat surface

c. For surfaces parallel to or hidden from the Sun's rays, the insolation value is zero



Source: Harrison and Jupp (1990) Figure 84

Insolation is a significant parameter in environmental physics and ecology. In conjunction with remote sensing based analyses, such data allow investigations of the relationship between spectral data and surface morphology. For example, registration between an image-based land cover classification and elevation data allows quantification of the occurrence and co-occurrence of classes at different topographic locations. Tabular summaries of such comparisons may be produced automatically in most image processing systems. Alternatively, land cover categories may be stratified by topography or morphology. The derivation of slope information also allows elevation categories to be stratified by slope classes for various management purposes.

[4] The zenith angle is 90° minus the sun altitude angle.

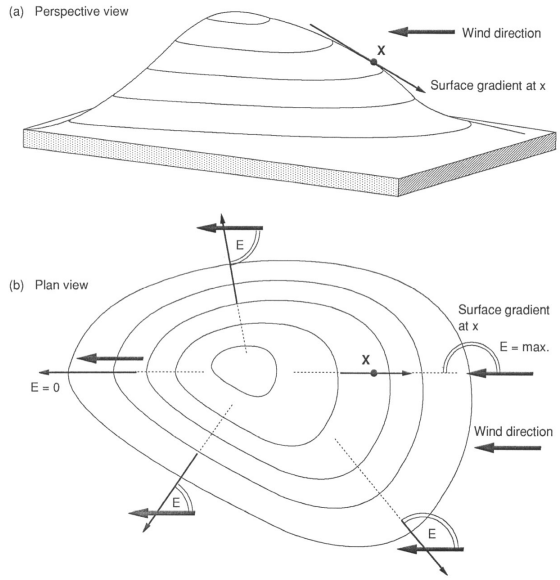
7.3 Exposure (Directional)

A special implementation of the differential filter is the exposure transformation. This transformation indicates the relative exposure for each pixel on an image channel surface to a user-defined direction as illustrated in Figure 7.4. The 'exposure' result is similar to relief shading with the 'sun position' effectively on the horizon.

Figure 7.4 Exposure transformation

a. Perspective view

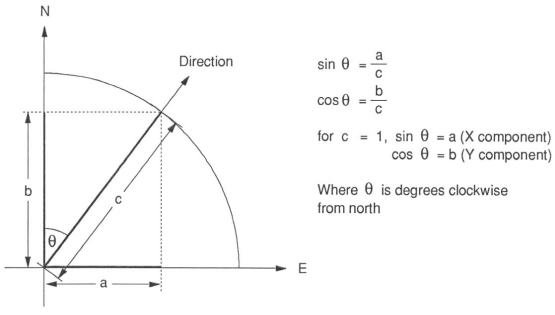
b. Plan view



Source: Harrison and Jupp (1990) Figure 81

This process is relevant to shallow water mapping applications where parameters for water depth and prevailing wind direction can be used to derive a value indicating the degree of exposure of the sea floor at each pixel to that wind direction. In shallow water mapping exercises visible blue or green channels from EO imagery can be used as a surrogate for water depth (Jupp *et al.* 1985). Directional parameters are expressed either in degrees, as a compass bearing clockwise from north, or in terms of X and Y components using polar coordinates, as illustrated in Figure 7.5.

Figure 7.5 Representing compass direction as X and Y components



Source: Harrison and Jupp (1990) Figure 82

The exposure transformation uses two smoothing differential filters over the neighbourhood of each pixel. The weights of these filters are:

-1	0	1

1	4	1

f_x

-4	0	4
-1	0	1

f_y

0	0	0
-1	-4	-1

with both filters having a divisor of 12. The exposure at each pixel is then derived from the equation:

$$E = -\left(\sin\theta \times f_x + \cos\theta \times f_y\right)$$

where

$\sin\theta$ represents the X component of the compass direction

f_x is the first partial derivative in the X direction

$\cos\theta$ represents the Y component of the compass direction

f_y is the first partial derivative in the Y direction

f is the image data, such as water depth (assumed to be oriented to north—see Section 4.2).

(Note the form of this equation differs from the gradient direction equation given in Section 7.1. This difference is due to the way the compass direction is defined. Here the direction is given as clockwise from north whereas in the context of relief shading the direction is effectively measured as anti-clockwise from east.)

A non-linear scaling of the resulting exposure value E may used to increase contrast in low image values, in which case, the final exposure image values are derived from the product of the scaled exposure component and the input image value at each pixel. For best results, images should be destriped and despiked before computing the exposure transformation as differential filters will enhance image noise such as striping patterns or spikes (see Sections 4.4 and 5.5). Some algorithms incorporate a dark value to indicate the data level corresponding to atmospheric noise (see Section 10.2 for more details about dark value selection). An expected maximum value may also be requested for rescaling purposes. This is best determined iteratively, but the 99% maximum is usually satisfactory. The resulting exposure channel may be displayed as a grey-scale and contrast-enhanced, or used to modify the brightness values of a colour image to indicate surface relief (see Section 9.3).

This transformation has proved to be valuable for mapping sub-surface morphology in coral reef studies as illustrated in Figure 7.6. In the exposure channel, bright areas indicate a higher local slope and aspect to the selected direction, and in the case of prevailing wind direction these areas indicate greater exposure. Darker areas infer sheltered positions relative to that direction. Thus, by using the relevant range of wind directions, this transformation allows a set of 'impact images' to be iteratively compiled for a study area.

Figure 7.6 Interpolation and directional (exposure) filtering

a. Transect samples of water depth over John Brewer Reef formed a sparse grid of points when registered with Landsat MSS imagery. Null pixels are shown as black.

b. Interpolation filtering can be used to create a continuous depth image.

c. Exposure (directional) filtering of the interpolated depth channel highlights reef morphology and exposure to the prevailing wind direction (southeast).

d. Landsat MSS4 (visible green) for John Brewer Reef needs to be despiked and smoothed to produce image (e) before directional filtering to give (f). The exposure image derived from EO data offers an economical approximation to (c).

May need to replace with more recent data

a. Real depth data

<image>

b. Interpolated data

<image>

c. Exposure on real depth

<image>

d. Landsat MSS4 raw data

<image>

e. Pre-processed MSS4

<image>

f. Exposure on MSS4

<image>

Source: Harrison and Jupp (1990) Plate 6

The ability to predict the exposure of sub-surface relief to prevailing weather conditions provides valuable information to coastal zone managers in formulating priorities for different sites. Spatial variations in the distribution of plant and animal species are usually strongly influenced by the wind and wave energy regime operative in the littoral and sub-littoral zones. Similarly, areas most prone to storm damage can be identified to allow the subsequent uses of such areas to be adjusted.

As a general directional filter, the exposure transformation can be applied to topographic surfaces, magnetics and gravity data or any data where a directional spatial pattern can be observed. The resulting channel is similar to sun shading (insolation) (see Section 7.1) but does not produce any 'black' shadow areas on slopes away from the 'sun'. The exposure transformation has also been used to enhance night-time thermal imagery for structural interpretation.

7.4 Surface Curvature (Surface Shape)

The use of differential calculus to quantify the changes in curvature of a two-dimensional surface, such as a DEM channel, was introduced in Section 6.1 with the first derivative representing the rate of change, or slope, of the surface, and the second derivative representing the change in slope or curvature. In two-dimensional data, these changes are expressed as partial derivatives, that is, two derivative values are computed separately indicating changes in the two dimensions. These second order statistics are invariant to rotation and slope, and describe intrinsic curvature properties of the data surface (Smith, 1983). These mathematical analyses may be applied to image data, using filtering techniques with appropriately defined weights, to determine the curvature in both across line (X) and down column (Y) directions for each pixel in the image.

As detailed in Section 6.1, four second-order partial derivatives can be computed for two-dimensional data to quantify the changes in surface curvature:

$$f_{xx} = \frac{\partial}{\partial X(f_x)} = \frac{\partial}{\partial X\left(\frac{\partial f}{\partial X}\right)} = \frac{\partial^2 f}{\partial X^2}$$

$$f_{xy} = \frac{\partial}{\partial Y(f_x)} = \frac{\partial}{\partial X\left(\frac{\partial f}{\partial Y}\right)} = \frac{\partial^2 f}{\partial X \partial Y}$$

$$f_{yx} = \frac{\partial}{\partial X(f_y)} = \frac{\partial}{\partial Y\left(\frac{\partial f}{\partial X}\right)} = \frac{\partial^2 f}{\partial Y \partial X}$$

$$f_{yy} = \frac{\partial}{\partial Y(f_y)} = \frac{\partial}{\partial Y\left(\frac{\partial f}{\partial Y}\right)} = \frac{\partial^2 f}{\partial Y^2}$$

These derivatives can be treated as a matrix to determine the direction and magnitude of maximum curvature within the region. When such results are derived from elevation data, they indicate the relative size and orientation of the major landform features within an image filter region.

Sections 2 and 8 describe some matrix algebra techniques which may be used to convert the 2×2 matrix of second partial derivatives of:

$$\begin{bmatrix} f_{xx} & f_{xy} \\ f_{yx} & f_{yy} \end{bmatrix}$$

to the form:

$$\begin{bmatrix} k_i & 0 \\ 0 & k_2 \end{bmatrix}$$

where

- is called the first eigenvalue and indicates the magnitude of a special vector in the direction of maximum data variation, called the first eigenvector, and
- is the second eigenvalue, which indicates the magnitude of the second eigenvector, this being defined as orthogonal to the first eigenvector.

In terms of elevation data, the first eigenvector of the second partial derivative matrix is aligned with the direction of curvature of the major landform feature (within a filter region at an image pixel), while the corresponding eigenvalue indicates the relative magnitude of curvature of the feature. The second eigenvalue then indicates the extent of curvature in the direction at 90° to the first eigenvalue.

Such results can then be used to categorise different landform features in elevation data, or in fact the surface curvature of any two-dimensional data such as an image channel. In terms of image processing then, the curvature values of each pixel describe the surface changes within the image sub-area defined by the filter size used to derive them.

It is appropriate to compute the surface curvature of an image channel using a 7×7 filter region (see Volume 2X—Appendix 5 for recommended filter weights). As with insolation (see Section 7.3), the input image geometry needs to be specified in terms of pixel width (X), pixel depth (Y) and the height scaling used to convert actual height to image values (Z). The Z value can also be modified to improve contrast in an output image.

Using appropriate filters, the following outputs could be computed as new image channels:

- the first eigenvalue of the second derivative matrix (based on 7×7 filter region). This is also referred to as λ_1 and indicates the extent of surface shape in the direction of maximum shape change;
- the second eigenvalue of the second derivative matrix (also referred to as λ_2). This is the extent of the surface shape change in the direction that is orthogonal to the maximum change direction in the 7×7 filter region;
- the Laplacian: (λ_1 + λ_2) / 2—this indicates the average extent or magnitude of the surface curvature within the filter region; and
- the Discriminant: (λ_1 – λ_2) / 2—summarises the directionality of surface curvature within the filter region (eg. ridge or gully).

The scaling of these values depends on the ranges specified for the expected minimum and maximum values. As with many transformations, the actual range of transformed values that will be relevant to an image can not be predicted so these range values are best determined iteratively.

When applied to elevation data these four values clearly differentiate the major landform features of ridges, gullies, hills, basins, saddles and flat ground within each 7×7 pixel region. The relationship between these features and the output image values are given in Table 7.1.

Table 7.1 Curvature image categories

Surface Curvature Category	Curvature Image Values			
	λ1	λ2	Laplacian	Discriminant
Flat ground	~0	~0	~0	small
Basin (bowl shape)	>>0	>>0	>>0	small
Hill (cap shape)	<<0	<<0	<<0	small
Saddle	>>0	<<0	~0	large
Gully line	>>0	~0	>>0	large
Ridge line	~0	<<0	<<0	large

The output channels could be used to form a colour composite. If λ_1 is displayed as blue, Laplacian as green and Discriminant as red, ridges would show as well-connected red lines, gullies as pink lines, undulating ground as aqua and flat ground as green. It should be noted that 'flat' applies to surface curvature within the filter region, not surface orientation, so that a uniformly sloping area with no change in surface shape within the filter will be described as 'flat'. A composite display of slope as blue, with the Laplacian and Discriminant as green and red respectively, provides enhanced contrast in the undulating ground. The Laplacian statistic is also referred to as the second vertical derivative in geophysics.

Li *et al.* (1986) discuss the application of these statistics to insolation imagery, which may be derived from either elevation data or EO imagery. Various methods have been proposed to isolate topographic shading (that is, effectively insolation) information from EO imagery (Ahmad 1987). Where land cover variation is relatively small, topographic shading is the major factor causing changes in image brightness and can be reasonably estimated as the first principal component (see Sections 2.2 and 9). As with other analyses, insolation imagery that relates to a high sun angle provides better discrimination of the landform features than is possible with low sun angle imagery.

While the results of curvature analysis on insolation imagery will necessarily differ from those of elevation data, some interesting landform features are highlighted. At the very least, these results can be used to stratify the image into 'flat' and non-flat areas. Such a stratification then allows the analysis of land use and land cover features in the flat areas without influencing class definition by topographic brightness variations.

7.5 Further Information

Texture

Jensen (2016) Section 8

Jensen, J.R. (2016). *Introductory Digital Image Processing: A Remote Sensing Perspective.* 4th edn. Pearson Education, Inc. ISBN 978-0-13-405816-0

7.6 References

Ahmad, W. (1987). A forestry inventory using Landsat MSS data in NE Tasmania Australia. PhD. Thesis. University of Tasmania, Hobart.

Gordon, D.K., and Philipson, W.R. (1986). A texture-enhancement procedure for separating orchard from forest in Thematic Mapper data. Int. J. Remote Sensing 7, 301–4.

Haralick, R.M. (1979). Statistical and Structural Approaches to Texture. Proc. IEEE, 67, 786–804.

Harris, J.L. Sr. (1977). Constant Variance Enhancement – A Digital Processing Technique. App. Optics 56, 569–74.

Harrison, B.A., and Jupp, D.L.B. (1990). *Introduction to Image Processing: Part TWO of the microBRIAN Resource Manual.* CSIRO, Melbourne. 256pp.

Jupp, D.L.B., Mayo, K.K., Kuchler, D.A., Van R. Classen, D., Kenchington, R.A., and Guerin, P.R. (1985). Remote sensing for planning and managing the Great Barrier Reef of Australia. Photogrammetria 40, 21–42.

Levine, M.D. (1985). Vision in Man and Machine. McGraw-Hill Inc. USA.

Li, R., Jupp, D.L.B., Harrison,B.A., and Ahmad, W. (1986). Extraction of landform information from remotely sensed data: an initial study. Proc. Beijing International Symposium on Remote Sensing, Beijing, China, 18–22 November, 67–73.

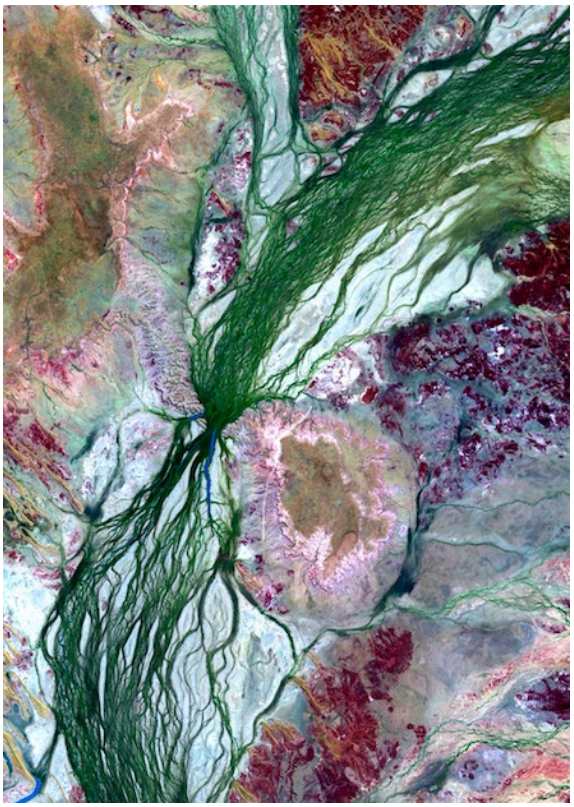
Niblack, W. (1986). An Introduction to Digital Image Processing. Prentice-Hall International. New Jersey.

Pickup, G., and Foran, B.D. (1987). The use of spectral and spatial variability to monitor cover changes on inert landscapes. Remote Sensing of Environment 23, 351–63.

Smith, G.B. (1983). Shape from shading: an assessment. Proc. NASA Symposium on Mathematical Pattern Recognition and Image Analysis, June 1–3, 1983. Johnson Space Centre, Houston, Texas, 543–76.

Woodcock, C.E., and Ryherd, S.L. (1989). Generation of texture images using adaptive windows. *Technical Papers, 55th Annual Meeting ASPRS*, 2:11–22.

MULTIPLE CHANNEL OPERATIONS



This Section is concerned with techniques that apply linear and non-linear operations to one or more image channels. Linear operations rely on matrix algebra and can implement simple rescaling or inversion of individual channels, or powerful multi-dimensional manipulation of image data (see Section 2). Non-linear operations involve multiplication and division of channel values (see Section 3).

A range of multi-channel transformations is offered by most image processing systems. Some of the most commonly available transformations involving multiple channels include:

- modifying image brightness (see Section 8);
- Principal Components Analysis (PCA; see Section 9);
- band ratios (see Section 10); and
- vegetation indices (see Section 11).

Some systems provide flexible 'band maths' options, whereby a sequence of linear and non-linear operations can be applied to selected sets of image channles. Such options offer significant flexibility, but need to be used intelligently with EO imagery to ensure that the transformed results are meaningful.

Background image on previous page: Landsat-8 OLI image of Diamantina Gates, southwestern Queensland, acquired in winter 2013. Due to constraints imposed by the adjacent Goyder and Hamilton Ranges, the Diamantina River narrows to about 1 km wide at Hunters Gorge, which is located near the centre of this image.

Source: Craig Shephard, DSITI.

8 Modifying image brightness

Background image: Astronaut photograph of Argyle Diamond Mine, Northern Territory, acquired from International Space Station on 22 June 2006 from an altitude of 339 km.

Source: NASA Photo ID ISS013-E-39697.

The affine transformation is introduced in Section 2. This multi-channel operation is commonly used to proportionally add and subtract selected channels in order to highlight specific image features. Some specific uses of affine transformation include modifying image brightness values to indicate surface relief (see Section 8.1 and removing spatial noise, such as atmospheric haze, from imagery (see Sections 8.2 and 8.3). Other methods that have been developed to highlight correlation between image channels are introduced in Section 8.4.

8.1 Relief Shading

Various image processing routines can be used to generate shaded relief imagery (see Section 7). For example, a relief channel can be integrated with the brightness of the composite colours in a colour composite image to represent changes in relief (see Figure 8.1). This process requires that the data range of the relief channel matches those channels forming the composite image. If the data range of an elevation channel is much larger than those channels forming the colour composite, the image channels must first be rescaled to the same range (usually with those parameters used to display the composite image) so that the relief shading will affect all channels equally. (Note that any channel that would normally be displayed as an inverted range (that is as a negative channel number; see Volume 2A—Section 7) in the colour composite should first be inverted using an affine transformation as described above.) The relief shading can then be simply merged into the colour image by averaging each rescaled composite channel with the relief channel. Weighting can be applied if required to this averaging process. The resulting shaded composite will have modified brightness values but the same hues as illustrated in Figure 8.1.

Figure 8.1 Modifying colour brightness using relief shading

An affine transformation can be used to replace the intensity axis of a colour composite image with some other intensity scale. In this example, an insolation channel (derived from elevation data for southeast Australia) has been used to highlight surface relief within colour-coded drainage divisions of the Murray-Darling basin. A segmentation mask has been used to restrict the effect of the transformation to within the basin. The same transformation may be used to merge multi-scale EO images (see Figure 13.3).

a. Original colour image

<image a.>

b. Shaded relief image

<image b.>

c. Colour image with shaded relief

<image c.>

Source: Harrison and Jupp (1990) Plate 17

Weighting values can be derived for this operation that replace the intensity component of each composite primary by the elevation channel value (see Volume 2X—Appendix 3.3). This would be implemented as:

$$\begin{bmatrix} \text{shaded comp 1} \\ \text{shaded comp 2} \\ \text{shaded comp 3} \end{bmatrix} = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} & -\frac{1}{3} & \frac{2}{\sqrt{3}} \\ -\frac{1}{3} & \frac{2}{3} & -\frac{1}{3} & \frac{1}{\sqrt{3}} \\ -\frac{1}{3} & -\frac{1}{3} & \frac{2}{3} & \frac{1}{\sqrt{3}} \end{bmatrix} \times \begin{bmatrix} \text{rescaled comp ch 1} \\ \text{rescaled comp ch 2} \\ \text{rescaled comp ch 3} \\ \text{shaded relief} \end{bmatrix}$$

The resulting shaded relief imagery is sometimes non-linearly scaled to enhance contrast in dark areas, so in general will not require rescaling for this operation. The same approach can be used to merge finer scale panchromatic imagery with a coarser scale colour composite (see Section 9.3) as shown in Figure 8.2.

Figure 8.2 Integrating imagery with differing spatial resolution

a. SPOT multi-spectral image of Canberra with 20 m pixel size.

b. SPOT panchromatic image with 10 m pixel size.

c. SPOTmulti-spectral image registered to panchromatic image scale.

d. Merged colour composite image combines the spatial resolution of the panchromatic image with the colours from the multi-channel data. An affine transformation was used to replace the intensity axis of the multi-spectral image with the panchromatic channel. The same transformation is illustrated in Figure 3.2 to merge shaded relief information with a spatially-registered colour image. (© CNES, 1987)

Replace with higher resolution imagery

<image a.> <image b.>

<image c.> <image d.>

Source: Harrison and Jupp (1990) Plate 18

8.2 Removing Spatial 'Noise'

Spatial 'noise' can occur in EO imagery for a number of reasons, such as atmospheric haze, specular reflection or uneven illumination (due to topography or uneven lighting as may occur during laboratory scanning of imagery). These spatial variations are not constant over the image, so we cannot correct for them by simply subtracting a constant 'noise' level from each pixel.

Methods that may be used to adjust for these spatial variations include image ratioing (see Section 10) or weighted channel subtraction. The choice of correction technique depends on the nature of the model that describes the interaction between ground reflectance and the image noise. Where the spatial noise is related to ground reflectance by a multiplicative model (that is the detected noise level is affected by target reflectance, such as with topographic shading) image ratios should be used. The models for atmospheric noise and sunglint (specular reflection from water) are linearly related to ground reflectance (that is the noise level is independent of target reflectance) so an affine transformation can be used to correct for these components in the image radiance values. That is:

$$x_{ij} = b_{ij} + d_{ij}$$

where

x_{ij} is image radiance value at pixel i in channel j

b_{ij} is the reflectance component of pixel i in channel j

d_{ij} is the atmospheric noise and/or sensor calibration offset component.

These methods use one image channel that contains the pattern as a 'reference' channel (in the case of scan-digitised imagery, this should always be specifically produced using a blank target) and require that the reference channel be 'sacrificed' in any subsequent processing. Thus the reference channel is best selected from channels that would not normally be used. Assuming some constants α_j and β_j :

$$d_{ij} = \alpha_j \times d_{ir} + \beta_j$$
$$x_{ij} - \left(\alpha_j \times d_{ir} + \beta_j\right) = b_{ij} - \alpha_j \times b_{ir}$$

where

when $j=r$, $x_{ir} = d_{ir}$ is the image radiance value at pixel i in the reference channel r

b_{ir} is the reflectance component of pixel i in the reference channel r. This reference channel is usually chosen so that this component is not significant for further processing.

In this equation, the result is free of d_{ij} and retrieves b_{ij} if $b_{ir} = 0$.

In an image to be used for water studies, for example, a haze pattern could be reduced by using an infrared (especially middle infrared) channel as the reference channel, provided it contains the same haze pattern. The near infrared and middle infrared wavelengths are virtually totally absorbed by water so would not be required for subsequent processing. Where several channels are available in this wavelength region, the first principal component (see Section 9.3) of those channels could be used.

The relationship between each image channel and the reference channel can be determined by image crossplotting (see Volume 2A—Section 8.1.3). As illustrated in Figure 8.3, the regression line for each crossplot can then be computed as:

$$\text{channel } i = a_i \times \text{reference channel} + b_i$$

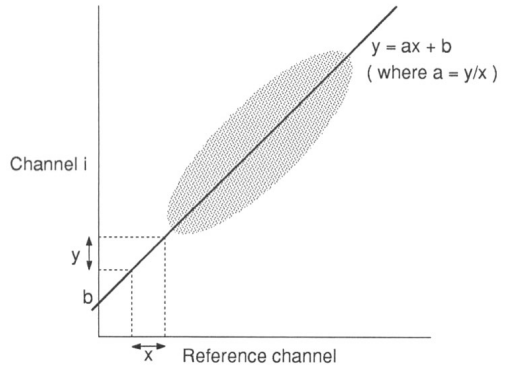
where

a_i is the regression coefficient

b_i is the regression offset.

Figure 8.3 Using channel crossplot to derive regression line with reference data

The regression equation summarises the relationship between the two channels.



Source: Harrison and Jupp (1990) Figure 86

The proportion of our image signal in channel i which is modelled as the spatial 'noise' can then be subtracted by the equation:

$$\text{channel } i' = \text{channel } i - a_i \times \text{reference channel}$$

To ensure the scaling of channel i' is consistent with channel *i* (and they have the same mean values), we usually add an offset of:

$a_i \times \underline{R}$

where

R is the mean of the reference channel. In this case the expected output range should be 0–254.

In an affine transformation of a four channel image this model would be specified as shown in Figure 8.4.

Figure 8.4 Filter to remove image noise

$$\begin{bmatrix} \text{corrected ch 1} \\ \text{corrected ch 2} \\ \text{corrected ch 3} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -a_1 \\ 0 & 1 & 0 & -a_2 \\ 0 & 0 & 1 & -a_3 \end{bmatrix} \times \begin{bmatrix} \text{input ch 1} \\ \text{input ch 2} \\ \text{input ch 3} \\ \text{reference ch} \end{bmatrix} + \begin{bmatrix} a_1 \times R \\ a_2 \times R \\ a_3 \times R \end{bmatrix}$$

The reference channel may need preprocessing for some image sources. For example, in scan-digitised imagery the blank target scan can have considerable local variation, so may need to be heavily smoothed (with a 5×5 or 7×7 average filter with all weights equal to 1) before being used for illumination correction. Other sources may require calibration to radiance units. To assess the impact of this correction, the transformed and original image pairs can be differenced (see Volume 2D—Section 1.1).

8.3 Removing 'Limb Brightening'

'Limb brightening' across an airborne scanner image can also be corrected using an affine transformation. In this case the wide scan angle causes edge pixels to be imaged through a significantly longer atmospheric path distance, which allows greater atmospheric scattering and attenuation and results in a characteristic increase in brightness towards the edges of the image. This brightness trend across the image can be estimated from the average brightness down each pixel column. The average brightness line may be represented in an image by duplicating the line to match the number of lines in the original image (see Volume 2A—Section 7.2.1.4) and subtracted from the original image (with coefficients of 1 and an offset equal to the mean of the original image) to produce a brightness corrected image (see Figure 8.5).

Figure 8.5 Filter to remove limb brightening

$$\begin{bmatrix} \text{corrected ch 1} \\ \text{corrected ch 2} \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} \text{ch 1} \\ \text{mean ch 1 intensity} \\ \text{ch 2} \\ \text{mean ch 2 intensity} \end{bmatrix} + \begin{bmatrix} \text{ch 1 mean} \\ \text{ch 2 mean} \end{bmatrix}$$

To correct all channels in an image, this operation would be performed more efficiently using image differencing (see Volume 2D—Section 1.1). The statistics associated with along-line brightness variation within an image can be accumulated in some image processing systems. For example, the mean intensity for each channel could be extracted to enable the correction described above.

8.4 Other Affine Transformations

A range of affine transformations have been proposed for multivariate EO imagery to variously highlight data or noise components for particular purposes. These methods analyse the spectral and/or spatial correlation of one or more images and are particularly relevant to change detection studies (see Volume 2D). Examples of such transformations include:

- Maximum Autocorrelation Factor (MAF)—derives new, orthogonal channels in which neighbouring pixels have maximum autocorrelation to highlight contiguous regions (Switzer and Green, 1984; Conradsen *et al.*, 1985; Nielsen *et al.*, 1998; Nielson, 2011). Autocorrelation statistics are derived from analysis of the dispersion matrix of the difference between each image channel and its spatially shifted self;
- Minimum/Maximum Noise Fraction (MNF)—orders transformed image components in terms of image quality (Green *et al.*, 1988; Nielsen, 2011); and
- Multivariate Alternation Detection (MAD)—analyses canonical correlations between pairs of image channels to highlight differences (Nielsen *et al.*, 1988; see Volume 2D—Section 2). Nielsen (2007) describes an extension to this transformation, termed the Iteratively Reweighted MAD method (IR-MAD).

Factor analysis methods are further discussed in Volume 2E.

8.5 Further Information

Sawyer (1955)

8.6 References

Conradsen K., Ersboll B. K., Thyrsted T. (1985). A comparison of min/max autocorrelation factor analysis and ordinary factor analysis. Nordic Symposium in Applied Statistics, Lyngby. 47-56.

Green, A.A., Berman, M., Switzer, P., and Craig, M.D. (1988). A transformation for ordering multispectral data in terms of image quality with implications for noise removal. IEEE Trans. Geosci. and Remote Sensing 26(1), 65–74.

Harrison, B.A., and Jupp, D.L.B. (1990). *Introduction to Image Processing: Part TWO of the microBRIAN Resource Manual.* CSIRO, Melbourne. 256pp.

Nielsen, A.A., Conradsen, K. and Simpson, J.J. (1998): Multivariate alteration detection (MAD) and MAF post-processing in multispectral, bi-temporal image data: new approaches to change detection studies. Remote Sensing of Environment, 64, pp. 1-19.

Nielson, A.A. (2007). The Regularized Iteratively Reweighted MAD Method for Change Detection in Multi- and Hyperspectral Data. IEEE Trans. on Image Proc. 16(2), 463–478.

Nielson, A.A. (2011). Kernel Maximum Autocorrelation Factor and Minimum Noise Fraction Transformations. IEEE Trans. Image Proc. 20(3), 612–624.

Sawyer, W.W. (1955). *Prelude to Mathematics.* Penguin Books Ltd. London.

Switzer, P., Kowalik, W.S., and Lyon, R.J.P. (1981). Estimation of atmospheric path-radiance by the covariance matrix method. Photogramm. Eng. Remote Sensing 47, 1469–76.

Switzer, P., and Green, A.A. (1984). Min/max autocorrelation factors for multivariate spatial imagery. Tech. Report 6, April 1984, Dept. Statistics, Stanford.

9 Principal Components Analysis

Most EO images contain some redundant information. For example, Landsat TM2 and TM3 contain similar data values since most ground cover types give similar responses in these two channels (that is, they are highly correlated). When redundancies do occur in an image, it is possible to 'remove' the duplicate information and represent the data in fewer channels. In most Landsat TM imagery, for example, over 95% of the data variation in the original six optical channels can be represented by two principal component channels (ref). This data reduction operation is particularly relevant to geological applications where feature identification relies on visual interpretation of enhanced imagery.

The Principal Components Analysis (PCA) uses spectral statistics of the image to define a rotation of the original image dimensions such that the data are arranged along axes of decreasing variance (see Section 2.2 for details and Volume 2X—Appendix 6 for the mathematical basis of this transformation). The major direction of data variation in multi-channel EO imagery rarely lies along one original channel axis. Similarly the variance gradient for specific image features is usually defined in at least two data dimensions. These statistical characteristics of EO data make it difficult to neatly analyse the component features or divide up the image brightness variations as channel themes (see Volume 2A—Section 9.1). By aligning the data axes with the variance of the whole image dataset, or just with a specific feature in it, we can access the changes within the major variance using a single channel.

The transformation parameters for PCA are derived from the covariance or correlation matrix of an image (or sub-image; see Section 2.2). This transformation is particularly useful for images with many similar channels, such as hyperspectral imagery. The resulting Principal Components (PCs) are totally uncorrelated and can be computed by adding proportions of the original image channel values.

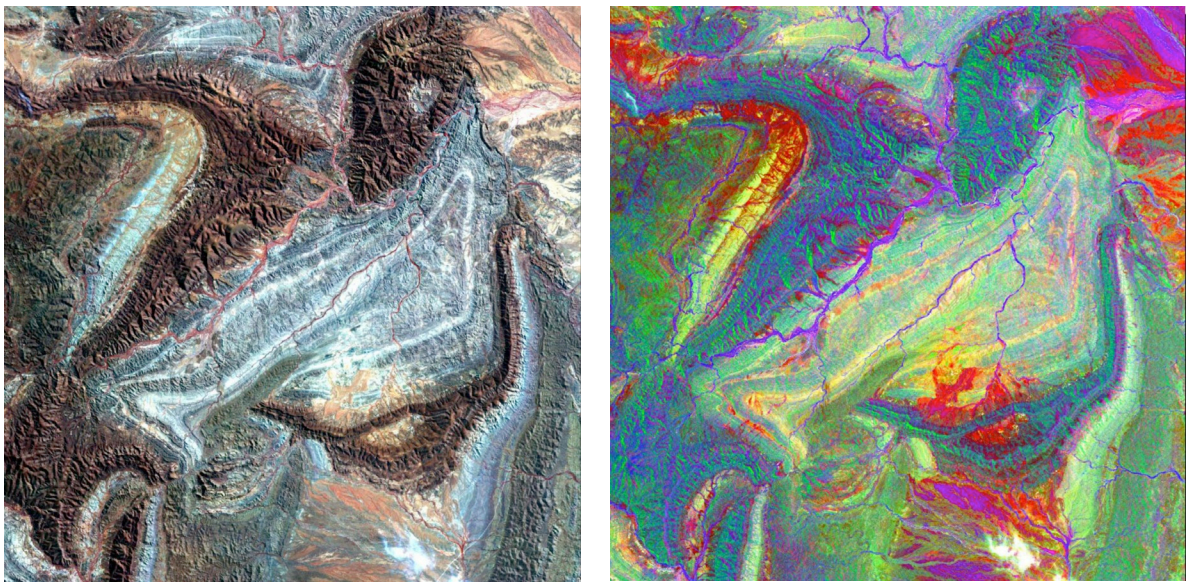
Imagery produced by PCA typically has much more saturated colours than standard colour composites as shown in Figure 9.1. This is due to the decorrelation between principal component channels. In terms of the RGB colour cube described in Volume 2A— 5, EO imagery generally displays little variation about the intensity axis with no potential to form the subtractive primary colours at the two-dimensional corners of the cube (see Figure 9.2). A colour composite of three principal component (PC) channels however, effectively fills the volume of this cube so that both additive and subtractive primaries are visible in the displayed imagery (see Excursus 9.1).

Figure 9.1 Principal Component transformation

Landsat TM image of Flinders Ranges, South Australia

a. Original false colour composite using bands 2, 3 and 4 as blue, green and red

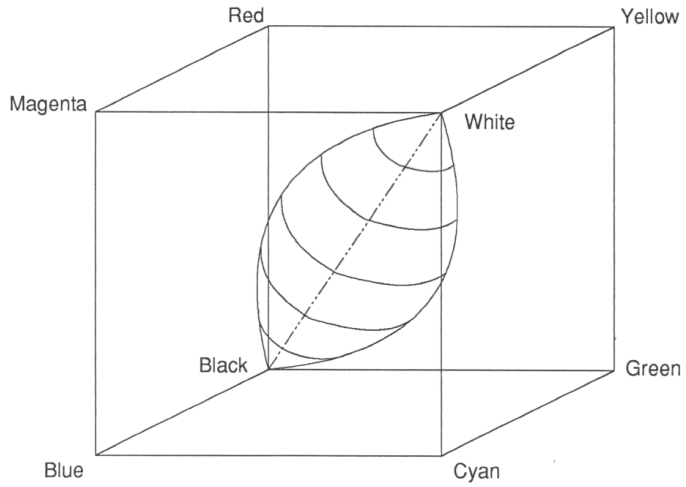
b. Principal component image using PCs 1, 2 and 3 highlights lithology, soils and vegetation



Source: Megan Lewis, University of Adelaide

Figure 9.2 RGB colour cube showing original image data range

The correlation between channels in EO imagery generally means that the subtractive primary colours do not occur in colour composite images.



Source: Harrison and Jupp (1990) Figure 90

Excursus 9.1 PCA Example

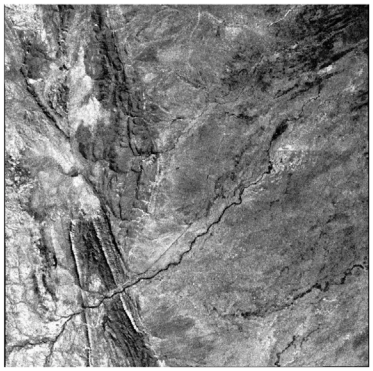
Source: Megan Lewis, University of Adelaide

An example of PCA transformation on Landsat TM imagery is illustrated in Figure 9.3 and Figure 9.4. The original TM bands (Figure 9.3a) are strongly correlated (Figure 9.3b) with little spectral contrast between land surfaces. The principal components transformation (Figure 9.4b) enables a new set of channels to be computed which isolate sensor noise to one component and maximises the spectral information in the higher components (Figure 9.4a). The original false colour composite (Figure 9.4c) contrasts with the colour composite from PCs 1, 2 and 3 (Figure 9.4c).

Figure 9.3 Principal Component transformation example—before

a. Original Landsat TM channels for ???
b. Correlation Matrix
c. Original false colour composite

Source: Megan Lewis

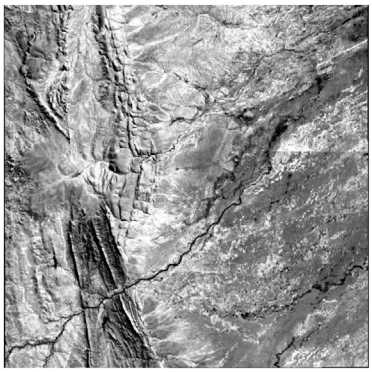


Band 1 - blue

Band 2 - green

Band 3 - red

Band 4 - near infra-red



Band 5 - near infra-red

Band 6 - mid infra-red

Band	Band 1	Band 2	Band 3	Band 4	Band 5	Band 6
1	1	0.788	0.65	0.546	0.554	0.531
2	0.788	1	0.902	0.803	0.775	0.773
3	0.65	0.902	1	0.902	0.873	0.879
4	0.546	0.803	0.902	1	0.836	0.821
5	0.554	0.775	0.873	0.836	1	0.928
6	0.531	0.77	0.879	0.821	0.928	1

88% of variance in band 3
explained by band 6 values

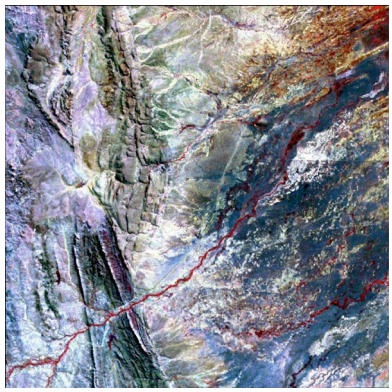
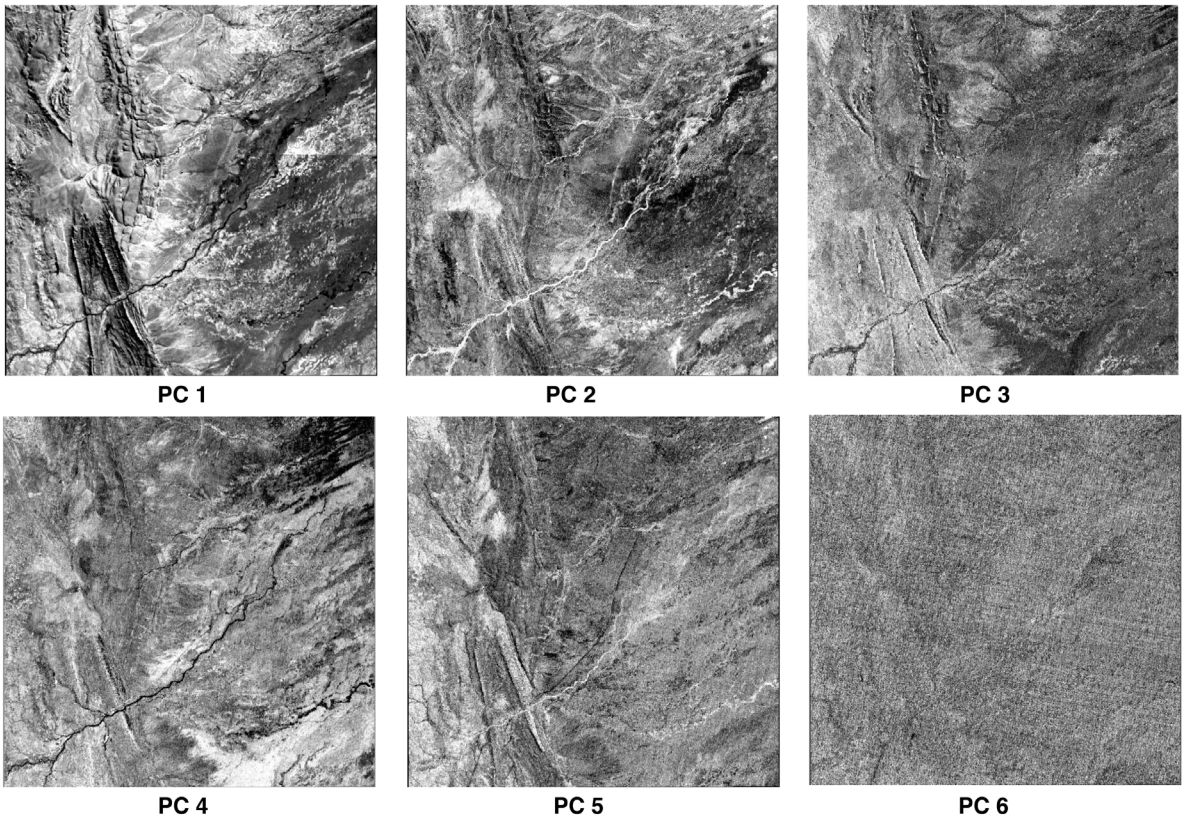


Figure 9.4 Principal Component transformation example—after

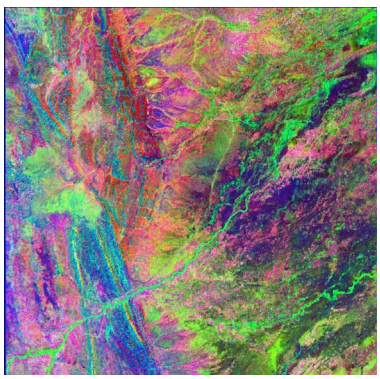
a. Principal Component channels for Figure 13.7
b. Principal Components transformation matrix
c. and colour composite from PCs 1, 2, 3

Source: Megan Lewis



PC	Band 1	Band 2	Band3	Band 4	Band 5	Band 6
1	0.058	0.059	0.045	0.375	0.593	0.706
2	0.153	0.154	0.137	0.534	0.432	-0.68
3	0.336	0.271	0.243	0.54	-0.652	0.196
4	0.453	0.758	-0.131	-0.426	0.15	0.008
5	0.603	-0.408	-0.675	0.115	-0.021	-0.016
6	0.54	-0.398	0.669	-0.298	0.117	0.006

PC1 = 0.058**B1**+ 0.059**B2** + 0.045**B3** + 0.375**B4** + 0.593**B5** + 0.706**B6**



9.1 Data Reduction

PCA is principally used for data reduction although frequently this is an intermediate step in other processes such as image classification, enhancement or integration. By representing the image data in fewer channels (that is, reducing the data dimensionality), various operations may be applied to the transformed channels before transforming back to the original data space. Such data reduction techniques are particularly relevant to hyperspectral imagery. Analysis of 128 channel HyMap airborne imagery showed that:

- 99% of the image variation is contained in PCs 1 to 4;
- PCs 5 to 50 mostly contain local variation information and anomalies; and
- PCs 50 to 128 mostly contain instrument noise (Megan Lewis, *pers. comm.*)

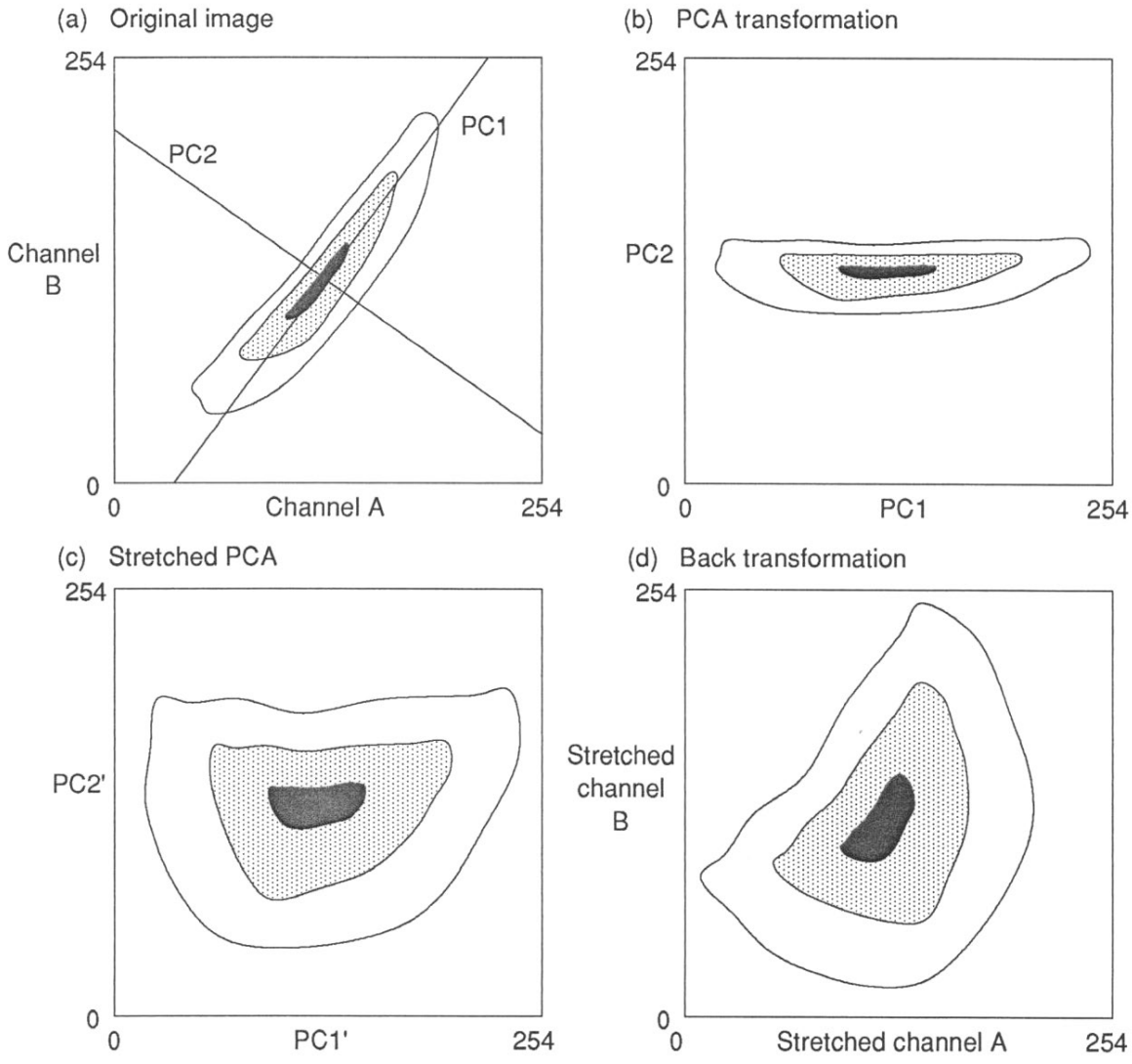
In image classification, PC1 and PC2 provide a useful basis for autoseeding methods based on defining channel themes (see Volume 2A— Section 9.1). Volume 2E describes this procedure in more detail. Once the image has been classified using a set of basic classes defined this way, they may be transformed back to the original image data channels using the 'spectral transfer' operation described in Volume 2D—Section 1.3.

9.2 Decorrelation Stretching

Decorrelation stretching provides a useful visual enhancement for increasing colour saturation in composite imagery (Gillespie *et al*. 1986). This process involves transforming image channels to the principal components space, stretching the image data along the PC axes to take on a spherical distribution, then transforming the data back onto the original channel axes. Rothery (1987) demonstrates the value of this enhancement for visual interpretation of Landsat Thematic Mapper imagery in a geological application. The results of this process are shown in Figure 13.5. The back transformation can also be applied with one component removed. This is useful to remove certain spatial noise patterns from imagery where the noise can be isolated as one component.

Figure 13.5 Decorrelation stretching

Images can be transformed to the principal component space for contrast stretching then back transformed to the original image channels. By rescaling the data in an uncorrelated data space, the resulting composite image has enhanced colour contrast. (This example has been scaled for a byte format image.) [Source: Gillespie *et al.* 1986]



Source: Harrison and Jupp (1990) Figure 91

9.3 Multi-scale Intensity Enhancement

Panchromatic imagery with finer spatial resolution than a multi-spectral image can be integrated with the multi-channel data to visually enhance its spatial detail. Cliche *et al.* (1985) and Price (1987a) suggest methods for integrating the two image resolutions. For example, high spatial resolution panchromatic imagery may be merged with medium resolution image data to produce colour composites at the scale of the panchromatic image. The HSI transformation has been used to merge such datasets by replacing the intensity channel with the (registered) panchromatic data then transforming back to the RGB space. This operation could be implemented using the same affine transformation matrix described in Section 2.1.4 for incorporating relief shading with a colour composite image (see Figures 3.2 and 3.3).

PCA can also be used for this purpose. This operation requires that the images are well registered at the scale of the panchromatic image (that is, the multi-spectral image needs to be resampled to the pixel size of the panchromatic image—see Volume 2A—Section 6 and Volume 2B). Those channels in the resampled multi-spectral image that are to be used for the colour composite are selected and transformed by PCA. The panchromatic channel can be substituted for the first principal component (see Volume 2D—Section 1.6) since PC1 represents the brightness or intensity variations in EO data. The panchromatic channel should first be rescaled to match PC1 using the regression technique discussed in Section 3.1.6 and Volume 2D—Section 2. The resulting image is then back-transformed to the original multi-spectral image channels to produce an image in which the spatial detail of the panchromatic channel is supplemented by the colours of the coarser composite image. The back transformation matrix will need to be adjusted to account for the scaling between the actual PC ranges and their representation in the PC image. This matrix would be implemented as an affine transformation (see Section 2.1).

This approach offers a very flexible method of merging new information with any spatial dataset, since the new 'vector' (or channel) can be used to replace any existing vector in the original data. However, for replacing the intensity component of a colour image, the direct method illustrated in Figure 3.2 and Figure 3.3 is most convenient (Shettigara, 1989).

Suits *et al.* (1988) describe procedures involving PCA and multivariate regression for using signals from one sensor as a substitute for signals in another. Such methods allow monitoring operations to transfer spectral interpretation techniques from one sensor to another if required. Other approaches to data fusion of EO imagery from different sensors include STARFM (Spatial and Temporal Adaptive Reflectance Fusion Model; Gao *et al.*, 2006) and STRUM (Spatial and Temporal Reflectance Unmixing Model; Gevaert and Garcia-Haro, 2014).

9.4 Directed PCA

Fraser and Green (1987) report an interesting technique called 'directed PCA' (DPCA) which substantially reduces the effects of vegetation in woodland environments (50–70% vegetation cover) for geological analyses. This technique involves calculating principal components on two input ratio channels. These ratio channels are selected so that one is a geological discriminant that is often confused with vegetation (such as short wave infrared / middle infrared, eg. 1.65 μm/2.22 μm) and another which is primarily a vegetation index (such as near infrared / red). PCA of these two channels then produces two uncorrelated data axes: the first aligned with vegetation and second with non-vegetation—in this case, clay.

9.5 Change Detection

PCA is also frequently used as a change detection method with multi-temporal imagery, by either transforming two or more images separately then selecting appropriate PCs for comparison, or registering the multiple images and transforming them as a merged dataset (see Volume 2D—Section 2.3.2). Many image processing systems offer options for adjusting channel variance in both the definition and application of the transformation matrix and also to back transform from the PCA space to the original data channels. The use of PCA for change detection studies is detailed in Volume 2E.

9.6 Further Information

Jensen (2016) Section 8

Gonzalez and Woods (2018) Section 12

9.7 References

Cliche, G., Bonn, F., and Teillet, P. (1985). Integration of the SPOT panchromatic channel into its multispectral mode for image sharpness enhancement. Photogramm. Eng. and Remote Sensing 51, 311–6.

Fraser, S.J., and Green, A.A. (1987). A software defoliant for geological analysis of band ratios. Int. J. Remote Sensing 8, 525–32.

Gao, F., Masek, J., Schwaller, M., & Hall, F. (2006). On the blending of the Landsat and MODIS surface reflectance: Predicting daily Landsat surface reflectance. IEEE Transactions on Geoscience and Remote Sensing, 44(8), 2207–2218, http://dx.doi.org/ 10.1109/TGRS.2006.872081

Gevaert, C.M., Carcía-Haro, F.J. (2015). A comparison of STARFM and an unmixing-based algorithm for Landsat and MODIS data fusion. Remote Sensing of Environment 156, 34–44. http://dx.doi.org/10.1016/j.rse.2014.09.012

Gillespie, A.R., Kahle, A.B., and Walker, R.E. (1986). Colour enhancement of highly correlated images. I. Decorrelation and HSI contrast stretches. Remote Sensing of Environment 20, 209–35.

Gillespie, A.R., Kahle, A.B., and Walker, R.E. (1987). Colour enhancement of highly correlated images. II. Channel ratio and chromaticity transformation techniques. Remote Sensing of Environment 22, 343–65.

Gonzalez, R.C., and Woods, R.E. (2018) *Digital Image Processing*. Pearson Educational Inc., New York.

Harrison, B.A., and Jupp, D.L.B. (1990). *Introduction to Image Processing: Part TWO of the microBRIAN Resource Manual.* CSIRO, Melbourne. 256pp.

Jensen, J.R. (2016). *Introductory Digital Image Processing: A Remote Sensing Perspective.* 4th edn. Pearson Education, Inc. ISBN 978-0-13-405816-0

Price, J.C. (1987a). Combining Panchromatic and Multispectral Imagery from Dual Resolution Satellite Instruments. Remote Sensing of Environment 21, 119–28.

Price, J.C. (1987b). Calibration of satellite radiometers and the comparison of vegetation indices. Remote Sensing of Environment 21, 15–27.

Rothery, D.A. (1987). Decorrelation stretching as an aid to image interpretation. Int. J. Remote Sensing 8, 1253–4.

Shettigara, V.K. (1989). A linear transformation technique for spatial enhancement of multi-spectral images using higher resolution data. Proc. IGARSS '89, Vancouver, Canada, pp 2615–8.

Suits, G., Malila, W., and Weller, T. (1988). Procedures for using signals from one sensor as substitutes for signals of another. Remote Sensing of Environment 25, 395–408.

10 Channel Ratios

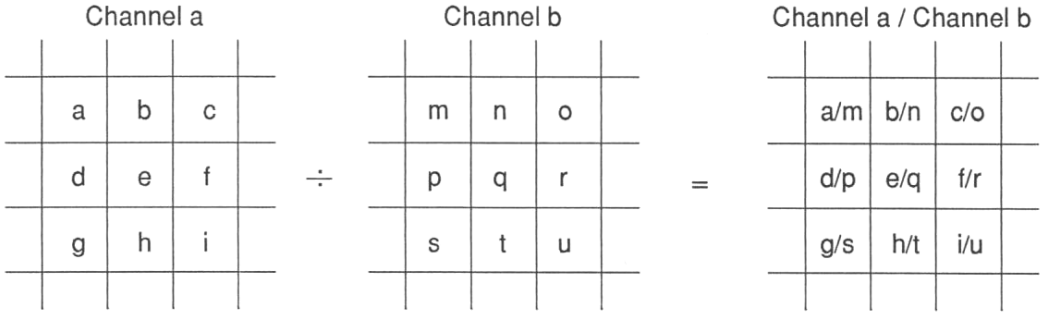
A simple but effective transformation that is commonly used with EO imagery involves computing the ratio values between two image channels. The simplest implementation of this transformation involves dividing the value of a pixel in one channel by its value in a second channel and rescaling the result to the available image data range for all pixels in the image (see Section 3). The operation of this method is illustrated in Figure 10.1.

Figure 10.1 Operation of simple channel ratio

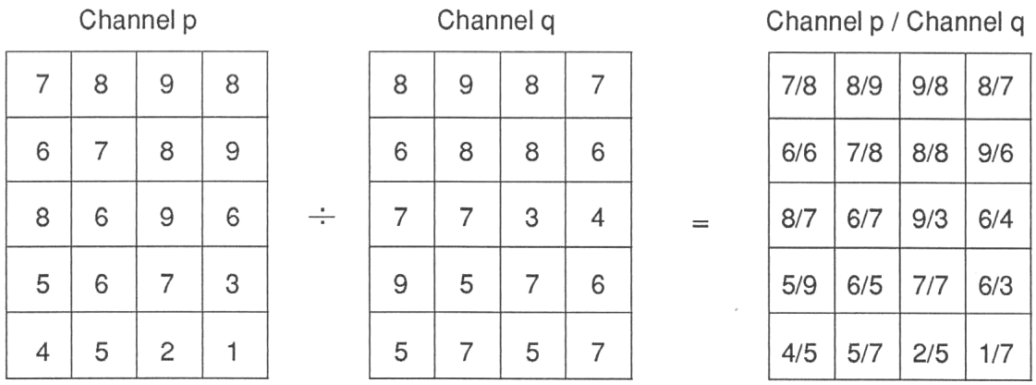
a. Process

b. Numeric Example. The resulting ratio values are rescaled to integer values in the output image.

(a) Operation



(b) Numeric example



Rescale minimum ratio value (1/7) → 0
maximum ratio value (9/3) → 100

So output value = (ratio - 1/7) x 100 / (9/3 - 1/7) =

Output channel

26	20	34	35
30	26	30	36
35	25	100	48
14	37	30	13
23	20	9	0

Source: Harrison and Jupp (1990) Figure 93

The result of channel ratioing is generally rescaled for recording in the image data range, often based on user-supplied scaling factors. These parameters should be known if the ratio channel is to be subsequently used with models relating ground measurements to image radiance. The scaling factors should relate to the minimum and maximum ratio values in the image to ensure that the resulting values are fully stretched over the available image data range. These factors are usually determined iteratively.

A wide distribution, but low frequency, of image values near the minimum and maximum values in a channel can sometimes make this rescaling operation difficult. If the reported actual minimum and maximum ratio values are used for scaling, they may not provide good overall contrast in the image. As with image display, this problem is data-dependent and can be resolved by reference to the ratio image histograms.

When a ratio channel has been computed to represent a larger range of ratios than is necessary for most pixels in the image, a suitable ratio range can be selected using the channel histogram and rescaled back to the actual ratio values for use in another processing run. This rescaling will require the expected ratio range (*a*,*b*) which was used to compute the ratio channel. Thus to rescale value P in the ratio channel to an actual ratio value we would compute:

$$\frac{P}{\text{image data range}} \times (b-a) + a$$

$$P \;/\; (\text{image data range}) \times (b-a) + a$$

where

a is the minimum ratio value used to rescale the ratio channel; and
b is the maximum ratio value used to rescale the ratio channel.

Channel ratios may be computed in a variety of ways. For example, a simple ratio may be computed between pairs of channels as shown in Figure 10.1. Alternatively a 'smoothed ratio' may be computed as detailed in Section 3.3. This produces a more 'stable' result by removing small variations in image values that would be accentuated by the ratio process. Ratios of a pair of image channels may also be computed by differencing the logarithms of the channels (see Section 3.2). Some commonly encountered ratios are introduced in Excursus 10.1.

When the channels in an image are not scaled consistently (due to variations in gains and/or calibration) the expected relationships between channels may not be observed in ratio results. Models to account for these variations are introduced in Section 10.1. Also, ratio variables are characterised by having a valid zero point (see Volume 1A—Section 2). When ratioing a pair of channels in EO imagery, computations need to account for the true minimum value in each channel. This minimum value is commonly called the 'dark' value (see Section 10.2) as it effectively accounts for sensor 'dark current' (see Volume 1A—Section 13). Finally, a modified approach to computing channel ratios, called 'Directed band ratioing', is introduced in Section 10.3, which enables the ratioed values to retain intensity information.

Excursus 10.1 Commonly Used Ratios

While the ratioing of image channels is a very simple operation computationally, it is a very effective way of highlighting the differences between two channels. Ratios of various channels are used to highlight differences in particular features of the Earth's surface. Some of the commonly used channel ratios include:

- green / red: soil colour or water colour;
- green / blue: water colour;
- near infrared / red: vegetation with green foliage;
- red / blue: ferric iron;
- shortwave infrared / near infrared: ferrous minerals; and
- shortwave infrared / middle infrared (eg. 1.65 μm/2.22 μm): geological features.

Colour composites of various ratio channels may be used to highlight particular surface features. For example, a composite of the Landsat MSS bands 4/5 as blue, 5/6 as green and 6/7 as red has been reported by many authors to enhance discrimination between altered rock types (Moik 1980; Drury 1987).

Chavez *et al.* (1982) discuss the selection of ratio channels to form colour composites for geological applications. These authors suggest an index of:

$$\frac{\sum \sigma}{\sum \text{absolute correlation values}}$$

where

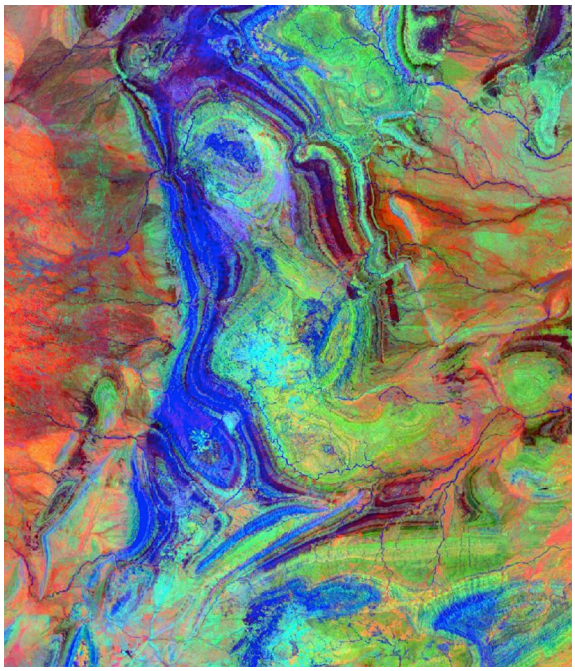
σ is the correlation coefficient (normalized covariance; see Volume 2A—Section 8.1.4)

be computed for each possible combination of channels then the combination with the largest index value be selected. The resulting composite image would have the most variation within, and least correlation between, channels so should be the most visually informative. This method confirmed the popular ratio composite mentioned above.

Crippen (1988) suggests that a colour composite of Landsat Thematic Mapper channels 3/1 as blue, 5/4 as green and 5/7 as red maximises lithological spectral information in well-exposed terrain. A variation on this combination is illustrated in Figure 10.2. Satterwhite and Henley (1987) evaluated a number of band ratioing techniques with Landsat TM data for discriminating spectral characteristics of arid region soils and vegetation conditions. Vogelmann and Rock (1988) found Landsat TM channel ratios of 5/4 and 7/4 correlated well with ground measurements of forest damage (% foliage loss in spruce-fir forests).

Figure 10.2 Colour composite of band ratios

Landsat TM image of ???. where Blue=band 5/band 7 (highlighting clay); Green=band 5/band 4 (highlighting ferrous minerals); Red=band 3/band 1 (highlighting ferric iron (iron oxide)).



Source: Megan Lewis, University of Adelaide

10.1 Calibration Impact

The non-linear nature of the ratio operation accentuates any small variations in the spectral response between different channels so frequently image channels are smoothed before ratioing (see also Section 3.3). Conversely, the similarities between different channels are reduced. This effect may be used to correct for, or reduce, spatially variable 'noise' such as topographic shading in EO imagery or uneven illumination effects in scan-digitised imagery. These sources of noise (represented in the equation below by the term a_i) are related to ground reflectance by a multiplicative model of the form:

$$x_{ij} = a_i \times b_{ij} + d_j$$

where

x_{ij} is the image radiance value at pixel i in channel j

b_{ij} is reflectance (or emissivity) of pixel i in channel j

a_i is illumination (or temperature) at pixel i

d_j is a constant atmosphere or sensor calibration offset factor in channel j (or the 'dark value' as explained below).

These sources of noise can be removed by ratioing using an appropriate reference channel (x_{ir}) that represents the noise pattern:

$$\frac{x_{ij} - d_j}{x_{ir} - d_r} = \frac{a_i \times b_{ij}}{a_i \times b_{ir}} = \frac{b_{ij}}{b_{ir}}$$

where subscript r refers to the reference channel.

Radar backscatter and scanner angular effects can similarly be corrected using ratioing techniques. A multiplicative model applies when the detected noise level is affected by target reflectance, that is, the noise factor is not simply added to the reflectance value of the target. (Where the spatial noise is related to ground reflectance by a linear or additive model, as is the case with atmospheric noise, it should be removed using an affine transformation—see Section 2.1.4.) For scan-digitised imagery, a 'reference' channel is usually scanned from a blank target and this is used as the denominator channel in computing channel ratios. Pre-processing and selection of suitable reference (denominator) channels for noise correction is further discussed in Section 2.1.4.

10.2 Dark Values

Ratio transformations generally subtract a constant 'dark value' from each channel before ratioing (see Section 3.1.5.1). In EO data, the dark value level is assumed to be the upper threshold of the response in a channel which is due to atmospheric effects and sensor calibration offsets (represented by the d_j term in the equation above). This represents the minimum level of information in the channel (see Volume 1A—Section 13).

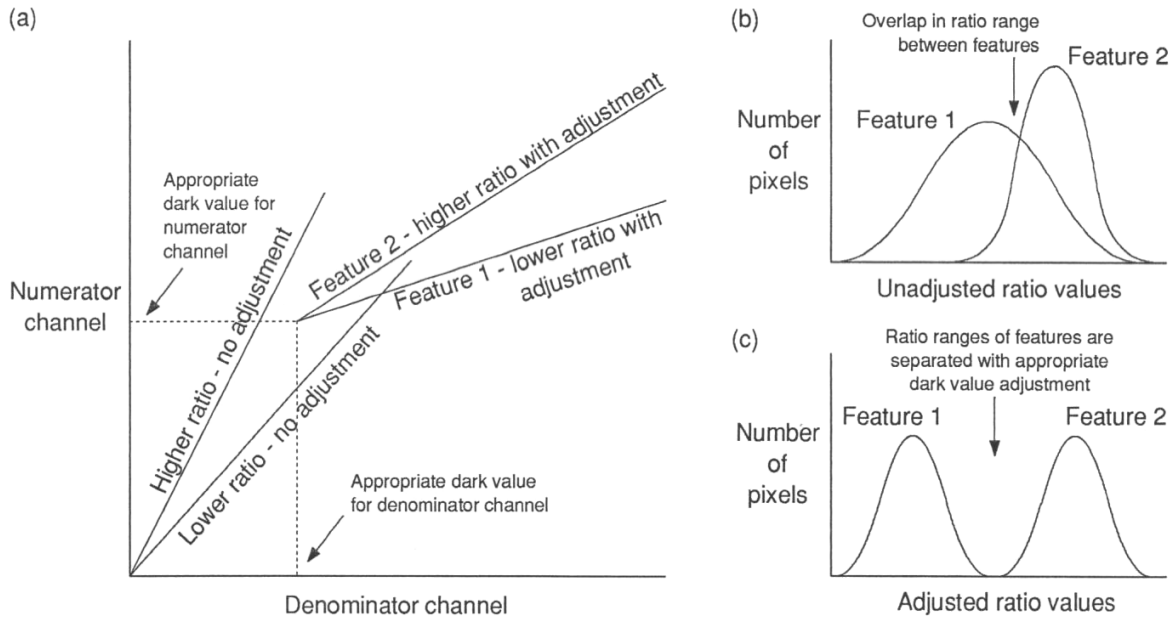
Crippen (1988) demonstrates the importance of appropriate selection of the dark value adjustment for producing useful ratio channels. When image channels are not properly adjusted before ratioing, the resulting channel can contain undesirable overlap between the ratio ranges of different image features (see Figure 10.3).

Figure 10.3 Effect of dark value adjustment in channel ratios

a. The origin of ratios, and hence their stratification of the image data space, is determined by dark value selection.

b. Without dark values the two example features cannot be separated in the ratio channel.

c. Using appropriate dark values, the features have differing ratio ranges.



Source: Harrison and Jupp (1990) Figure 95 [Adapted from Crippen 1988]

Dark values are so named as the values corresponding to the darkest features in the image, such as shadows (Crane, 1971), and may be derived by interactive training on such features (see Volume 2A—Section 9.1.2). A simple approximation of the dark value is one value less than the absolute minimum value of a channel. More sophisticated methods for determining dark values to correct for atmospheric path radiance were proposed by Switzer *et al.* (1981), Crippen (1987) and Chavez (1988).

In some image processing systems, image values less than the dark value become negative (that is, they are not clipped to zero) so a dark value greater than the maximum value in the channel effectively inverts the channel before ratioing. To avoid partial inversion of a channel in ratioing (that is, values less than the dark value in both channels producing a positive ratio value) the dark value should be outside the data range for the channel.

10.3 Directed Band Ratioing

Crippen *et al.* (1988) suggest a method called 'Directed Band Ratioing' (DBR) by which topographic and albedo information can be retained in colour composites formed from ratio channels. Various ratio channels are useful enhancements of geological information and the retention of geomorphic structure as relief shading can improve the interpretation of such imagery.

Directed band rationg involves retaining the illumination information as a consistent intensity variation in all ratio channels so that in a colour composite the illumination components are correlated (channel invariant) and thus only vary as a grey-scale while reflectance (possibly lithographic) variations are uncorrelated (channel variant) between ratios so are shown as colours. Thus, while topographic variations may mask reflectance information in individual channels, correlation of illumination variation between channels in a multi-channel composite separates topography as an achromatic feature from other coloured features. (This correlation could also be advantageous in multi-dimensional analyses such as classification or PCA to isolate topographic or albedo variations).

The procedure for directed band ratioing involves preprocessing the channels by subtracting appropriate dark values, multiplying the denominator channel by a constant, adding a constant to both channels then

ratioing the results. The principle of this processing is to effectively increase the ratio 'angle' of reflectance variations relative to illumination variations in a channel crossplot (see Figure 10.4). In some image processing systems, the required preprocessing can be done using affine transformation (see Section 2.1) before ratioing.

Figure 10.4 Directed band ratioing

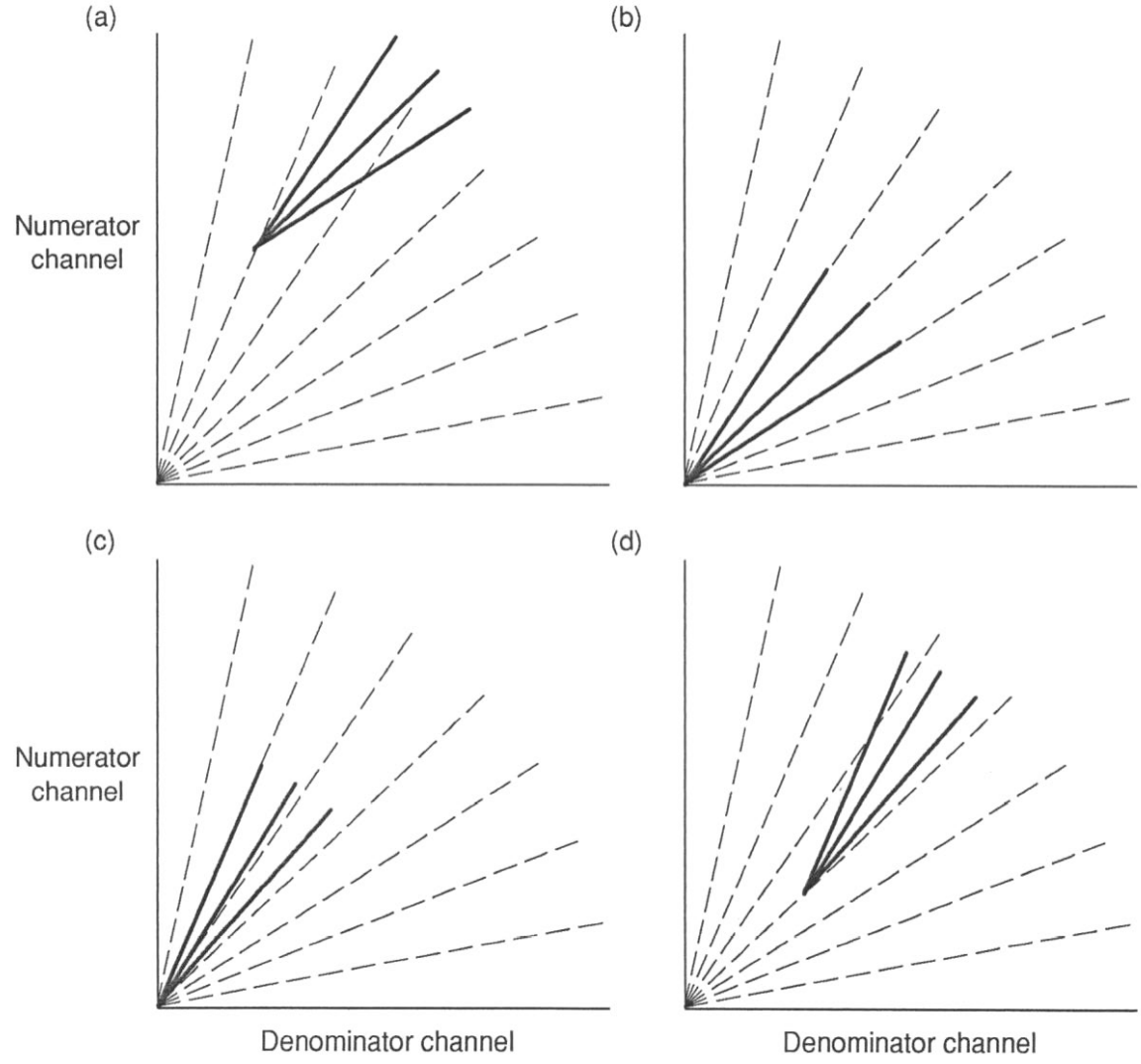
This process increases the ratio 'angle' of reflectance variations in the image relative to the illumination variations. This preserves the topographic shading information, which is generally reduced by ratioing, but can provide useful information for visual interpretation especially for geological applications.

a. Crossplot of original data channels

b. Ratios after dark value adjustment

c. The slope of data lines is increased by multiplying the denominator channel by a constant value.

d. Equal constants are added to each channel before ratioing



Source: Harrison and Jupp (1990) Figure 96 [Adapted from Crippen *et al.* 1988]

10.4 Further Information

Jensen (2016) Section 8

10.5 References

Chavez, P.S. (1988). An improved dark-object subtraction technique for atmospheric scattering correction of multispectral data. Remote Sensing of Environment 24, 459–79.

Chavez, P.S., and Bauer, B. (1982). An automatic optimum kernel size selection technique. Remote Sensing of Environment 12, 23–38.

Chavez, P.S., Berlin, G.L., and Sower, L.B. (1982). Statistical method for selecting Landsat MSS ratios. J. Appl. Photographic Engineering 8, 23–30.Drury, S.A. (1987). Image Interpretation in Geology. Allen and Unwin (Publ) Ltd, London.

Crane, R.B. (1971). Preprocessing techniques to reduce atmospheric and sensor variability in multispectral scanner data. Proc. 7th International Symposium on Remote Sensing of Environment, Ann Arbor, Michigan, 17–21 May, 1345–55.

Crippen, R.E. (1987). The regression intersection method of adjusting image data for band ratioing. Int. J. Remote Sensing 8, 137–55.

Crippen, R.E. (1988). The dangers of underestimating the importance of data adjustments in band ratioing. Int. J. Remote Sensing 9, 767–76.

Crippen, R.E., Blom, R.G., and Heyada, J.R. (1988). Directed band ratioing for the retention of perceptually-independent topographic expression in chromaticity-enhanced imagery. Int. J. Remote Sensing 9, 749–65.

Jensen, J.R. (2016). *Introductory Digital Image Processing: A Remote Sensing Perspective.* 4th edn. Pearson Education, Inc. ISBN 978-0-13-405816-0

Harrison, B.A., and Jupp, D.L.B. (1990). *Introduction to Image Processing: Part TWO of the microBRIAN Resource Manual.* CSIRO, Melbourne. 256pp.

Moik, J.G. (1980). Digital Processing of Remotely Sensed Images. NASA SP–431. Washington, DC. USA.

Satterwhite, M.B., and Henley, J.P. (1987). Spectral characteristics of selected soils and vegetation in northern Nevada and their discrimination using band ratio techniques. Remote Sensing of Environment 23, 155–75.

Switzer, P., Kowalik, W.S., and Lyon, R.J.P. (1981). Estimation of atmospheric path-radiance by the covariance matrix method. Photogramm. Eng. Remote Sensing 47, 1469–76.

Vogelmann, J.E., and Rock, B.N. (1988). Assessing forest damage in high elevation coniferous forests in Vermont and New Hampshire using Thematic Mapper data. Remote Sensing of Environment 24, 227–46.

11 Vegetation Indices

Various indices have been proposed to highlight and differentiate vegetation 'greenness'. These vegetation indices have been well reviewed by many authors, including Jackson (1983), Perry and Lautenschlager (1984) and more recent?).

Healthy green vegetation typically has high reflectance in near infrared and low reflectance in red wavelengths. Other image features may have either high near infrared or low red values so it becomes difficult to identify vegetation pixels using only one of these channels. Most vegetation indices are based on highlighting the difference between the red and near infrared pixel values.

The following three vegetation indices are commonly encountered in EO analyses:

- simple ratio (SR—see Section 11.1);
- normalised difference vegetation index (NDVI—see Section 11.2); and
- Kauth-Thomas greenness transformation (see Section 11.3).

Uses of vegetation indices are further considered in Volume 3A—Terrestrial Vegetation.

11.1 Simple Ratio

Dividing a near infrared channel value by the red channel value for each pixel in the image gives us a ratio result which is high for the vegetation pixels only as illustrated in Figure 11.1. With vegetation, the strengths of both the near infrared reflectance and the red absorption are indicative of plant vigour or type (see Volumes 1 and 3A). Consequently the ratioed values not only identify the feature but also allow us to stratify condition classes within that feature. By summarising the ratio result as a single channel we can subsequently use interactive density slicing to define and highlight these classes (see Volume 2A—Section 9.2.1 and Figure 11.1).

Figure 11.1 Vegetation greenness ratio

Landsat MSS image of Coleambally, NSW, date

a. Near infrared channel
b. Red channel
c. Ratio channel formed by dividing near infrared by red
d. Pseudo-colouring of ratio channel
e. Standard false colour composite image
f. Density-slicing of ratio channel into three categories representing rice crop yield on colour composite background

a. Near infrared channel

b. Visible red channel

c. Ratio channel (near infrared/red)

d. Pseudo-coloured ratio channel

e. Colour composite image

f. Density slicing showing crop yield

Source: Harrison and Jupp (1990) Plate 19

The ratio of near infrared to red reflectance is often referred to as the simple ratio (SR) vegetation index. It is commonly used as a measure of green biomass, but can also indicate water content (Tucker *et al.* 1980a) and crop condition (Tucker *et al.* 1980b, Thompson and Wehmanen 1979). As well as vegetation 'greenness' or condition mapping, these indices have been related to yield using logarithmic or linear regression analysis (for example, ????).

11.2 Normalised Difference Vegetation Index (NDVI)

One of the most commonly used vegetation indices is the 'normalised difference vegetation index' (NDVI), which is computed as:

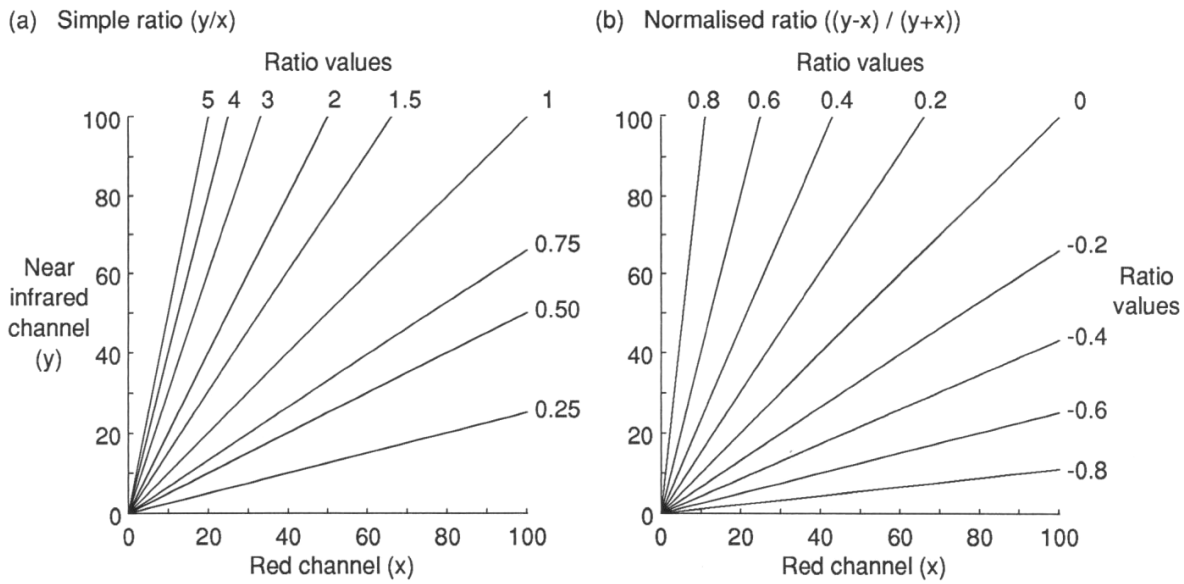
$$\frac{\text{near infrared - red}}{\text{near infrared + red}}$$

As illustrated in Figure 11.2, the normalising of NDVI results in a controlled range of raw ratio values between 1 and -1 and thus greatly simplifies the scaling of output data and comparison of multi-date imagery (Mather 1987). In some image processing systems, NDVI is computed by ratioing the transformed channels after addition and subtraction of the near infrared and red values (see Section 2.1).

Figure 11.2 Normalisation effect in 'Normalised Difference Vegetation Index' (NDVI)

a. Simple ratio

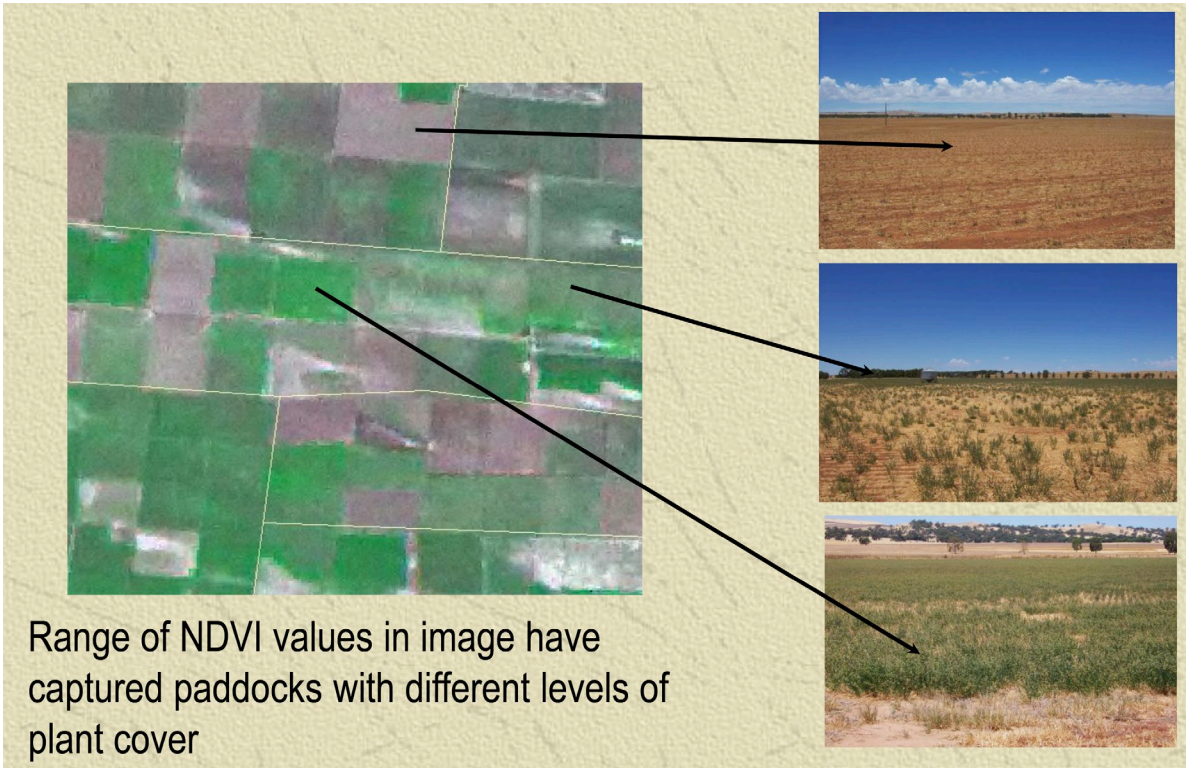
b. Normalised ratio



Source: Harrison and Jupp (1990) Figure 94 [Adapted from Mather 1987]

NDVI has been related to a range of vegetation characteristics, including cover and condition, which may indicate crop yield. For example, the correspondence between image NDVI values and dryland lucerne cover is illustrated in Figure 11.3. Various modifications to the basic index have been proposed to correct for atmospheric or soil-induced variations (such as Huete 1988; Singh and Saull 1988), including the popular Enhanced Vegetation Index (EVI; Huete *et al*., 2002). Other vegetation indices are detailed in Volume 3A.

Figure 11.3 Relating plant cover to NDVI



Source: Megan Lewis, University of Adelaide

11.3 Kauth-Thomas Greenness Transformation

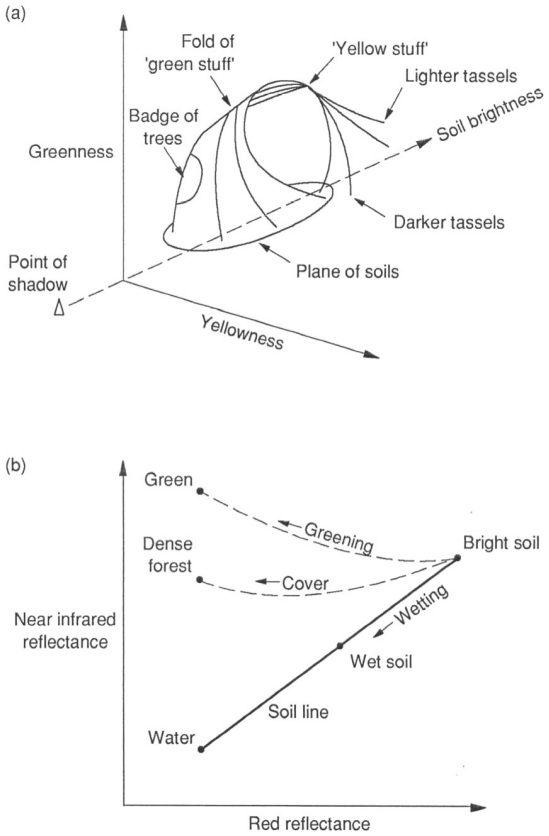
Another vegetation index is computed from a linear combination of visible and near infrared channels. This transformation is sensor specific and was originally defined for Landsat MSS (Kauth and Thomas, 1976). A range of similar transformations now exist for other sensors (Crist and Cicone 1984; Cicone and Metzler 1984) and can be used to analyse specific feature classes in imagery from multiple sensors. Jackson (1983) reviews a number of reported indices and provides a detailed description for calculating the coefficients for 'n-space' indices (that is, linear combinations of n-spectral channels in n-dimensional space) using the Gram-Schmidt process with a minimum of data points.

The Kauth-Thomas transformation describes the essential three dimensions of the Landsat MSS data space as forming the shape of a tasselled cap (see Figure 11.4a). This cap 'sits' on the principal data axis, which has been described as the plane of soils. The initial change in crop reflectance during the crop growing cycle is described as moving down the soil line due to shadowing (though the extent of shadowing varies with the existence and orientation of crop rows). Once maximum soil shadowing has been reached, the trajectory of reflectance moves away from the soil line. This trajectory curves up to the region of green stuff as the crop canopy becomes denser, then folds over to converge on the region of yellow stuff as the vegetation hays off. The shadowed nature of tree canopies renders them a special position in this model, which is fancifully described as the badge on the 'Tasselled Cap'. The point of shadow represents the minimum reflectance level and can be considered as the equivalent of dark values in image ratioing.

Figure 11.4 The 'Tasselled Cap' feature in Landsat MSS image data

a. The major dimensions in Landsat MSS imagery can be described as Brightness, Greenness and Yellowness. The Tasselled Cap feature graphically maps different stages of vegetation growth in terms of these dimensions.

b. One plane through the Tasselled Cap shows the effect of changes in vegetation 'greenness' on red and near infrared reflectance.



Source: Harrison and Jupp (1990) Figure 92

Figure 11.4b illustrates the way changes in vegetation greenness affect its reflectance of red and near infrared wavelengths. This dynamic crossplot represents one plane of the Tasselled Cap space.

Using general land cover categories such as vegetation and soils, a generalised affine transformation was defined to rotate Landsat MSS data channels into dimensions referred to as Brightness, Greenness, Yellowness and Non Such (Kauth and Thomas 1976). The plane of Brightness versus Greenness contains at least 95% of the total variation for agricultural imagery, with Brightness defining the direction of soil reflectance variation and the perpendicular direction, Greenness, being associated with the reflectance characteristics of green vegetation (Crist and Kauth 1986).

The matrix defined for this rotation is shown in Figure 11.5.

Figure 11.5 Kauth Thomas greenness transformation matrix

$$\begin{bmatrix} Brightness \\ Greenness \\ Yellowness \\ Non\ such \end{bmatrix} = \begin{bmatrix} 0.433 & 0.632 & 0.586 & 0.264 \\ -0.290 & -0.562 & 0.600 & 0.491 \\ -0.829 & 0.522 & -0.039 & 0.194 \\ 0.223 & 0.012 & -0.543 & 0.810 \end{bmatrix} \times \begin{bmatrix} MSS4 \\ MSS5 \\ MSS6 \\ MSS7 \end{bmatrix} + \begin{bmatrix} 32 \\ 32 \\ 32 \\ 32 \end{bmatrix}$$

Here we see *Brightness* as a weighted average of the four original channels while *Greenness* is essentially a weighted difference of near infrared and red which is typical for vegetation indices (see Section 14). The *Greenness* feature can also be normalised by the '*Green Brightness*', that is:

$$Unnormalised\ Greenness = -0.29 \times MSS4 - 0.562 \times MSS5 + 0.60 \times MSS6 + 0.491 \times MSS7$$

and

$$Normalised\ Greenness = Unnormalised\ Greenness / Green\ Brightness$$

where

$$Green\ Brightness = 0.29 \times MSS4 + 0.562 \times MSS5 + 0.60 \times MSS6 + 0.491 \times MSS7$$

This normalisation partially compensates for variations in solar elevation and so allows more valid comparisons of greenness changes in multi-temporal images.

A similar transformation was defined for six channels of Landsat TM data, which produced three significant features called *Brightness (or Bareness)*, *Greenness* and *Wetness* (Crist and Cicone, 1984):

$$Brightness = 0.33183 \times TM1 + 0.33121 \times TM2 + 0.55177 \times TM + 0.42514 \times TM4 + 0.48087 \times TM5 + 0.25252 \times TM7$$

$$Wetness = 0.13929 \times TM1 + 0.22490 \times TM2 + 0.40359 \times TM3 + 0.25178 \times TM4 - 0.70133 \times TM5 - 0.45732 \times TM7$$

$$Greeness = -0.24717 \times TM1 - 0.16263 \times TM2 - 0.40639 \times TM3 + 0.85468 \times TM4 + 0.05493 \times TM5 - 0.11749 \times TM7$$

<check>

Variations in these land cover related features are represented by changes in two or more channels in the original imagery. As with PCA, rotation of the original data space to axes which are aligned with the major data variation in land cover features allows them to be more simply analysed and related to ground-based measurements.

Dark values can generally be specified for each channel to represent the level of atmospheric or sensor noise and can be defined by interactive training on shadowed or deep water areas or defaulted to image minimum values as described in Section 14.3.

This transformation has been used in various agricultural models. As a single channel it both allows simpler relationships to be developed between image radiance and ground-based measurements, and permits vegetation categories to be analysed easily using interactive image painting (see Volume 2A—Section 9.2.1).

Mention other indices for water, soil moisture, minerals etc. – can use text in 3A for this

11.4 Further Information

Jensen (2016) Section 8

11.5 References

Cicone, B.C., and Metzler, M.D. (1984). Comparison of Landsat MSS, Nimbus–7, CZCS, and NOAA–7 AVHRR features for land-use analysis. Remote Sensing of Environment 14, 257–65.Crippen, R.E. (1987). The regression intersection method of adjusting image data for band ratioing. Int. J. Remote Sensing 8, 137–55.

Crippen, R.E. (1988). The dangers of underestimating the importance of data adjustments in band ratioing. Int. J. Remote Sensing 9, 767–76.

Crippen, R.E., Blom, R.G., and Heyada, J.R. (1988). Directed band ratioing for the retention of perceptually-independent topographic expression in chromaticity-enhanced imagery. Int. J. Remote Sensing 9, 749–65.

Crist, E.P., and Cicone, R.C. (1984). Application of the Tasselled Cap concept to simulated Thematic Mapper data. Photogramm. Eng. and Remote Sensing 50, 343–52.

Crist, E.P., and Kauth, R.J. (1986). The Tasselled Cap de-mystified. Photogramm. Eng. and Remote Sensing 52, 81–6.

Harrison, B.A., and Jupp, D.L.B. (1990). *Introduction to Image Processing: Part TWO of the microBRIAN Resource Manual.* CSIRO, Melbourne. 256pp.

Huete, A.R. (1988). A soil-adjusted vegetation index (SAVI). Remote Sensing of Environment 25, 295–309.

Huete, A., K. Didan, T. Miura, E. P. Rodriguez, X. Gao, and L. G. Ferreira (2002), Overview of the radiometric and biophysical performance of the MODIS vegetation indices, Remote Sens. Environ., 83, 195–213.

Jackson, R.D. (1983). Spectral Indices in n-Space. Remote Sensing of Environment 13, 409–21.

Jensen, J.R. (2016). *Introductory Digital Image Processing: A Remote Sensing Perspective.* 4th edn. Pearson Education, Inc. ISBN 978-0-13-405816-0

Kauth, R.J., and Thomas, G.S. (1976). The Tasselled Cap – a graphic description of the spectral-temporal development of agricultural crops as seen by Landsat. Proc. Symposium on Machine Processing of Remotely Sensed Data, Purdue University, West Lafayette, Indiana, 4B41–4B51.

Perry, C.R.Jr., and Lautenshlager, L.F. (1984). Functional equivalent of spectral vegetation indices. Remote Sensing of Environment 14, 169–82.

Singh, S.M., and Saull, R.J. (1988). The effect of atmospheric correction on the interpretation of multitemporal AVHRR-derived vegetation index dynamics. Remote Sensing of Environment 25, 37–51.