12
Statistical Image Analysis

12.1 Introduction

We are witnessing a tremendous development in digital imaging. Ten years ago, one needed special equipment to take digital images and quantify them—today it is a routine task. *Image processing* is a well-established discipline with a number of textbooks on the subject: Starck et al. (1998), Petrou and Bosdogianni (1999), Seul et al. (2000), Gonzales and Woods (2002), and Petrou and Petrou (2010). Image processing is concerned with picture enhancement, restoration, and segmentation and is a part of *signal processing.* However, we are concerned with image analysis, particularly *statistical* analysis of a sample of images (we prefer the word *ensemble*). We refer the reader to a book by Barrett and Myers (2004) for a complete in-depth discussion of image science and image reconstruction.

Our primary assumption is that images are random. For example, if one is interested in the quality of paint finishing, one can assume that images of different parts of the painted object will differ up to a certain degree. One wants to know if the image variation is within established limits. Since images are random, the statistical approach becomes relevant. Typical questions of statistical analysis of images are: (a) are two images the same up to a random deviation, or in statistical language, do two images belong to the same general population? (b) are two ensembles of images the same, like a series of images taken before and after treatment? (c) can we generalize a t-test for images? and (d) can we compute a p-value for image comparison as we routinely do in statistics for sample comparison? *Pattern recognition,* as a part of signal processing, also deals with image comparison but in terms of image classification. Although the problems of statistical image comparison and classification are close, they are not the same and, in particular, the latter does not address image-specific variation.

We have to admit that today, statistical image analysis and comparison are unsatisfactory. Perhaps the most advanced image statistics application is functional MRI (fMRI), where time series image frames are analyzed to detect the signal around the brain activation area. But image analysis should go far beyond fMRI because image appeal becomes commonplace in scientific research. For example, a routine practice in cancer research showing two microscopic tissue images before and after treatment as proof of a method's validity is unacceptable. Images vary considerably across sections and animals (or humans). Proper analysis would involve the comparison of dozens or hundreds of images; in our terminology, an *ensemble* of images. This is where the human eye cannot judge and statistics come into play.

The idea of distinguishing two types of image variation, within image and between images, leads immediately to the mixed model as the workhorse of statistical image analysis. Thus, the theory developed in previous chapters becomes the key to the statistical analysis of digital images. For example, images of the same tissue taken at different locations or different time points can be viewed as repeated measurements and therefore may be analyzed by the relevant statistical methodology.

It is routine to use least squares as a criterion in image processing: for example, for image registration. Why least squares? Why not weighted least squares? Why sum of squares? We suggest an elaborative statistical model for images that implies a justified criterion enabling statistical image analysis and comparison.

The goal of this chapter is to lay out the foundation for statistical analysis of images using mixed effects modeling techniques for repeated measurements as described in Chapters 2 through 8. Depending on the level of complexity and sophistication, statistical image models can lead to either linear mixed effects, or the generalized linear or nonlinear mixed effects models. In no way can our description be considered as complete; it serves only as a start for future, more sophisticated statistical image modeling.

12.1.1 What is a digital image?

Mathematically, a digital image is a matrix. Consequently, matrix algebra is the major mathematical tool in image analysis. There are two kinds of image, grayscale and color. A grayscale (monochrome) image is a matrix with intensity represented as integer values from 0 (black) to 255 (white). A grayscale image has 256 levels (0 to 255) because the human eye can distinguish approximately this many levels, and because that number fits conveniently in one byte (a byte has 8 bits, $2^8 = 256$). A *binary* image is a special case of the grayscale format; it has intensity values of only 0 and 255, or pure black and pure white.

Each matrix element corresponds to a pixel on the image. If an image is represented as a $P \times Q$ matrix M, the value at each point on the image is rescaled as $255(M - M_{\min})/(M_{\max} - M_{\min})$ and rounded. In digital imaging, P and Q may be hundreds or even thousands, leading to very large files. For example, a file of a 121×74 gray image will be $121 \times 74 = 8954$ bytes, although the actual size of the image file depends on the format used. See Section 12.5 for a discussion of image compression.

Because the three primary colors (Red, Green and Blue) can be combined to produce other colors, a color image can be represented by three gray images—the

RGB format. Thus, a color image can be represented numerically by a triple of $P \times Q$ matrices with integer values from 0 to 255 as color saturation values. Although to the human eye, color and three gray images do not seem to be equivalent, they are mathematically. We take this approach in this book when dealing with color images. Thus, instead of displaying a color image, we display three gray images. The original color images may be viewed on the Internet.

Many formats are used to store images. Popular image formats include jpg, gif, and tiff. Gray images may be converted to a text file using the Portable Graymap (pgm) format, and color images may be converted to a text file using the Portable Pixelmap (ppm) format. Although the resulting text file will be much larger, the format gives the researcher full access to the image data for further processing. There are many photo-editing, image-processing, and file-conversion products; we use PaintShop Pro from Jasc Software, Inc.

In image analysis we often treat matrix elements as a function of the indices, so in this chapter we use the notation $M(p, q)$ to refer to the (p, q)th element. For example, we work with p and q as arguments for image alignment. Moreover, we shall deal with random p and q.

12.1.2 Image arithmetic

Like matrices, images can be added, subtracted, and even multiplied and divided. However, there is a substantial limitation to image arithmetic because images are generally not aligned. For example, if one is interested in finding the image difference between *before* and *after,* it is tempting to take image difference by subtracting pixel intensities. However, it will soon be realized that pixels on the first image do not exactly correspond to pixels on the second image; therefore, a simple difference does not make sense. Even after image alignment, it may be realized that the objects in the images moved slightly, and again, image difference becomes problematic. Image alignment and registration is an essential problem of further image analysis.

12.1.3 Ensemble and repeated measurements

In statistical image analysis, we usually deal with a sample of images. This is new to image processing, where one deals with one image at a time. A typical problem of statistical image analysis is that one wants to know if two sets of images, $\{\mathbf{M}_{i1}, i = 1, ..., N_1\}$ and $\{\mathbf{M}_{i2}, i = 1, ..., N_2\}$, belong to the same general population. In this book we use the term *ensemble* to indicate sample images. Typically, ensemble means that the images are independent and identically distributed (iid) up to a transformation. Thus, the notion *iid* for images has a broader sense and just means that images are of the same object and independent. Statistical analysis of images is often complicated by the fact that images may have different resolution, size, viewpoint, position, lighting, and so on. We suggest modeling an ensemble of images using repeated measurements or mixed effects methodology. This methodology was applied to a random sample of shapes in Chapter 11 and is well suited to model within- and between-image variation.

12.1.4 Image and spatial statistics

Statistical image analysis and spatial statistics have much in common. Both work with a planar distribution, and therefore an issue of spatial correlation (local dependence of pixels) is central. Classic reference books for spatial statistics are those of Ripley (1981) and Cressie (1991). Statistical analysis of images is more complicated because it usually deals with an ensemble; in addition, images may be available up to a planar transformation. Image analysis borrows the modeling techniques of spatial correlation from spatial statistics, such as SAR and AR models. To model more general image spatial correlation, the theory of a Markov random field is used (Chellappa and Jain, 1993).

12.1.5 Structured and unstructured images

Images may be divided roughly into two groups: structured and unstructured. Examples of the first group are images of an easily recognized object, such as a human face, building, carrot, rug, and so on. Examples of unstructured images are microscopic images, pieces of painted wall, and aerial images. The human eye is good at differentiating an apple from a TV set, but bad when it comes to comparing hundreds of unstructured images, such as medical histology images. An unstructured image may be well represented by its gray level distribution, and therefore it is image-content independent. An advantage of this approach is that a difficult problem of image alignment is eliminated. On the other hand, it may well happen that images of an apple an a TV set will produce the same gray distribution. Thus, a gray-distribution model assumes that images are of the same type of object.

A structured or content-dependent statistical image analysis is more complex because, in addition to image alignment, it requires specification of the image content and spatial correlation. For example, if cells in a microscopic image are elliptical in shape and one wants to count cells, one has to define the cell shape. The simplest structured image is texture where the pattern repeats in a stochastic manner. Examples of textures are wood and fabric (Cross and Jain, 1983).

Structured images are complex, and unlike unstructured images, a multinomial distribution for gray levels may serve as a uniform probabilistic model. Structured images thus require different statistical techniques.

12.2 Testing for uniform lighting

Digital pictures may be taken at different exposures and lighting. In this section we use the F-test of Section 3.8 to test whether the lighting is uniform . Moreover, we estimate the direction and position of the light.

A 662×621 gray image of a rug is shown in Figure 12.1. The following statistical model is assumed:

$$M(p,q) = \beta_0 + \beta_1 p + \beta_2 q + \varepsilon(p,q), \tag{12.1}$$

where $p = 1, ..., P = 662$, $q = 1, ..., Q = 621$ are pixel coordinates, and $\varepsilon(p,q)$ is an error term with zero mean and constant variance. In other words, the grayscale level $M(p,q)$ is considered to be a linear function of the pixel coordinates, p and q.

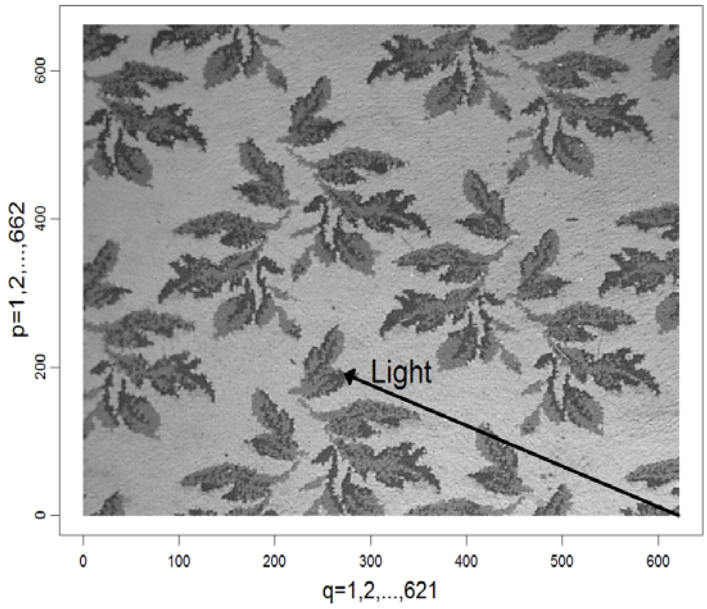

FIGURE 12.1. Rug image and the light direction derived from the linear regression model (the light comes from the bottom-right corner).

The relation may be viewed as a planar regression where β_0 is the intercept, the light intensity at the lower left corner; β_1 is the slope coefficient, the rate at which the light intensity increases or drops vertically; and β_2 is the rate at which the light changes horizontally. If the lighting is uniform, $\beta_1 = 0$ and $\beta_2 = 0$; thus, we can find if the light is uniform by testing the significance of these regression coefficients. In matrix form, (12.1), can be rewritten as

$$\mathbf{M} = \beta_0 \mathbf{1}_P \mathbf{1}_Q' + \beta_1 \mathbf{d}_P \mathbf{1}_Q' + \beta_2 \mathbf{1}_P \mathbf{d}_Q' + \mathbf{E},$$

where $\mathbf{d}_P = (1, 2, ..., P)'$, $\mathbf{d}_Q = (1, 2, ..., Q)'$, and $\mathbf{1}$ is the vector of 1s of the respective dimension. Taking the vec operator of both sides, we rewrite (12.1) as a planar relation in vector form suitable for regression analysis,

$$\mathbf{m} = \beta_0 \mathbf{1}_{PQ} + \beta_1 (\mathbf{d}_P \otimes \mathbf{1}_Q) + \beta_2 (\mathbf{1}_P \otimes \mathbf{d}_Q) + \boldsymbol{\varepsilon},$$

where $\otimes$ indicates the matrix Kronecker product and $\mathbf{m} = \text{vec}(\mathbf{M})$ is the $PQ \times 1$ vector. Applying least squares, we find that $\widehat{M} = 134.4 - 0.0693p + 0.0383q$. As follows from this regression, the average grayscale level is minimum ($\widehat{M} = 88$) when $p = P$ and $q = 1$, corresponding to the upper-left corner of the image (darker). In the lower-right corner the image is lighter ($\widehat{M} = 158$) because maximum $\widehat{M}$ occurs at $p = 1$ and $q = Q$. To test whether the image has uniform lighting we test the null hypothesis: $H_0 : \beta_1 = \beta_2 = 0$ using the F-test (3.57). Since the errors are assumed to be iid, we have $\mathbf{V} = \mathbf{I}$. The residual sum of squares under the null is $RSS_0 = \sum_{p,q}(M(p,q) - \overline{M})^2 = 9.83 \times 10^8$ and the minimal residual sum of squares from regression is $RSS = 8.92 \times 10^8$. For this example, the number of estimated

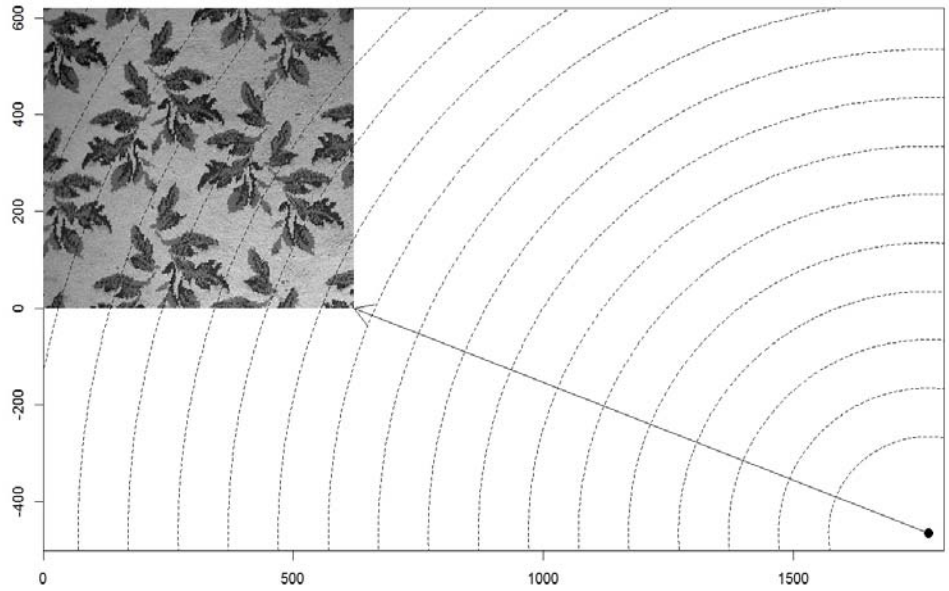

FIGURE 12.2. Source light location with contours via the nonlinear regression model. The linear and nonlinear models yield a close light angle. The location of the light source is estimated at (1770, -465).

parameters is $m = 3$, the total number of observations is $n = P \times Q = 662 \times 621$, and $q = 2$. Formula (3.57) gives $F = 23274$, but the threshold value assuming that the light is uniform is about 3. Thus, the hypothesis that the light is uniformly distributed is rejected overwhelmingly.

12.2.1 Estimating light direction and position

Moreover, having coefficients β_1 and β_2, we may estimate where the light comes from. As noted above, since the first slope coefficient is negative and the second is positive, the light comes from the lower-right corner. Since the tangent of the angle is 0.0693/0.0383, we estimate that the light comes at the angle 150° from the q-axis (indicated by the arrow).The linear model (12.1) assumes that the source light is linear because the levels of the light field are straight lines. Alternatively, one can assume that there is one source of light, say at position (x, y). The levels of the light field are concentric circles. We want to estimate x and y having the rug image in Figure 12.1. As follows from the basic laws of optics, when the light absorption is low, the light intensity is the reciprocal of the distance. To simplify, we take the nonlinear regression model $M(p, q) = 255 - \nu\sqrt{(q-x)^2 + (p-y)^2} + \varepsilon$, where ν is the absorption coefficient and ε is the error term. Estimating parameters of this model by nonlinear least squares gives $\widehat{\nu} = 0.75$, $\widehat{x} = 1770$, and $\widehat{y} = -465$. This means that the estimated location of the light is $(1770, -465)$, see Figure 12.2. Interestingly, the linear and nonlinear models give a similar light angle, indicated by the arrow.

The R function that plots Figure 12.2 and estimates the coordinates of the light source is shown below.

```
 carpet=function()
{
    dump("carpet", "c:\\MixedModels\\Chapter12\\carpet.r")
    d <- scan("c:\\MixedModels\\Chapter12\\carpetc.pgm",what="")
    d <- as.numeric(d[2:length(d)])
    K <- d[1] # number of rows
    J <- d[2] # number of columns
    y=d[4:length(d)] # image data
    carp.dat <- matrix(y, nrow = K, ncol = J)
    x1=rep(1:K,times=J);x2=rep(1:J,each=K)
    print("Linear model:")
    o <- lm(y ~x1 + x2)
    print(summary(o))
    print("Nonlinear model:")
    o <- nls(y ~a1 * sqrt(((a2 - x1)^2 + (x2 - a3)^2)),
        start = c(a1 = sqrt(0.006), a2 = 2000, a3 = -500))
    print(summary(o))
    a <- coef(o)
    par(mfrow = c(1, 1), err = -1, mar = c(3, 3, 1, 2))
    image(1:J, 1:K, t(carp.dat), axes = T, xlab = "", ylab = "",
        xlim = c(0, 1800), ylim = c(-500, J),col=gray(0:255/255))
    points(a[2], a[3], pch = 16, cex = 1.5)
    N <- 30
    h <- 3000/N
    theta <- seq(from = 0, to = 8 * atan(1), by = 0.01)
    for(i in 2:(N - 1)) {
        dc <- h * i
        x <- dc * cos(theta)
        y <- dc * sin(theta)
        lines(a[2] + x, a[3] + y, lty = 2)
    }
    arrows(a[2], a[3], J, 0)
}
```

We make a few remarks on this code: (1) the txt file with this code is saved using `dump` command every time the code is issued; and (2) the carpet image is downloaded into an R session using the `scan` command as an array string, the third and fourth elements of the array are the number of rows and columns in the image,

and the image itself starts from the fifth element (this is a typical representation of the image in the PGM format).

Problems for Section 12.2

1. Test the hypothesis that the distance from the light source to the bottom-right corner of the rug is more that 2000 pixels. Approximate the squared distance $(\widehat{x}-621)^2+\widehat{y}^2$ by a normal distribution and use the delta method to estimate the variance.

2*. Take a picture of an object near the window and measure the location of the window. Save the image in PGM format and repeat the analysis of the light location. Is your estimate close to the actual window location?

12.3 Kolmogorov–Smirnov image comparison

An essential task of image analysis is image comparison. Under the assumption that images are subject to random noise, we want to test if the images are the same. In this and the following section we say that two images are the same if they have the same grayscale distribution. Clearly, if two images are the same, up to a small noise, they should have close grayscale distributions. The reverse is not true. Thus, grayscale distribution analysis is helpful when images of the same content are compared. In this section and the next, nonparametric and parametric approaches are developed.

The image histogram is a frequently used technique of image processing. However, in addition to the histogram, one can compute the distribution, or more specifically, the cumulative distribution function, as the sum of the probabilities that a pixel takes a grayscale level of less than g, where $g=0,...,255$. If $\{h_g\}$ is the image histogram,

$$F_g=\sum_{g'=0}^{g} h_{g'} \tag{12.2}$$

is the empirical cumulative distribution function. As for any distribution function, F_g is nondecreasing within the interval $[0,1]$.

One advantage of distribution analysis is that it facilitates visual image comparison by plotting grayscale distribution functions on the same scale (it is difficult to plot several histograms on the same scale). Another advantage is that the distribution function allows the application of nonparametric statistical tests, such as Kolmogorov–Smirnov.

12.3.1 Kolmogorov–Smirnov test for image comparison

Let $F^{(1)}=\{F_g^{(1)}, g=0,...,255\}$ and $F^{(2)}=\{F_g^{(2)}, g=0,...,255\}$ be two gray level distributions for the $P_1\times Q_1$ and $P_2\times Q_2$ images M_1 and M_2. We compute the maximum, $\widehat{D}=\max_g\left|F_g^{(1)}-F_g^{(2)}\right|$, the distance of one distribution from the other. Kolmogorov and Smirnov proved that if theoretical distributions are the same, then

the probability that the observed distance, $\widehat{D}$, is greater than D is

$$Q_{KS}(\lambda) = 2\sum_{j=1}^{\infty}(-1)^{j-1}\exp\left(-2j^2\lambda^2\right),$$

where $\lambda_{KS} = D[\sqrt{J} + 0.11/\sqrt{J} + 0.12]$ and $J = (P_1Q_1P_2Q_2)/(P_1Q_1 + P_2Q_2)$; see Hollander and Wolfe (1999) for more details. Thus, $Q_{KS}(\lambda)$ may be treated as the p-value of the test. The greater the distance between distributions, the less the probability $Q_{KS}(\lambda_{KS})$. For example, if two images yield distance D and the computed probability $Q_{KS}(\lambda_{KS}) < 0.05$, we reject the hypothesis that the two images are the same with a 5% error. We can find λ_{KS} such that $Q_{KS}(\lambda_{KS}) = 0.05$, giving the threshold $\lambda_{KS} = 1.358$.

As a word of caution, all nonparametric tests, including the Kolmogorov–Smirnov, have the alternative $H_A : F_1(x) \neq F_2(x)$ for at least one x. Thus, these tests may be conservative.

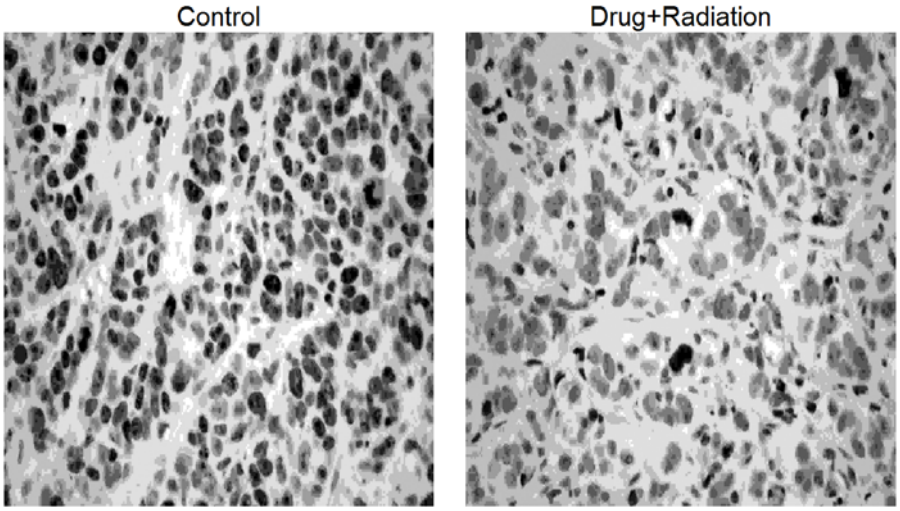

FIGURE 12.3. Histology sections of untreated and treated tumors. The living cancer cells are the dark spots (blobs). To test statistically that the two images have the same grayscale distributions, the Kolmogorov–Smirnov test is used.

12.3.2 Example: histological analysis of cancer treatment

We illustrate the Kolmogorov–Smirnov test with a histological analysis of breast cancer treatment (Sundaram et al., 2003). Two 512×384 images of proliferative-activity tumor tissue sections are shown in Figure 12.3. The dark blobs are cancer cells. In the control tumor (left), no treatment was given. In the treated tumor (right), a combination of drug, EB 1089, and radiation seems to have reduced the number of living cancer cells. We want to confirm this reduction statistically by the Kolmogorov–Smirnov test by computing the p-value.

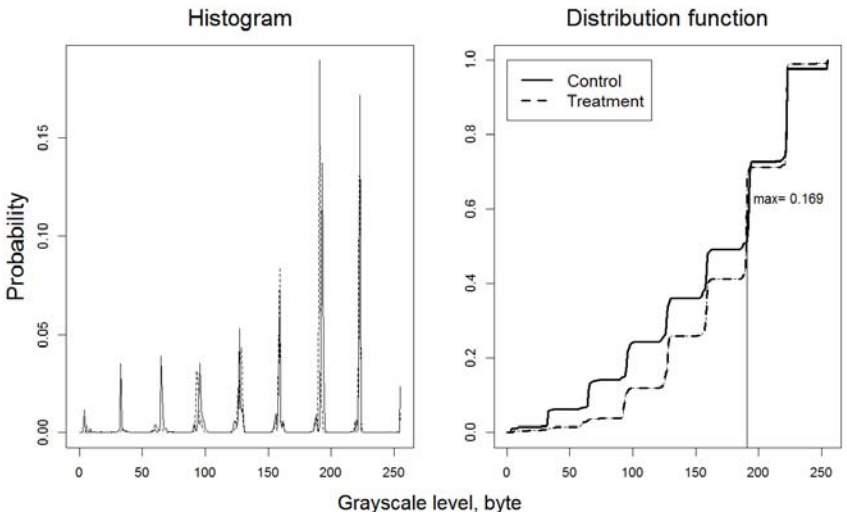

FIGURE 12.4. Histogram and cumulative distribution functions for two histology images. The distribution function of the treated tumor is less (almost everywhere) than that of the control (the maximum difference is 0.169), which means that the control image is darker. The Kolmogorov–Smirnov distribution function distance is used to test the statistical significance.

The grayscale histogram and the distribution functions for these images are shown in Figure 12.4. Clearly, it is difficult to judge the difference in the images by histogram. To the contrary, the distribution functions reveal the difference with the absolute maximum 1/10. We notice that the treatment distribution function is below control (for most gray levels), which means that the right image is lighter. For these images $P_1Q_1 = P_2Q_2 = 512 \times 384 = 1.9661 \times 10^5$, yielding $\lambda_{KS} = 52.882$ and $Q(\lambda_{KS}) < 0.0001$, near zero. Since the p-value is very small, we infer that the null hypothesis that the two images are the same should be rejected.

The `R` program that plots the two images (Figure 12.3) and computes λ (Figure 12.4) is shown below.

```
 KSimage=function(job=1)
{
dump("KSimage","c:\\MixedModels\\Chapter12\\KSimage.r")
if(job==1)
{
 par(mfrow = c(1, 2), mar = c(1, 1, 3, 1), omi = c(0, 0, 0, 0))
 d <- scan("c:\\MixedModels\\Chapter12\\grp11.pgm",what="")
 d <- as.numeric(d[9:length(d)])
 nr <- d[1];nc <- d[2]
 d <- matrix(d[4:length(d)], nrow = nr, ncol = nc)
```

```
 image(1:nr, 1:nc, d, xlab = "", ylab = "", axes = F,
                  col=gray(0:255/255))
 mtext(side = 3, "Control", line = 0.25, cex = 2)
 d <- scan("c:\\MixedModels\\Chapter12\\grp51.pgm",what="")
 d <- as.numeric(d[9:length(d)])
 nr <- d[1];nc <- d[2]
 d <- matrix(d[4:length(d)], nrow = nr, ncol = nc)
 image(1:nr, 1:nc, d, xlab = "", ylab = "", axes = F,
                  col=gray(0:255/255))
 mtext(side = 3, "Drug+Radiation", line = 0.25, cex = 2)
}
if(job==2)
{
 par(mfrow = c(1, 2), mar = c(4, 4, 3, 1))
 d <- scan("c:\\kluwer\\image\\sujatha\\grp11.pgm",what="")
 d <- as.numeric(d[9:length(d)])
 nr <- d[1]
 nc <- d[2]
 J1 <- nr * nc
 d1 <- d[4:length(d)]
 d <- scan("c:\\kluwer\\image\\sujatha\\grp51.pgm",what="")
 d <- as.numeric(d[9:length(d)])
 nr <- d[1]
 nc <- d[2]
 print(c(nr, nc))
 J2 <- nr * nc
 d2 <- d[4:length(d)]
 h1 <- f1 <- h2 <- f2 <- rep(0, 256)
 for(i in 0:255) {
 h1[i + 1] <- length(d1[d1 == i])/length(d1)
 h2[i + 1] <- length(d2[d2 == i])/length(d2)
 }
 for(i in 2:256) {
 f1[i] <- f1[i - 1] + h1[i]
 f2[i] <- f2[i - 1] + h2[i]
 }
 f1[256] <- f2[256] <- 1
 matplot(cbind(0:255, 0:255), cbind(h1, h2), type = "l",
                          col = 1,xlab="",ylab="")
 mtext(side = 2, "Probability", line = 2.5, cex = 1.75)
 mtext(side = 3, "Histogram", line = 1, cex = 1.75)
```

```
 matplot(cbind(0:255, 0:255), cbind(f1, f2), type = "l",
                          col = 1,xlab="",ylab="")
 lines(0:255, f1, lwd = 3)
 lines(0:255, f2, lwd = 3, lty = 2)
 mf <- max(abs(f1 - f2))
 jm <- (0:255)[abs(f1 - f2) == mf]
 segments(jm, -0.25, jm, 0.6)
 text(jm+5, 0.63, paste("max=", round(mf, 3)),adj=0)
 mtext(side = 3, "Distribution function", line = 1, cex = 1.75)
 legend(0, 1, c("Control", "Treatment"), lty = 1:2, cex = 1.25,
                                      lwd = 3)
 mtext(side = 1, "Grayscale level, byte", line = -1,
                           outer = T,cex = 1.5)
 J <- (J1 * J2)/(J1 + J2)
 lambda <- mf * (sqrt(J) + 0.11/sqrt(J) + 0.12)
 j <- 1:10000
 js <- rep(1, 10000)
 js[seq(from = 2, to = 10000, by = 2)] <- -1
 Q <- 2 * sum(js * exp(-2 * j^2 * lambda^2))
 cat("\nbyte.max =",jm," max.cdf.distance =",round(mf,3),
    " lambda =",round(lambda,3)," QKS =", round(Q,4),"\n")
}
}
```

We make several comments: (1) The two tasks correspond to `job=1` (Figure 12.3) and `job=2` (Figure 12.4); (2) different software uses different preambules when saving images in PGM format, in this particular case starting from the 9th element; and (3) arrays `h1` and `h2` contain histograms and `f1` and `f2` contain cdf values for the two images, alternatively, one can use the `cumsum` function to compute the cdf.

Problems for Section 12.3

1. Plot $Q_{KS}(\lambda)$ as a function of λ using `lambda=seq(from=1,to=2,by=.01)`. Replace ∞ with 1000 and use matrix operation to avoid the loop: compute the value under the sum as a matrix with 1000 rows and `length(lambda)` columns. Then use `%*%` to get $Q_{KS}(\lambda)$. Confirm that $Q_{KS}(1.358) \simeq 0.05$.

2. Use a t-test (Z-score) to test that the treatment image is darker than the control (this is how images are compared using traditional statistics). Does this test confirm the KS test? What test is preferable?

12.4 Multinomial statistical model for images

The aim of this section is to develop a statistical model for gray images based on the grayscale distribution (or simply, gray distribution). As in the previous section, it is

assumed that the image is specified by 256 grayscale levels, and therefore our image analysis is content-independent. Based on this model, we shall develop parametric tests for image comparison.

The gray distribution of an image $\{M(p,q), p = 1, ..., P, q = 1, ..., Q\}$ is specified by 256 probabilities $\pi_g = \Pr(M = g)$, where $g = 0, 1, ..., 255$ is the gray level. Assuming that among PQ image pixels there are k_0 black pixels, the k_1 pixels have gray level $g = 1$, k_2 pixels have gray level $g = 2$, etc. the mutual probability can be modeled via the *multinomial* distribution, namely,

$$f(k_0, k_1, ..., k_{255}) = \frac{(PQ)!}{k_0!k_1!...k_{255}!}\pi_0^{k_0}\pi_1^{k_1}...\pi_{255}^{k_{255}}, \tag{12.3}$$

Rao (1973), Bickel and Doksum (2001), Agresti (2002). In this formula, the sum of probabilities is 1, $\sum_{g=0}^{255}\pi_g = 1$, and $\sum_{g=0}^{255}k_g = PQ$ is the total number of pixels. Value k_g is called the frequency, so (12.3) specifies the probability that in PQ independent experiments the random variable M takes the value 0 k_0 times, takes the value 1 k_1 times, and so on. An important assumption is that these random experiments are independent. For a binary image, the multinomial distribution reduces to the binomial distribution. Model (12.3) is called a *multinomial* model for gray images.

The log-likelihood function for the multinomial gray level model is

$$l(\pi_0, ..., \pi_{255}) = C + \sum_{g=0}^{255} k_g \ln \pi_g,$$

where $C = \ln(PQ)! - \sum_{g=0}^{255}\ln k_g!$, a constant. To estimate probabilities $\{\pi_g\}$ from a $P \times Q$ image by maximum likelihood, we maximize l with respect to $\{\pi_g\}$ under the restriction that $\sum_{g=0}^{255}\pi_g = 1$. Introducing the Lagrangian

$$\mathcal{L}(\pi_0, ..., \pi_{255}, \lambda) = \sum_{g=0}^{255} k_g \ln \pi_g - \lambda\left(\sum_{g=0}^{255}\pi_g - 1\right),$$

and taking derivatives with respect to π_g, we obtain $\partial\mathcal{L}/\partial\pi_g = k_g/\pi_g - \lambda = 0$ which implies that $\pi_g = k_q/\lambda$. Hence, the ML estimate of the gth probability is

$$\widehat{\pi}_g = h_g = \frac{k_g}{PQ}. \tag{12.4}$$

This is a familiar estimate of the gray level probability—the ML estimate is just the histogram value, h_g. The variance and covariance of these probability estimates are

$$\mathrm{var}(\widehat{\pi}_g) = \frac{1}{PQ}\pi_g(1 - \pi_g), \quad \mathrm{cov}(\widehat{\pi}_g, \widehat{\pi}_{g'}) = -\frac{1}{PQ}\pi_g\pi_{g'}. \tag{12.5}$$

To estimate the variance and covariance, we use the histogram value h_g instead of π_g in formulas (12.5). As follows from (12.5), $\widehat{\pi}_g$ is consistent because its variance vanishes for large images, $PQ \to \infty$. We shall apply the multinomial model for image comparison and entropy computation.

12.4.1 Multinomial image comparison

Using the notation of Section 12.3, let $M^{(1)}$ and $M^{(2)}$ be $P_1 \times Q_1$ and $P_2 \times Q_2$ gray images. We want to test whether they have the same gray distribution. In other terms, the null hypothesis is $H_0 : \pi_0^{(1)} = \pi_0^{(2)}, ..., \pi_{255}^{(1)} = \pi_{255}^{(2)}$. Two tests can be suggested, assuming that the images are independent: the χ^2 and the likelihood ratio test. Unlike the Kolmogorov–Smirnov test, these tests are parametric because they assume the multinomial distribution specified by the 256 probability parameters.

In the χ^2-test, we estimate $256 \times 2 = 512$ probabilities $\widehat{\pi}_g^{(1)} = h_{1g}$ and $\widehat{\pi}_g^{(2)} = h_{2g}$ as the proportion of pixels with the gray level g. Assuming that images are independent, the variance of the difference is the sum of variances, and as follows from formula (12.5), $\text{var}(h_{1g} - h_{2g}) = h_{1g}(1 - h_{1g})/(P_1Q_1) + h_{2g}(1 - h_{2g})/(P_2Q_2)$. Then, under the null hypothesis, the scaled sum of squares approximately has a χ^2-distribution,

$$\sum_{g=0}^{255} \frac{(h_{1g} - h_{2g})^2}{h_{1g}(1 - h_{1g})/(P_1Q_1) + h_{2g}(1 - h_{2g})/(P_2Q_2)} \sim \chi^2(255). \tag{12.6}$$

We take off one degree of freedom because the sum of probabilities is 1. This has little effect because the total degrees of freedom is large. If both h_{1g} and h_{2g}, are zero, the corresponding term is dropped from the summation. One can interpret (12.6) as the squared scaled distance between the two histograms. Following the line of statistical hypothesis testing, if the value on the left-hand side of (12.6) is greater than the $(1 - \alpha)$th quantile of the χ^2-distribution with 255 degrees of freedom, we reject the hypothesis that the two images have the same gray level distribution with error α. Alternatively, one can report the p-value as the χ^2-tail density.

In (12.6) we assumed that $\{h_{1g} - h_{2g}, g = 0, ..., 255\}$ are independent, but as follows from (12.5), they are negatively correlated. To account for correlation, we remove the histogram component with the maximum value $h_{1g}+h_{2g}$ so that $\mathbf{h}_{1*}-\mathbf{h}_{2*}$ is the 255×1 vector of histogram differences with the corresponding 255×255 covariance matrix $\mathbf{V}_{1*} + \mathbf{V}_{2*}$. By construction, the sum of elements is less than 1 and the covariance matrix is nonsingular. Then, in matrix form, similar to (12.6), we have an alternative χ^2-test,

$$(\mathbf{h}_{1*} - \mathbf{h}_{2*})'(\mathbf{V}_{1*} + \mathbf{V}_{2*})^{-1}(\mathbf{h}_{1*} - \mathbf{h}_{2*}) \sim \chi^2(255). \tag{12.7}$$

In the likelihood ratio test, we need to compute three log-likelihood values, an individual value from each image and a combined value. More precisely, the maximum value of the log-likelihood function from image $i = 1, 2$ can be expressed in terms of frequencies $\{k_{ig}\}$ as follows:

$$\begin{aligned} l_i &= \ln n_i! - \sum \ln k_{ig}! + \sum k_{ig} \ln k_{ig} - \ln n_i \sum k_{ig} \\ &= (\ln n_i! - n_i \ln n_i) + \sum (k_{ig} \ln k_{ig} - \ln k_{ig}!), \end{aligned}$$

where $n_i = P_iQ_i$ is the number of pixels in the ith image. Next, we combine the two images into one gray level set with $n_3 = P_1Q_1 + P_2Q_2$ elements yielding the frequencies $\{k_{3g}, g = 0, ..., 255\}$ and the resulting log-likelihood maximum value

$$l_3 = (\ln n_3! - n_3 \ln n_3) + \sum (k_{3g} \ln k_{3g} - \ln k_{3g}!).$$

According to the likelihood ratio test, under the null hypothesis,

$$2(l_1 + l_2 - l_3) \sim \chi^2(256). \tag{12.8}$$

Again, if the left-hand side of (12.8) is larger than the $(1-\alpha)$th quantile of the χ^2-distribution, we reject the hypothesis.

Problems for Section 12.4

1. Apply the χ^2 and the likelihood ratio tests to the histologic images in the previous section. Compute the p-values for both tests. Modify the function `KSimage`. Use `lgamma(k+1)` to compute $\ln k!$.

2*. Generate synthetic treatment images from the control image in Figure 12.3 using the formula $M'_\nu(p,q) = \lfloor 255 \times (M(p,q)/255)^\nu \rfloor$, where $M(p,q)$ is the gray intensity of the (p,q)th pixel of the original control image, ν is a positive number ($\nu = 1$ does not change the image), and $\lfloor . \rfloor$ means that the number is rounded to the nearest smallest (`R` function `floor`); see the function `KSimageR`. Parameter ν is image-specific, e.g. it may take values according to a beta distribution with parameters a and β. Use this method to generate random images for computing the power function of image comparison for three methods: KS, chi-square, and log-likelihood. Make your statement regarding the efficiency of the methods.

12.5 Image entropy

The purpose of an image is to convey information; thus information theory can play an important role in image analysis. Specifically, we use the notion of *entropy* to measure the amount of image information (Resa, 1961; Kulback, 1968). In the image processing literature, image entropy is used in the context of image coding. Here, we apply this concept for optimal image reduction. In this section we show how image entropy can be used for optimal image reduction and enhancement. Although entropy, as the major concept of Shannon communication theory, has some application in image science (see Barrett and Myers, 2004 for a review), we apply it for optimal image reduction. First, we demonstrate the use of entropy for binary images and then for gray and color images.

Recall that if X is a discrete random variable, which takes values $\{x_i, i = 1, 2, ..., n\}$ with probability $\{\Pr(X = x_i) = p_i\}$, the entropy is defined as $\mathcal{E} = -\sum_{i=1}^{n} p_i \log p_i$. In the special case when X is binary, the entropy is $\mathcal{E} = -[p \log p + (1-p)\log(1-p)]$. Usually, in communication theory, the base of the logarithm is taken to be 2, making interpretation of the entropy a piece of information measured in bits. For example, for a binary random variable with $p = 0$, the entropy is zero. Indeed, by L'Hospital's rule,

$$\begin{aligned} \mathcal{E}(0) &= -\lim_{p\to 0} p\log_2 p - (1-0)\log_2(1-0) \\ &= -\lim_{p\to 0} \frac{\log_2 p}{p^{-1}} = \frac{1}{\ln 2}\lim_{p\to 0}\frac{1/p}{1/p^2} = 0. \end{aligned}$$

Similarly, one can prove that $\mathcal{E}(1) = 0$. These results have a clear interpretation: when $p = 0$ or $p = 1$, the binary variable takes a constant value, and therefore there

is no information in the message. Maximum entropy in a binary message occurs when $p = 1/2$. Indeed, differentiating the entropy, one obtains

$$\begin{aligned} \mathcal{E}' &= -1 - \log_2 p + \log_2(1-p) + 1 = \log_2(1-p) - \log_2 p, \\ \mathcal{E}'' &= -p^{-1} - (1-p)^{-1} = -1/[p(1-p)] < 0. \end{aligned}$$

The inequality says that $\mathcal{E}$ is a concave function on $(0, 1)$. Maximum $\mathcal{E}$ occurs where $\mathcal{E}' = 0$, which yields $p = 1/2$. This means that maximum information contained in a sequence of zeros and ones is attained when they occur with equal probability.

For example, the amount of information in a $P \times Q$ binary image with the proportion of white pixels equal p is

$$PQ \times \mathcal{E}(p) = -PQ\left[p \log_2 p + (1-p)\log_2(1-p)\right] \text{ bits.} \tag{12.9}$$

In (12.9) it is assumed that pixel gray levels are uncorrelated. Now we apply the entropy notion to optimal image gray level reduction.

FIGURE 12.5. The original image Lena, the canon of image processing, and three optimal reductions with minimum information loss. The Entropy Per Pixel (EPP) of the original image is close to the absolute maximum, 8 bits. Information in the binary image is almost one-eighth of that in the original image.

12.5.1 Reduction of a gray image to binary

Let M be the original $P \times Q$ gray image with distribution $F = \{F_g, g = 0, ..., 255\}$. We want to reduce this image to a binary image by determining a threshold g_* such that all gray levels less than g_* are set to 0 and all gray levels greater than g_* are set to 255. We want to determine a g_* that reduces the original gray image with minimum information loss. As follows from (12.9), the entropy of a binary image gets maximum when the number of black and white pixels is the same, $p = 1/2$. Thus, g_* is defined by the equation $F_{g_*} = 1/2$, the median. In other words, an optimal threshold is the median of gray distribution. This choice yields minimum information loss.

We illustrate the image reduction in Figure 12.5. This image, *Lena*, is a canon in the image processing community, and many works use this image as an example. The reduction of the original image to binary is such that the number of black and white points in the binary image is the same. The `R` function that plots Figure 12.5 and computes EPP can be downloaded by issuing

```
source("c:\\MixedModels\\Chapter12\\lena.r")
```

12.5.2 Entropy of a gray image and histogram equalization

Histogram equalization is a well known technique of image processing that helps improve an image. In this section we provide an information basis for this technique and develop a general algorithm to reduce a gray image with minimum information loss.

Assuming that levels of a gray image follow a multinomial distribution (12.3), the entropy of a $P \times Q$ image is defined as

$$\mathcal{E} = -PQ \sum_{g=0}^{255} \pi_g \log_2 \pi_g.$$

Since the theoretical probability, π_g, is estimated by the histogram value, h_g, we come to the following.

Definition 45 *Image Entropy Per Pixel (EPP) is defined as*

$$EPP = -\sum_{g=0}^{255} h_g \log_2 h_g \ \textit{bits}, \tag{12.10}$$

where h_g is the histogram value.

Theorem 46 *The absolute maximum of EPP is 8 bits. This maximum is attained when each of 256 gray levels occurs with equal probability 1/256 (i.e., when the histogram is flat).*

Proof. The proof is similar to the maximum likelihood estimation at the beginning of this section. We want to maximize (12.10) over $\{h_g\}$ under the restriction $\sum_{g=0}^{255} h_g = 1$. Introducing the Lagrangian

$$\mathcal{L}(h_0, ..., h_{255}, \lambda) = \sum_{g=0}^{255} h_g \ln h_g - \lambda \left(\sum_{g=0}^{255} h_g - 1 \right)$$

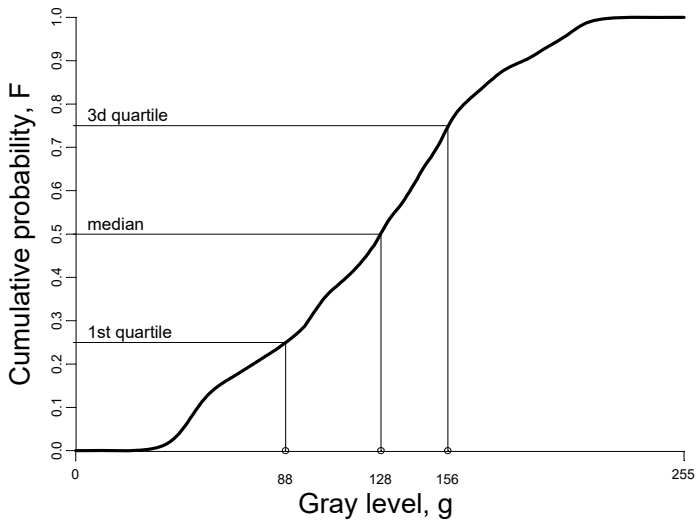

FIGURE 12.6. Optimal reduction of the original image, Lena to a binary and four-level gray image. To reduce Lena to a four-level image with minimum information loss, the thresholds must be 88 (first quartile), 128 (median), and 156 (third quartile).

and taking the derivative with respect to h_g, we obtain h_g =const= 1/256. This corresponds to the flat histogram. The absolute maximum is

$$EPP = -256\frac{1}{256}\log_2\frac{1}{256} = 8.$$

■

This theorem creates a theoretical basis for histogram equalization technique: by modifying the gray levels to make the histogram flat, we increase the EPP to a maximum.

For example, as follows from entropy theory, to reduce a gray image to an image with four gray levels with minimum information loss, the thresholds must be the quartiles of the distribution function F, see Figure 12.6. In other words, let q_1 (the first quartile) be the gray level such that $(PQ)/4$ pixels of the original image have gray levels of less than q_1. Let q_2 be the median, i.e., 50% of pixels have gray levels of less than q_2. Finally, let q_3 (the third quartile) be the gray level such that the number of pixels with gray levels greater than q_3 is $(PQ)/4$. This choice of thresholds makes the four-level image the most informative.

For a binary image with an equal number of black and white pixels EPP = 1 bit because $-0.5\log_2(1/2) = 1$. For an image with four equal gray levels, EPP = 2 bits. Generally, if an image has 2^m equal gray levels, EPP = m bits (see Figure 12.5).

Problems for Section 12.5

1. The frequency of English letters can be downloaded as `read.table("c:\\MixedModels\\Chapter12\\EnglishLetters.txt")`. Compute the entropy of the statement: *I love statistics* based on the letters frequency.

2. Prove that the maximum enropy of a categorical random variable is attained at $p_i = 1/n$, where $p_i \geq 0$ and $\sum_{i=1}^{n} p_i = 1$.

3. Define a 16-bit image and compute the maximum entropy.

4. The function `histgr` plots the two images depicted in Figure 12.7. Apply histogram equalization to the original image. Compare your result with that produced by commercial software.

FIGURE 12.7. Two images produced by function `histgr`. Histogram equalization is done by commercial software. Use your own histogram equalization image processing technique to see if the result is the same.

12.6 Ensemble of unstructured images

Usually, one deals with several images of the same object. Then we speak of an *ensemble* of images. Similar to a sample of shapes, we distinguish two features. On one hand, it is assumed that images are *of the same object*, or in statistical terms, they belong to the same general population. On the other hand, they may have image-specific features associated with image variation. For example, if several aerial pictures are taken to assess fire damage, they are all taken over the same area but may differ because the fire damage is not uniform. The same principle was used in the previous chapter, where within- and between-shape variations were recognized. In contrast, in classical statistics, it is assumed that a sample is drawn from one general population and therefore that there would be no room for image-specific variation. In the language of the mixed model methodology, population features are described by population-averaged parameters, and image-specific features are specified by image (or cluster, subject) -specific parameters.

In this section we deal with a sample of independent gray images defined by the $P_i \times Q_i$ integer (gray level) matrices $\mathbf{M}_i, i = 1, 2, ..., N$. Typical questions: (a) Are N images the same? (b) Are two groups of images the same? (c) Does image $(N + 1)$ belong to the same group of images? A statistical test for two gray images was developed in Section 12.3, here we assume that $N > 2$.

There are two extreme approaches to modeling an ensemble of images. First, we can assume that all N images $\mathbf{M}_i$ have *identical* gray level multinomial distributions (12.3). Then one can pool the images and obtain estimates of the probabilities as if we had only one image. Clearly, if all images have the same size, the histogram and the distribution function are the arithmetic means of the individual histograms and the distribution functions. Under this assumption, a comparison of two groups of images reduces to a comparison of two distribution functions, and the Kolmogorov–Smirnov test applies.

Second, we can assume that images have *different* gray level multinomial distributions, and therefore estimation collapses to a separate estimation yielding N vectors $\widehat{\boldsymbol{\pi}}_i$.

Perhaps the most attractive approach would be to assume that, on the one hand, images are from same general population, but on the other hand, they have image-specific variation. Such an approach takes an intermediate position between the two extreme approaches.

In this section we model the image through its gray level distribution, and therefore such analysis is content independent; usually, such images are unstructured. For example, using this approach one may come to the conclusion that images of an apple and an orange are the same because they have the same gray level distribution. However, this approach may be useful when a sample of apples is analyzed. The model for an ensemble of structured (content dependent) images is described in Section 12.8.

The volumetric or mean intensity approach, which can be expressed via histogram values as $\sum_{g=0}^{255} g h_g$, is used traditionally to quantify gray images, especially in the analysis of functional MRI data. One may expect the present approach to be more powerful because it is based on an analysis of all 256 histogram values $\{h_g, g = 0, ..., 255\}$.

Application of the theory of mixed models is crucial to the analysis of an ensemble. Modeling hypotheses reduce image analysis to a linear mixed effects model (Chapters 2 to 4), a generalized linear mixed model (Chapter 7), or a nonlinear mixed effects model (Chapter 8).

12.6.1 Fixed-shift model

The fixed-shift model assumes that the ensemble of N gray images $\{\mathbf{M}_i, i = 1, 2, ..., N\}$, have the same gray level distribution up to an image-specific gray level shift. This model, the fixed-shift intensity model, may be viewed as a generalization of the fixed subject-specific intercept model of Section 7.2.2. This is perhaps the simplest model for an ensemble of images; in the next section we consider a more complex random-shift model. One application of this model is when microscopic images of the same homogeneous tissue are taken at different spots and different exposures. If the images were taken at the same exposure, they would produce the same (up to a random deviation) histogram and distribution function, but a nonconstant exposure implies that some images are darker and some lighter.

We assume that all 256 gray levels are modeled; however, this is not a strict requirement. For example, one can safely omit gray levels with zero frequency for all images.

When modeling a series of dependent images, it is more convenient to parameterize the multinomial distribution (12.3) in a different way, as follows:

$$\pi_0 = \frac{1}{1+\sum_{j=1}^{255} e^{\rho_j}},\ \pi_g = \frac{e^{\rho_g}}{1+\sum_{j=1}^{255} e^{\rho_j}},\ g = 1, ..., 255, \tag{12.11}$$

where the $\{\rho_1, ..., \rho_{255}\}$ are new parameters. Obviously, this parameterization implies that all probabilities are positive and the sum is 1. One can easily express the new parameters in terms of the old:

$$\rho_g = \ln \pi_g - \ln \pi_0,\ g = 1, ..., 255, \tag{12.12}$$

so ρ_g can be interpreted as the relative probability on the log scale. There are some advantages to working with ρ_g rather than π_g: (a) representation (12.11) guarantees that all probabilities are positive for any ρ_g, and (b) there is no restriction on $\{\rho_1, ..., \rho_{255}\}$, unlike $\sum_{g=0}^{255} \pi_g = 1$. The fact that $\{\rho_g\}$ may take any value from $-\infty$ to ∞ makes it possible to assume a normal distribution, substantially simplifying the estimation problem, see the next Section. Transformation (12.12) was used by Besag (1974) but without reference to the multinomial distribution. This transformation is the basis for reduction of a nonlinear multinomial model to a linear mixed effects model, Section 12.6.3. For reasons explained later, this transformation will be called *logit*.

In this section we assume that N images have the same gray level distribution but differ by a constant b_i. Thus, on the log scale the ith image is specified by $\{\rho_g + b_i, g = 1, ..., 255\}$. Then, letting $B_i = \exp(b_i)$, we come to the ith multinomial model, as a straightforward generalization of (12.11),

$$\pi_{i0} = \frac{1}{1+B_i\sum_{j=1}^{255} e^{\rho_j}},\ \pi_{ig} = \frac{B_i e^{\rho_g}}{1+B_i\sum_{j=1}^{255} e^{\rho_j}},\ g = 1, ..., 255.$$

Assuming that the N images are independent, the joint log-likelihood function, up to a constant, takes the form $l = \sum_{i=1}^{N}\sum_{g=0}^{255} k_{ig} \ln \pi_{ig}$, where k_{ig} is the frequency in the ith image. In terms of ρ_g, we have

$$l = \sum_{i=1}^{N}\left[(n_i - k_{i0})\ln B_i - n_i \ln\left(1 + B_i\sum_{g=1}^{255} e^{\rho_g}\right)\right] + \sum_{g=1}^{255} d_g\rho_g,$$

where $d_g = \sum_{i=1}^{N} k_{ig}$ is the total frequency of the gth gray level. Our aim is to obtain the maximum likelihood estimators (MLEs) for B_i and ρ_g, as maximizers of l, in closed form. When the $\{\rho_g\}$ are held fixed, we find the maximizer for B_i exactly from the equation $\partial l/\partial B_i = 0$, yielding

$$B_i = \frac{n_i - k_{i0}}{k_{i0}\sum_{g=1}^{255} e^{\rho_g}}, \quad i = 1, ..., N.$$

Plugging this solution back into l, we eliminate nuisance parameters $\{B_i\}$ to obtain a profile log-likelihood function, up to a constant term,

$$l(\rho_1, ..., \rho_{255}) = -(N_T - k_0)\ln\sum_{g=1}^{255} e^{\rho_g} + \sum_{g=1}^{255} d_g\rho_g, \tag{12.13}$$

where $k_0 = \sum_{i=1}^{N} k_{i0}$ is the frequency of the background and $N_T = \sum n_i$ is the total number of pixels. Taking the derivative with respect to ρ_g we finally come to the MLE,

$$\widehat{\rho}_g = \ln \frac{d_g}{N_T - k_0}, \quad g = 1, ..., 255. \tag{12.14}$$

As the reader can see, this solution follows from $e^{\widehat{\rho}_g} = d_g / \sum_{g=1}^{255} d_g$ as a simple estimator of the probability from the total frequency (note that $\sum_{g=1}^{255} e^{\widehat{\rho}_g} = 1$). Consequently, the MLE for the shift is $\widehat{B}_i = n_i/k_{0i} - 1$.

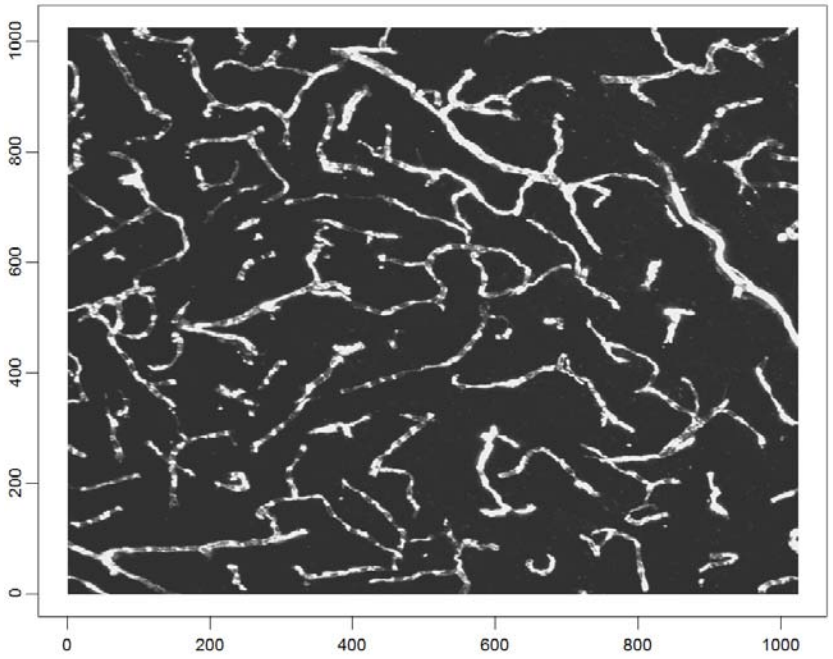

FIGURE 12.8. Typical 1024×1024 gray image of the rat brain. The white bands are vessels filled with oxygenated blood.

12.6.2 Random-shift model

When the number of images (N) is large, a random-shift model may be useful. The similarity to the random intercept model in logistic regression is obvious, see Section 7.2 for a discussion.

Assuming that the random intercepts are iid with normal distribution $b_i \sim \mathcal{N}(0, \sigma^2)$, following the line of Section 7.3, we come to a generalized linear mixed model with a marginal log-likelihood function to maximize

$$l = -\frac{N}{2} \ln \sigma^2 + \sum_{g=1}^{255} d_g \rho_g + \sum_{i=1}^{N} \ln \int_{-\infty}^{\infty} e^{(n_i - k_{i0})b - n_i \ln\left(1 + \sum_{g=1}^{255} e^{b+\rho_g}\right) - \frac{1}{2\sigma^2} b^2} db.$$

Several approaches are available. First, one can use numerical integration to evaluate the integral with a given precision. Second, the FSL approach of Section 7.3.2 can be applied. Third, approximation methods such as methods based on Laplace approximation are straightforward to generalize, see the respective sections of Chapter 7.

Hypoxia BOLD MRI data analysis

We illustrate the fixed-shift model with hypoxia BOLD MRI rat brain data, Dunn et al. (2002). The research was concerned with how a shortage of oxygen affects the brain oxygen concentration shortly after hypoxia. First, the MRI images were derived for eight normal rats before treatment (the control group). Next, the rats were shortly put in a hyperbaric chamber with a below-normal oxygen concentration and the MRI images were repeated right after (the hypoxia group). A typical 1024×1024 gray image before treatment is shown in Figure 12.8; the white bands are vessels filled with oxygen. The null hypothesis is that the hypoxia group has the same oxygen concentration as that of the control group. In Figure 12.9 we show all image data for the two groups. Obviously, the animal variation is substantial and overshadows possible differences between groups.

As in image analysis in general, and in this example in particular, quantification is an important step. How should we quantify oxygen in an image: by vessel count, area, length, density? There exists software, such as NIH Image, that facilitates image segmentation and counts the number of vessels (or, more precisely, distinct objects in the image) automatically. However, we should warn the reader that imaging software is far from perfect, and although "automatic" sounds tempting for this example, it is difficult, if not impossible, to count vessels because we are dealing with brain sections. The vessels are cut at an angle, the same vessels are cut twice, and so on. Instead, we prefer to quantify oxygen by the amount of white color, or more precisely, by the density expressed via the gray level distribution. An advantage of this approach versus vessel count is that it reflects the oxygen concentration in the vessel and therefore may be more representative.

For exploratory statistical analysis, we compare the cumulative gray level distribution functions for each rat group in Figure 12.10. As the reader can see, in general, the images from the control group are lighter suggesting that the amount of oxygen in the hypoxia group is higher (note that if X and Y are two random variables such that $X < Y$, then $F_X(t) > F_Y(t)$, where $F(t)$ is the distribution function). To confirm this statistically, we apply the fixed-shift model of Section 12.6.1. This model seems adequate because the distribution functions for each rat are similar up to a shift. Thus, we (a) analyze each group separately, assuming that rat gray distributions are the same up to a shift b_i, $i = 1, ..., 8$, (b) and compare the resulting $\{\widehat{\rho}_g^{control}\}$ and $\{\widehat{\rho}_g^{hypoxia}\}$ using the χ^2-distribution as in Section 12.4.1.

In Figure 12.11 we show the maximum likelihood estimates of ρ_g computed by formula (12.14). Obviously, the control MR images are darker, and therefore right after hypoxia, oxygen flow into the brain exceeds normal. The χ^2-test confirmed this visual finding with a p-value of less than 0.001.

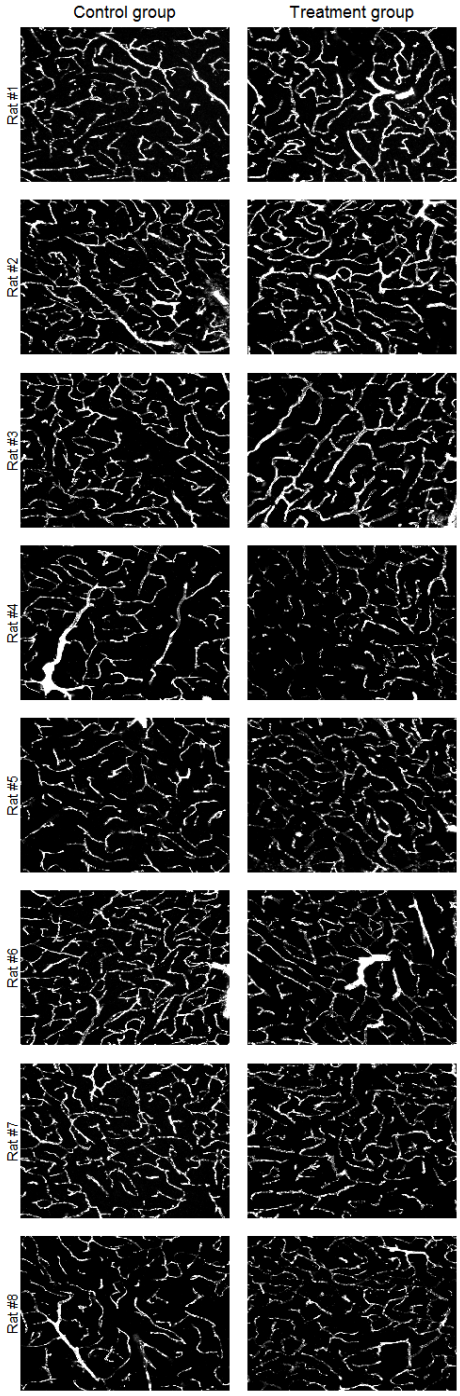

FIGURE 12.9. Rat brain images in two groups of animals (the `R` function `hypoxiaRAT`). The variation across animals is substantial and overshadows the difference between groups. A proper statistical model should address animal heterogeneity. The random-shift mixed model seems to be appropriate.

Below we show the R function that reads the data and plots Figure 12.9.

```
 hypoxiaRAT=function()
{
dump("hypoxiaRAT", "c:\\MixedModels\\Chapter12\\hypoxiaRAT.r")
n <- 1024
x <- 1:n
# we will save the graph in the file
bmp(file="c:\\MixedModels\\Chapter12\\Hypoxia\\hypoxia.bmp",
                          width=500,height=1500)
par(mfrow=c(8,2),mar=c(1,1,1,1),omi=c(0,0.25,0.25,0))
for(i in 1:8)
for(igr in 1:2)
{
    if(igr==1) cc="c" else cc=""
    fn=paste("c:\\MixedModels\\Chapter12\\Hypoxia\\Group",
                          igr,"\\_",i,cc,"_1a_p.pgm",sep="")
    d <- scan(fn,what="")
    d <- matrix(as.numeric(d[12:length(d)]), n, n)
    image(x, x, d, xlab = "", ylab = "", axes = F,
                          col=gray(0:255/255))
    if(igr==1) mtext(side=2,paste("Rat #",i,sep=""),
                          line=0.25,cex=1.25)
    if(i==1 & igr==1) mtext(side=3,"Control group",
                          line=1,cex=1.5)
    if(i==1 & igr==2) mtext(side=3,"Treatment group",
                          line=1,cex=1.5)
}
dev.off() # saving the graph
}
```

12.6.3 Mixed model for gray images

It is straightforward to generalize the random-shift model to a mixed model with a more complex statistical structure. A statistical model for an ensemble of gray images has a hierarchical structure. In a first-stage model, it is assumed that the gray distribution of each image $i = 1, ..., N$ is specified by the multinomial model (12.11) with random, image-specific probabilities

$$\pi_{i0} = \frac{1}{1 + \sum_{j=1}^{255} e^{\tau_{ij}}},\ \pi_{ig} = \frac{e^{\tau_{ig}}}{1 + \sum_{j=1}^{255} e^{\tau_{ij}}},\ g = 1, ..., 255. \tag{12.15}$$

In this model, $\{\tau_{ig}\}$ are random and specified in the linear second-stage model. For example, for the random-shift model of Section 12.6.2, the second-stage model

takes the form $\tau_{ig} = \rho_g + b_i$, where b_i is the random effect. More generally, if $\boldsymbol{\tau}_i = (\tau_{i,1}, ..., \tau_{i,255})'$, the second-stage model in vector form can be expressed as

$$\boldsymbol{\tau}_i = \boldsymbol{\rho} + \mathbf{b}_i, \quad i = 1, ..., N, \tag{12.16}$$

where $\boldsymbol{\rho} = (\rho_1, ..., \rho_{255})'$ defines the population-averaged gray distribution. The error term $\mathbf{b}_i$ is the vector of the random effects and has zero mean and a 255×255 covariance matrix $\mathbf{D}_*$, see Section 7.7. For example, for the random-shift model, we have $\mathbf{b}_i = b_i\mathbf{1}$. Obviously, the covariance matrix of the random effects, $\mathbf{D}_*$, should be structured in a parsimonious way because otherwise the number of distinct elements would be too large to estimate, $256 \times (256 + 1)/2 = 32,896$.

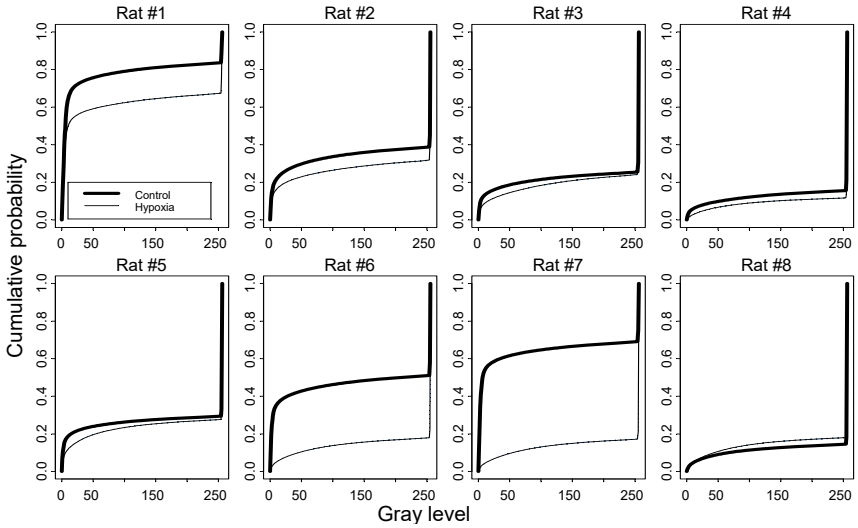

FIGURE 12.10. Cumulative gray-level distributions in two groups for each rat. For most rats the brain images are generally lighter in the control group. We apply the χ^2-distribution to test the overall difference between the groups.

Several covariance structures for $\mathbf{D}_*$ may be suggested:

- Isotropic structure. Matrix $\mathbf{D}_*$ is proportional to the identity matrix, i.e., the off-diagonal elements are zero and all variances are equal.

- Heterogeneous independent structure. The off-diagonal elements are zero, but the diagonal elements are different and unknown. The number of unknown elements in $\mathbf{D}_*$ is 255.

- Toeplitz covariance structure, (11.15). It is assumed that K neighboring gray levels are dependent with $K + 1$ parameters to estimate.

- Band covariance structure such that the covariance between the gth and jth elements of $\mathbf{b}_i$ is zero if $|g - j| > p$. If $p = 0$, we obtain the heterogeneous structure. If $p = 1$, only two neighboring gray levels correlate.

The second-stage model may contain explanatory variables, as did the linear growth curve model of Section 4.1,

$$\boldsymbol{\tau}_i = \mathbf{A}_i\boldsymbol{\rho} + \mathbf{b}_i, \tag{12.17}$$

where $\mathbf{A}_i$ is the design matrix. For example, if two ensembles are compared $\mathbf{A}_i\boldsymbol{\rho} = \boldsymbol{\rho}_1$ if the image is from the first ensemble, and $\mathbf{A}_i\boldsymbol{\rho} = \boldsymbol{\rho}_1 + \boldsymbol{\delta}$ if the image is from the second ensemble. Then, the two ensembles are the same if $H_0 : \boldsymbol{\delta} = \mathbf{0}$. Model (12.17) can incorporate image differences due to gender, age, etc. In functional MRI it may reflect the time when a stimulus occurred.

The statistical model for gray levels specified by (12.15) and (12.17) belongs to the family of generalized linear mixed models studied in Chapter 7. Exact methods of estimation may be computationally intensive, especially when the dimension of the random effect is 255.

12.6.4 Two-stage estimation

It is attractive to apply the two-stage estimation for an ensemble because (a) gray probabilities are random and image-specific, and (b) individual estimates are easy to obtain. The two-stage estimation approach was applied to a nonlinear mixed model in Section 8.5. As follows from (12.4), the individual probability estimate is equal to the histogram value, h_{ig}. Hence, assuming that the image background is black (h_{i0} prevails), we compute logits

$$t_{ig} = \ln h_{ig} - \ln h_{i0}, \quad g = 1, ..., 255 \tag{12.18}$$

as individual estimates of $\{\tau_{ig}\}$ in model (12.15). Having estimates (12.18), we substitute $\boldsymbol{\tau}_i$ with $\mathbf{t}_i = (t_{i,1}, ..., t_{i,255})'$ in model (12.17) and arrive at the second-stage model $\mathbf{t}_i = \mathbf{A}_i\boldsymbol{\rho} + \boldsymbol{\eta}_i$. Notice that the error term $\boldsymbol{\eta}_i$ differs from $\mathbf{b}_i$ because $\mathbf{t}_i$ is an estimate of $\boldsymbol{\tau}_i$, and consequently, the variance of $\boldsymbol{\eta}_i$ is larger than that of $\mathbf{b}_i$. Since the covariance matrix of individual probability estimates has an exact form, (12.5), we can approximate the covariance of $\mathbf{t}_i$ by the delta-method, as is realized below.

Proposition 47 *Let $\widehat{\boldsymbol{\pi}}$ be a 256×1 vector estimate with the variance-covariance matrix specified by (12.5). Let $t_g = \ln \widehat{\pi}_g - \ln \widehat{\pi}_0$, $g = 1, ..., 255$ be components of vector $\mathbf{t}$. Then*

$$\operatorname{cov}(\ln \pi_g - \ln \pi_0, \ln \pi_j - \ln \pi_0) \simeq \frac{1}{PQ} \begin{cases} 1 + 1/\pi_0 + 1/\pi_g & \text{if } g = j \\ 1 + 1/\pi_0 & \text{if } g \neq j \end{cases}$$

or, in matrix form,

$$\operatorname{cov}(\mathbf{t}) \simeq \frac{1}{PQ}\left[(1 + \pi_0^{-1})\mathbf{1}\mathbf{1}' + \mathbf{D}_1^{-1}\right],$$

where $\mathbf{D}_1 = diag(\pi_1, ..., \pi_{255})$.

Proof. We use the delta-method to approximate the covariance matrix of $\mathbf{t} = (\ln \widehat{\pi}_1 - \ln \widehat{\pi}_0, ..., \ln \widehat{\pi}_{255} - \ln \widehat{\pi}_0)'$. Letting $\boldsymbol{\pi}_1 = (\pi_1, ..., \pi_{255})'$ and $\mathbf{D}_1 = \operatorname{diag}(\boldsymbol{\pi}_1)$

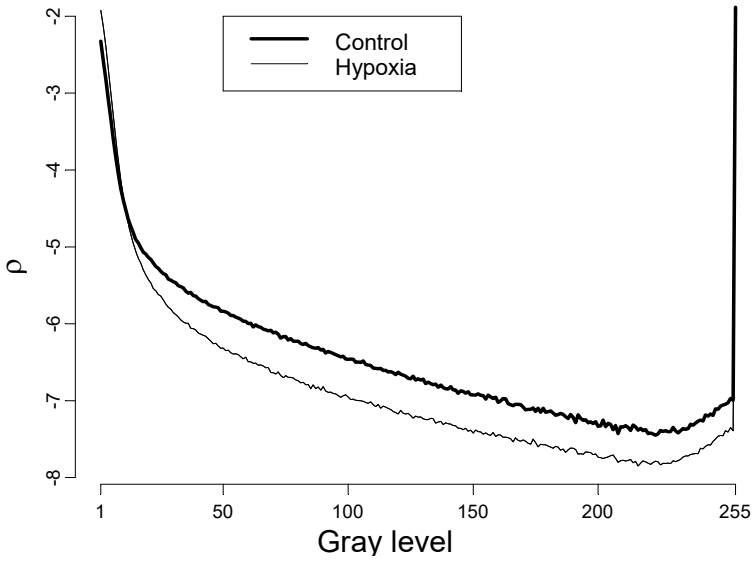

FIGURE 12.11. Maximum likelihood estimates of ρ_g for $g = 1, ..., 255$, for the fixed-shift model, (12.14). Obviously, the MR images of the control group are lighter than those of the hypoxia group. Consequently, the damage to brain vesels by hypoxia is statistically significant.

we express $\partial \mathbf{t}/\partial \widehat{\boldsymbol{\pi}} = [-\pi_0^{-1}\mathbf{1}; \mathbf{D}_1^{-1}]$, a 255×256 derivative matrix. The 256×256 covariance matrix of $\widehat{\boldsymbol{\pi}}$ can be partitioned as

$$\mathbf{C} = \begin{bmatrix} \pi_0(1-\pi_0) & -\pi_0 \boldsymbol{\pi}_1' \\ -\pi_0 \boldsymbol{\pi}_1 & \mathbf{D}_1 \end{bmatrix}.$$

By the delta-method, the covariance matrix $\mathbf{t}$ can be approximated as

$$\begin{aligned}(\partial \mathbf{t}/\partial \widehat{\boldsymbol{\pi}})\mathbf{C}(\partial \mathbf{t}/\partial \widehat{\boldsymbol{\pi}})' &= [-\pi_0^{-1}\mathbf{1}; \mathbf{D}_1^{-1}] \begin{bmatrix} \pi_0(1-\pi_0) & -\pi_0 \boldsymbol{\pi}_1' \\ -\pi_0 \boldsymbol{\pi}_1 & \mathbf{D}_1 \end{bmatrix} \begin{bmatrix} -\pi_0^{-1}\mathbf{1}' \\ \mathbf{D}_1^{-1} \end{bmatrix} \\ &= \pi_0^{-2}\pi_0(1-\pi_0)\mathbf{1}\mathbf{1}' + \pi_0^{-1}\mathbf{1}\pi_0\boldsymbol{\pi}_1'\mathbf{D}_1^{-1} + \mathbf{D}_1^{-1}\pi_0\boldsymbol{\pi}_1\pi_0^{-1}\mathbf{1}' + \mathbf{D}_1^{-1}\mathbf{D}_1\mathbf{D}_1^{-1} \\ &= \pi_0^{-1}(1-\pi_0)\mathbf{1}\mathbf{1}' + 2\mathbf{1}\mathbf{1}' + \mathbf{D}_1^{-1} = (1+\pi_0^{-1})\mathbf{1}\mathbf{1}' + \mathbf{D}_1^{-1}.\end{aligned}$$

■

Applying this result and assuming that the $\{\mathbf{t}_i\}$ have a normal distribution, we arrive at the linear model for logits,

$$\mathbf{t}_i \sim \mathcal{N}(\mathbf{A}_i\boldsymbol{\tau}, \mathbf{D}_* + \mathbf{T}_i), \quad i = 1, ..., N, \tag{12.19}$$

where matrix $\mathbf{T}_i$ is fixed and given by

$$\mathbf{T}_i = \frac{1}{n_i}\left[(1 + h_{i0}^{-1})\mathbf{1}\mathbf{1}' + \mathbf{D}_{i1}^{-1}\right]$$

and $n_i = P_iQ_i$ is the number of pixels and $\mathbf{D}_{i1}$ =diag$(h_{i1}, ..., h_{i,255})$ is the diagonal matrix of histogram values. In a special case without explanatory variables, (12.16),

we have $\mathbf{A}_i = \mathbf{I}$. The unknown parameters are $\boldsymbol{\tau}$ and $\mathbf{D}_*$. If $\mathbf{D}_*$ were known, vector $\boldsymbol{\tau}$ could be estimated by Generalized Least Squares (GLS),

$$\widehat{\boldsymbol{\tau}} = \left(\sum_{i=1}^{N} \mathbf{A}_i'(\mathbf{D}_* + \mathbf{T}_i)^{-1}\mathbf{A}_i \right)^{-1} \left(\sum_{i=1}^{N} \mathbf{A}_i'(\mathbf{D}_* + \mathbf{T}_i)^{-1}\mathbf{t}_i \right)$$

with covariance matrix

$$\operatorname{cov}(\widehat{\boldsymbol{\tau}}) = \left(\sum_{i=1}^{N} \mathbf{A}_i'(\mathbf{D}_* + \mathbf{T}_i)^{-1}\mathbf{A}_i \right)^{-1}.$$

For model (12.16), we have

$$\widehat{\boldsymbol{\tau}} = \left(\sum_{i=1}^{N} (\mathbf{D}_* + \mathbf{T}_i)^{-1} \right)^{-1} \left(\sum_{i=1}^{N} (\mathbf{D}_* + \mathbf{T}_i)^{-1}\mathbf{t}_i \right).$$

Matrix $\mathbf{D}_*$ may be estimated either by maximum likelihood or by the method of moments for the linear growth curve model of Section 4.1. For example, for model (12.16) in the heterogeneous case, compute $s_g^2 = N^{-1}\sum_i (t_{ig} - \overline{t}_g)^2$ and estimate $\mathbf{D}_*$ as $\mathbf{S} - N^{-1}\sum_{i=1}^{N} \mathbf{T}_i$, where $\mathbf{S}$ is the diagonal matrix with the (g,g)th element s_g^2. If matrix $\widehat{\mathbf{D}}_*$ is not positive definite, we apply the projection procedure described in Section 2.15.2.

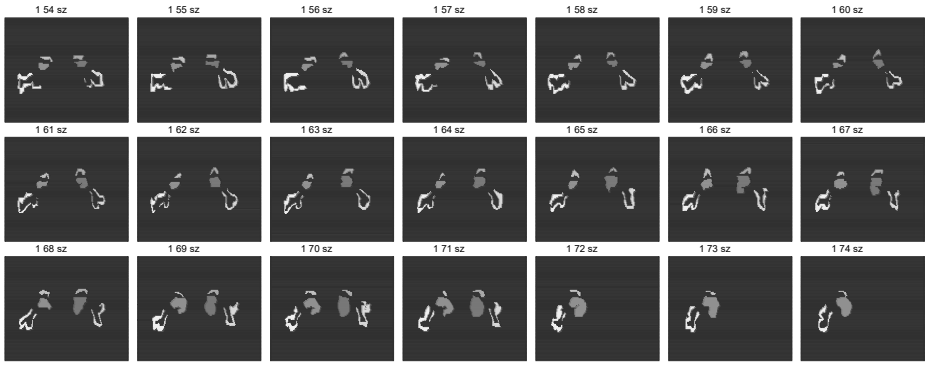

FIGURE 12.12. Typical MR frame images of the amygdala–hippocampal complex of a schizophrenia patient.

12.6.5 Schizophrenia MRI analysis

Schizophrenia is a major mental disorder and is characterized by impaired thinking and hallucinations. It affects about 1% of the general population. Advances in Computerized Tomography (CT) and magnetic resonance imaging (MRI) created a new dimension for brain research, including for schizophrenia (Shenton, 2001). The

existing approaches to studying the human brain and its abnormalities using MRI data can be roughly classified into three groups: (1) shape, (2) asymmetry, and (3) volumetric analysis. Here we focus on the latter approach. In the volumetric approach, the organ or region of interest is quantified by one number (Godszal and Pham, 2000). In our approach, we analyze all nonzero gray levels, and therefore the analysis may be more powerful.

In this section we analyze the temporal lobe, or more precisely the amygdala-hippocampal complex, with the MRI data kindly provided by Dr. M. Shenton of Harvard Medical School. The study design description and the appropriate statistical analysis are given in the original article by Shenton et al. (1992). The MRI brain data consist of equidistant frames for 15 schizophrenia patients and 15 matched normal controls. In Figure 12.12 MR frame images of the amygdala–hippocampal complex of the first schizophrenia patient are shown. In Figure 12.13, image frames are shown for the first normal control. All images have the same size, $n_i = P_i Q_i$ =const, $i = 1, ..., 30 = N$. We want to determine whether the amygdala-hippocampal complex of the schizophrenia patients and that of controls is different.

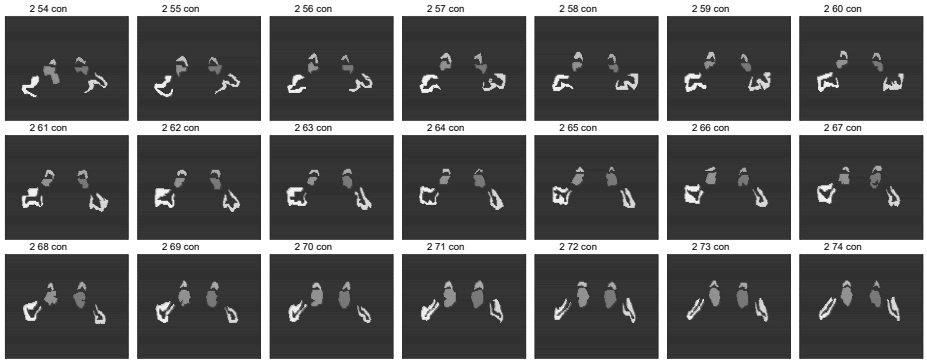

FIGURE 12.13. Typical images of the amygdala-hippocampal complex of a normal control.

The R function below reads 39 cases and plots the MRI frames from 41 to 61 as a 3×7 panel plot (the first 15 cases are controls and the remainder belong to schizophrenia patients).

```
schiz=function ()
{
 dump("schiz", "c:\\MixedModels\\Chapter12\\schiz.r")
 cc = "c:\\MixedModels\\Chapter12\\schiz\\case"
 for (i in 1:30) {
 par(mfrow = c(3, 7), mar = c(1, 1, 1, 1),omi=c(0,0,.25,0))
 for (j in 1:21) {
 d = scan(paste(cc, i, "\\case", i, ".0", j + 50,".pgm",
                        sep = ""), what = "", quiet = T)
```

```
 dm = matrix(as.numeric(d[12:length(d)]), nrow = 256, ncol = 256)
 image(1:256, 1:256, dm, col = gray(0:255/255),xlab="",ylab="",
                       axes=F)
 mtext(side=3,paste("Frame",j+50),line=.2,cex=.75)
 }
 mtext(side=3,paste("Case",i),outer=T,cex=1.25,line=.25)
 }
}
```

Each image has only eight different gray levels: 0, 95, 127, 159, 191, 212, 223, and 255 (0 is the black background), so instead of 256 gray levels, we have 8. We start the analysis by computing the logits $t_{ig} = \ln(h_{ig}/h_{i0})$, where h is the histogram value and $i = 1, ..., 30$, $g = 1, ..., 7$. Then we plot the mean logit for each gray level in each group; see Figure 12.14. This plot reveals that for gray levels 95, 127, 159,

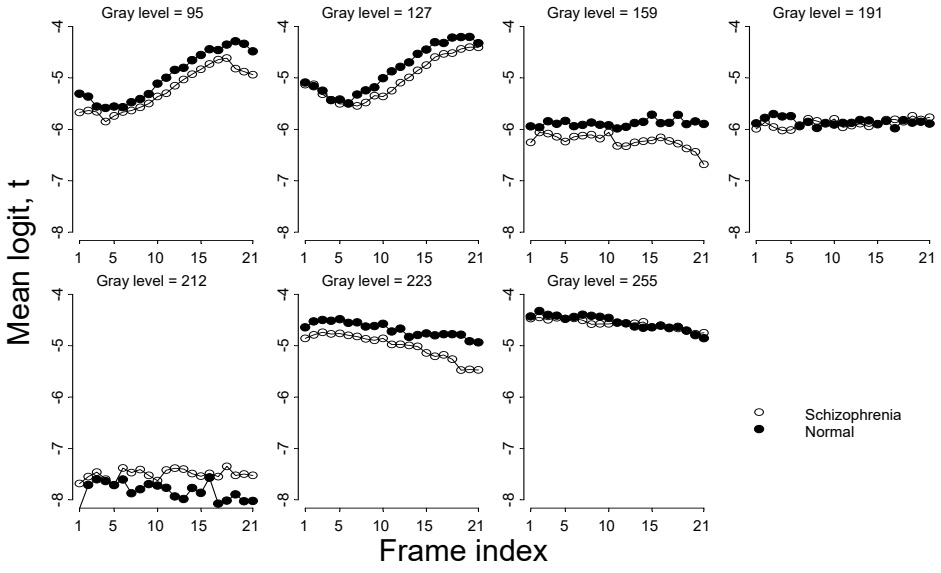

FIGURE 12.14. Mean logit as a function of the frame index for different gray levels in two groups. On average, normals have a higher logit.

and 223 the logits for normals are higher than those for controls. This finding hints of the following linear statistical model for logits

$$\begin{aligned} t_{ijfg} &= \beta_{gf} + \delta(j-1) + b_{ijfg}, \\ i &= 1, 2, ..., 15;\ j = 1, 2;\ f = 1, 2, ..., 21;\ g = 1, 2, 3, 4. \end{aligned} \tag{12.20}$$

where i is the patient index in group j ($j = 1$ codes a schizophrenia patient and $j = 2$ codes a normal control); f is the MRI frame index; and g is the gray level index (gray levels are: 95, 127, 159, and 223). As follows from this model, the logits for each gray level and frame are different, with the mean β_{gf} and the difference

between the two groups constant, δ. Assuming that the $\{b_{ijfg}\}$ are iid, we can estimate (12.20) by simple regression analysis, yielding the OLS estimate $\widehat{\delta} = 0.129$ with the t-statistic 9.79 and p-value < 0.00001. The difference in gray levels 95, 127, 159, and 223 for schizophrenia and controls is significant. Model (12.20) can be modified in many ways. For example, one may consider heterogeneous or dependent logits. Also, the logit model may serve as a diagnostic tool to identify schizophrenia using the membership test of Section 3.8.1.

Model (12.20) is easy to apply to compare two samples of images, as a generalization of the standard t-test, where $H_0 : \delta = 0$.

Problems for Section 12.6

1. Develop a for zero-shift test, $H_0 : b_1 = b_2 = ... = b_N = 0$ in the framework of a fixed-shift model.

2*. Develop a maximum likelihood estimation algorithm for maximization l from Section 12.6.2. Write an `R` function that implements the GH quadrature. Apply Laplace approximation, as discussed in Chapter 7, to approximate the log-likelihood. Write an `R` function and compare its performance against GH quadrature MLE.

3*. Plot logits and cumulative distribution functions for the rat hypoxia data for the two groups; reproduce Figure 12.10. Compute a two-stage estimate and reproduce the result depicted in Figure 12.11. Use the approximate covariance matrix to test the statistical significance of the treatment effect.

4*. Generalize the two-stage estimation procedure for two groups with different logit parameter δ, the treatment effect. Generalize methods developed in the previous problem to estimate the treatment effect and apply them to the rat hypoxia data. Use the likelihood-ratio test and the Wald test based on the covariance matrix in Proposition 47.

5. The function `bioimage` reads 28 PGM histology image files and plots them as depicted in Figure 12.15. Use the mixed model for grayscale intensities from Section 12.6.3 to compare the treatment effect (the more living cells, the darker the image, see Figure 12.3). Compute the pairwise p-values for image comparison. Is there a synergy between radiation and drug? Use formula (10.17) to estimate the synergy.

6. Modify the function `schiz` to plot 30 cdfs for 15 controls and 15 schizophrenia patients (use the built-in function `cumsum` and use different colors for the two groups). Do the same but for the mean on the logit scale as in Figure 12.14. Apply model (12.20) to discriminate controls from patients. Develop a mixed model with a subject-specific intensity level. Test whether the heterogeneity variance is statistically significant.

12.7 Image alignment and registration

Earlier, we dealt with an image gray level distribution that is content-independent. Starting in this section, we consider content-dependent images. Before doing statistical analysis or image comparison, images must be at the same scale, and consequently, they must be properly aligned and rotated (registered). We use the terms *alignment* and *registration* as synonyms, although sometimes alignment is used when only translation is allowed. For example, if one wants to know the difference between

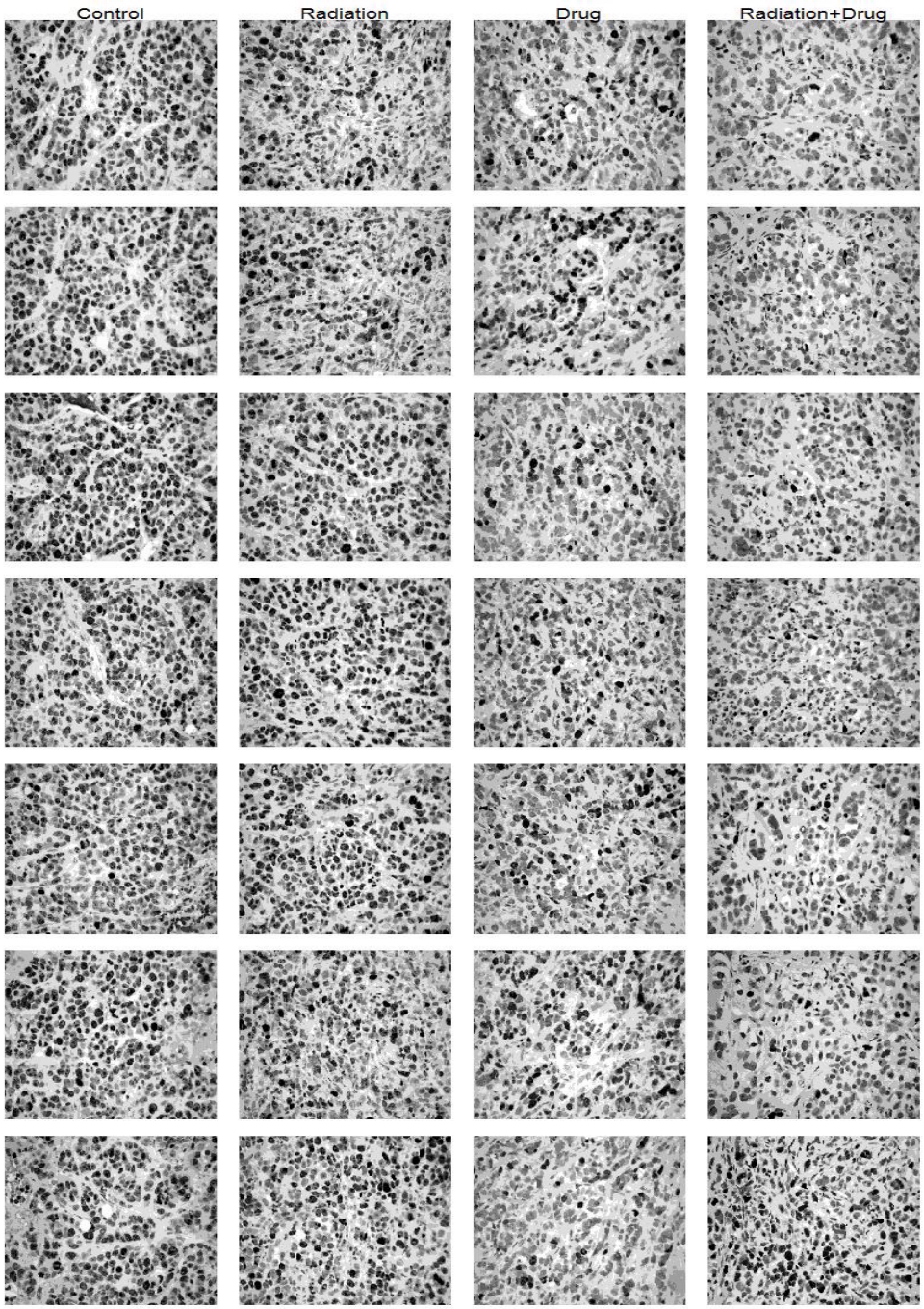

FIGURE 12.15. Twenty eight histology images in four cancer treatment groups ('control' means no treatment). This plot was created by the `R` function `bioimage.` Are the differences in the treatments statistically significant?

images (as in the before-and-after analysis), it is tempting to take a pixel-by-pixel difference. The problem is that usually a pixel on one image does not exactly correspond to a pixel on another image. An example of slightly different images is shown in Figure 12.16. Before taking the difference, the images must be aligned (we describe the detail of alignment in Section 12.7.8). Remember that we faced a similar problem with shapes where they must be re-sized and rotated, see Section 11.6.

Image registration is a frequently used technique, especially in medical applications, particularly in the context of brain and medical imaging. See the survey literature by Maintz and Viergever (1998) and a collection of papers in the book by Hajnal et al. (2001). Several commercial software packages for image registration are available, including Automated Image Registration (AIR: http://bishopw.loni.ucla.edu/AIR5) and Statistical Parametric Mapping (SPM: http://www.fil.ion.ucl.ac.uk/spm). The same methodology is also used for image *coregistration,* when images are of the same object or region but derived by different imaging techniques, such as MRI and PET (Kiebel et al., 1997; Ashburner and Friston, 1997).

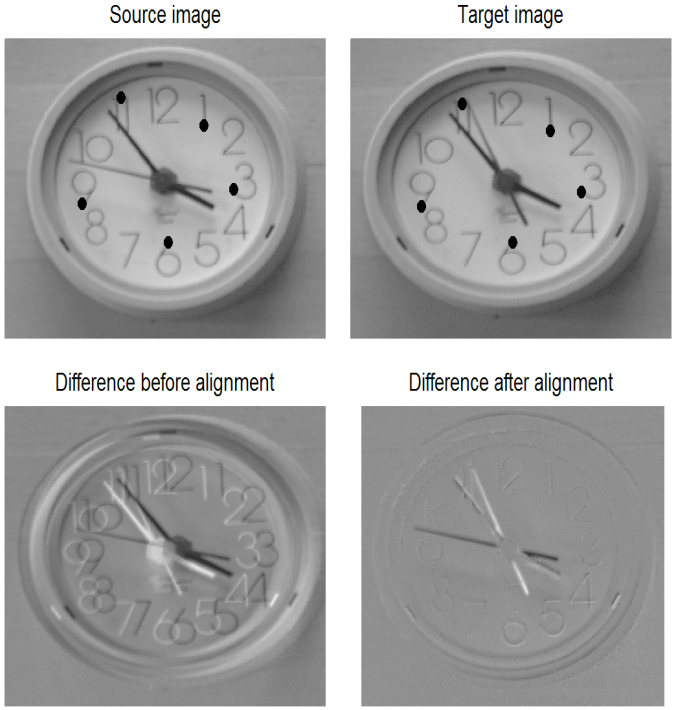

FIGURE 12.16. Two images taken a few seconds apart and their difference before and after alignment. Before computing a pixel-by-pixel difference, the source and target images must be aligned. The first image is 525×504 and the second image is 483×508. Points on the images (black dots) serve as the landmarks for the alignment.

The purpose of this section is to introduce the problem of image registration and discuss several cost functions (registration criteria). To account for coordinate-specific image transformation, random registration is introduced. We show that

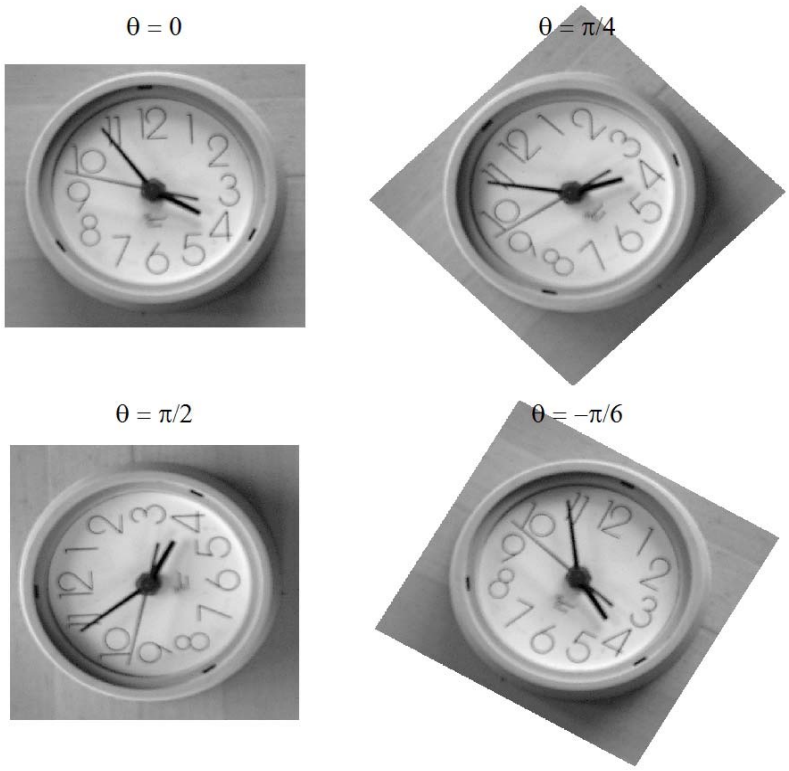

FIGURE 12.17. Rigid transformation of the clock image (rotation by angle θ). This figure was generated by the `R` function `clockROT`.

random registration can be studied in the framework of the nonlinear mixed effects model of Chapter 8. To simplify, we consider only planar images.

12.7.1 Affine image registration

Several aspects should be taken into account when two images have to be co-registered: (a) they may have different mean intensity; (b) they may have different scale intensity; (c) the registration may be linear or nonlinear; (d) rotation may be allowed or not allowed; (e) and the transformation may be rigid or general affine (linear). An important step in image registration is the choice of the criterion or cost function (Woods et al., 1998a,b).

Let $\mathbf{M}_i = \{M_i(p,q), p = 1, ..., P_i, q = 1, ..., Q_i\}$ be two images, $i = 1, 2$. Sometimes, $\mathbf{M}_1$ is referred to as the source and $\mathbf{M}_2$ as the target image. We want to find an affine (linear) transformation such that the mean sum of squares,

$$S = \frac{1}{|\mathcal{M}|} \sum_{(p,q)\in\mathcal{M}} \left[M_1(p,q) - \nu M_2(\beta_1 + \beta_2 p + \beta_3 q, \beta_4 + \beta_5 p + \beta_6 q) - \mu\right]^2, \quad (12.21)$$

is minimum. Sometimes (12.21) is referred to as the mean-squared error (MSE) criterion or cost function. Eight parameters are to be determined: the intensity shift (μ), the scale intensity (ν), and the vector of affine parameters, $\boldsymbol{\beta} = (\beta_1, ..., \beta_6)'$. Integer indices p and q must satisfy the following restrictions: $1 \le p \le P_1$, $1 \le q \le Q_1$, $1 \le \beta_1 + \beta_2 p + \beta_3 q \le P_2$, $1 \le \beta_4 + \beta_5 p + \beta_6 q \le Q_2$. Apparently, indices for the target image must be rounded to the nearest integer unless the values are interpolated. The set of pairs (p, q) that satisfy these restrictions constitutes the set $\mathcal{M}$, so the summation in (12.21) is over all pairs $(p, q) \in \mathcal{M}$. Since the number of terms in the sum depends on the affine parameters, we normalize it dividing by the number of elements in the set, $|\mathcal{M}|$. Therefore, S is the mean-squared error (MSE) criterion . An implicit assumption of (12.21) is that the variance is constant, otherwise a weighted MSE criterion should be used. Parameters β_2 and β_6 control image zooming or shrinkage; parameters β_3 and β_5 control rotation; parameters β_1 and β_4 control image translation (shift). An advantage of affine transformation is that no restrictions are imposed on $\boldsymbol{\beta}$. If one wants to allow only translation and rigid transformation (rotation and resizing), six parameters are reduced to four by letting $\beta_2 = \beta_6$ and $\beta_3 = -\beta_5$. If the sizes of the source and target images are the same, we come to a nonlinear problem because the coordinates of the target image are $\beta_1 + p\cos\theta + q\sin\theta$ and $\beta_4 - p\sin\theta + q\cos\theta$, where θ is the (clockwise) rotation angle, see Figure 12.17 for an illustration. As with shapes, the choice of transformation should be dictated by the way in which images are sampled. If images have the same or close intensity distribution (histogram), we may assume that $\nu = 1$ and $\mu = 0$, which leads to a simpler MSE.

As is easy to see, criterion S is roughly equivalent to maximization of the correlation coefficient between the two images. Indeed, assuming that the affine parameters are fixed, the minimum of (12.21) in a vector form is attained at the least squares solution for ν and μ, or more precisely,

$$\min_{\nu,\mu} \sum (m_{1i} - \nu m_{2i} - \mu)^2 = (1 - r^2) \sum (m_{1i} - \overline{m}_1)^2$$
$$= \frac{[\sum (m_{1i} - \overline{m}_1)(m_{2i} - \overline{m}_2)]^2}{\sum (m_{2i} - \overline{m}_2)^2}, \tag{12.22}$$

where r^2 is the squared correlation coefficient. This identity implies that instead of S, one can maximize (12.22), with μ and ν eliminated and with m_{1i} substituted for by $M_1(p, q)$ and m_{2i} substituted for by $M_2(\beta_1 + \beta_2 p + \beta_3 q, \beta_4 + \beta_5 p + \beta_6 q)$. However, unlike the MSE, (12.22) is not a quadratic function of parameters even after linearization. To avoid this nonlinearity, one can alternate between the MSE minimization and ν and μ estimation by ordinary least squares. A statistical model-based approach to image registration allows the testing of various hypotheses regarding the type of registration, e.g. whether images are statistically indifferent up to a rigid transformation, see more details in a recent paper by Demidenko (2009).

12.7.2 Weighted sum of squares

From a statistical point of view, registration criterion (12.21) implicitly assumes that the variance of the difference between the two images is the same (gray level independent). Consequently, white will dominate because it has magnitude around

255, and black, 0. We can modify the unweighted criterion, S, by adjusting for the gray level variance, (12.5). Let h_g be the image histogram so that the variance of M is proportional to $h_M(1-h_M)$. Assuming that the two images are independent, we arrive at the weighted criterion:

$$S_w = \frac{1}{|\mathcal{M}|} \sum_{(p,q)\in\mathcal{M}} \frac{[M_1(p,q) - \nu M_2(\beta_1 + \beta_2 p + \beta_3 q, \beta_4 + \beta_5 p + \beta_6 q) - \mu]^2}{w(p,q)} \tag{12.23}$$

with the weight $w(p,q) = h_{M_1}(1-h_{M_1}) + h_{M_2}(1-h_{M_2})$ as the variance of the difference, $M_1 - M_2$. This weight is parameter independent and can be computed beforehand.

In an alternative approach, we assume that the gray values follow the Poisson distribution, see Section 7.5.1. Since for this distribution the mean equals th evariance, we come to the weight $w(p,q) = M_1(p,q) + M_2(p,q)$.

12.7.3 Nonlinear transformations

Affine transformation may be extended to nonlinear transformation: for example, to account for individual variation. Perhaps the easiest nonlinear transformation is a quadratic transformation of the form $\gamma_1 + \gamma_2 p + \gamma_3 q + \gamma_4 p^2 + \gamma_5 q^2 + \gamma_6 pq$. An advantage of this transformation is that it is still linear in parameters. Polynomials of a higher order can be employed, such as in the package AIR mentioned above.

12.7.4 Random registration

The image registrations discussed so far are rigid because the coordinate system of the target image is expressed as a function of the coordinates of the source image. In real image comparison and registration, the coordinate system may be randomly deformed, as illustrated in Figure 12.18. The aim of this section is to show how random registration can be described in the framework of the nonlinear mixed effects model studied in Chapter 8.

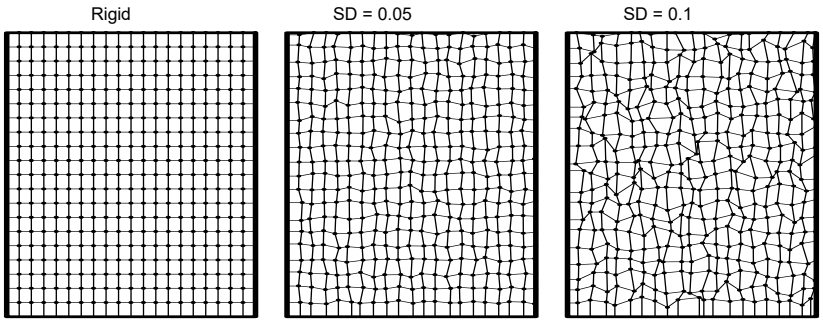

FIGURE 12.18. Regular (rigid) and random registration mapping. The strength of the coordinate system deformation $(p+b_p, q+b_q)$ depends on the standard deviation of random b_p and b_q.

In random registration, we assume that the affine vector $\boldsymbol{\beta}$ is random and location-specific. The statistical model for random registration is written in a hierarchical fashion. To simplify, we shall assume that the identity for the scale intensity, $\nu = 1$, and the intensity shift, $\mu = 0$. In the first-stage model, we express the source through the target image with random coordinates as

$$M_1(p,q) = M_2(\beta_{1pq} + \beta_{2pq}p + \beta_{3pq}q, \beta_{4pq} + \beta_{5pq}p + \beta_{6pq}q) + \varepsilon(p,q), \qquad (12.24)$$

where $\boldsymbol{\beta}_{pq} = (\beta_{1pq}, \beta_{2pq}, \beta_{3pq}, \beta_{4pq}, \beta_{5pq}, \beta_{6pq})'$ is a six-dimensional random vector and ε is a random variable with zero mean. It is assumed that the random affine vector $\boldsymbol{\beta}_{pq}$ and the error term $\varepsilon(p,q)$ are independent. In the second-stage model, we specify the affine vector as a random vector with unknown means,

$$\boldsymbol{\beta}_{pq} = \boldsymbol{\beta} + \mathbf{b}_{pq}, \quad p = 1, ..., P, q = 1, ..., Q, \qquad (12.25)$$

where $\mathbf{b}_{pq}$ is a vector of random effects with zero mean. Following the line of mixed model terminology, $\boldsymbol{\beta}_{pq}$ is local coordinate-specific and $\boldsymbol{\beta}$ is a global or population-averaged vector of affine parameters. Several assumptions can be made regarding the covariance structure. For example, to address spatial correlation, one may assume that neighboring elements correlate following the planar autoregression or Markov random field scheme

$$b_{h,pq} = \sum_{j=-J}^{J} \sum_{k=-K}^{K} \alpha_{jk} b_{h,p+j,q+k} + \eta_{h,pq}, \quad h = 1, 2, ..., 6. \qquad (12.26)$$

This model can be approximated parsimoniously with a Toeplitz covariance matrix as in Section 4.3.4; see more details in Section 12.9.1.

In matrix form, we combine the models (12.24) and (12.25) into one as

$$\mathbf{M}_1 = \mathbf{M}_2(\boldsymbol{\beta} + \mathbf{b}) + \boldsymbol{\varepsilon}. \qquad (12.27)$$

This is a nonlinear mixed effects model where $\mathbf{M}_1$ is treated as data and $\mathbf{M}_2$ as a nonlinear function. Although $\mathbf{M}_2$ is a discrete function (matrix), we can assume that the size of this matrix is large enough to treat it as a continuous function of coordinates. To meet this assumption, interpolation methods may be applied. As follows from Sections 8.7 and 8.8.2, assuming normal distribution for the error term and random effects, Laplace approximation leads to minimization of the penalized sum of squares

$$\|\mathbf{M}_1 - \mathbf{M}_2(\boldsymbol{\beta} + \mathbf{b})\|^2 + \mathbf{b}'\mathbf{V}^{-1}\mathbf{b} \Rightarrow \min_{\nu,\mu,\boldsymbol{\beta},\mathbf{b}}, \qquad (12.28)$$

where $\mathrm{cov}(\mathbf{b}) = \sigma^2\mathbf{V}$ and $\mathrm{cov}(\boldsymbol{\varepsilon}) = \sigma^2\mathbf{I}$. Several approximate methods to estimate affine global parameters and covariance matrix $\mathbf{V}$ are suggested in Chapter 8.

12.7.5 Linear image interpolation

So far, we have assumed that the target image is evaluated at the rounded values $\beta_1 + \beta_2 p + \beta_3 q$ and $\beta_4 + \beta_5 p + \beta_6 q$. A more precise evaluation is based on image interpolation. Several methods exist, including B-splines (Gonzalez and Woods, 2002). Here we consider the simplest one—linear interpolation.

Mathematically, the problem is formulated as follows. Let a grayscale image be given by the matrix $\mathbf{M} = \{M(p,q), p = 1, ..., P, q = 1, ..., Q\}$. We want to approximate $M(p+\alpha, q+\beta)$ using the M-values at four neighboring points $\{M(p+s, q+s), s = 0, 1, t = 0, 1\}$, where, without loss of generality, α and β are positive and less than 1. To interpolate, we split the unit square into two triangles and then use the linear interpolation on each of them (sometimes this process is called triangulation). The are two ways to split the square into two triangles—dividing by the diagonal (0,1)–(1,0) or (0,0)–(1,1); to be specific, we take the latter. It is elementary to find that the triangular interpolation formula is given by

$$M(p+\alpha, q+\beta) = M(p,q) \tag{12.29}$$
$$+ \begin{cases} [M(p+1,q+1) - M(p,q+1)]\alpha + [M(p,q+1) - M(p,q)]\beta \text{ if } \beta > \alpha \\ [M(p+1,q) - M(p,q)]\alpha + [M(p+1,q+1) - M(p+1,q)]\beta \text{ if } \beta \le \alpha \end{cases}$$

It is elementary to check that this interpolation is continuous and consists of two planes, with the edge at the main diagonal. An example of the triangular interpolation is shown in Figure 12.19 (see the function `imageLI`). Image interpolation is useful when several images of different size are aligned simultaneously, as in Section 12.8.1. To indicate that an image is linearly interpolated, we use a tilde, $\tilde{}$.

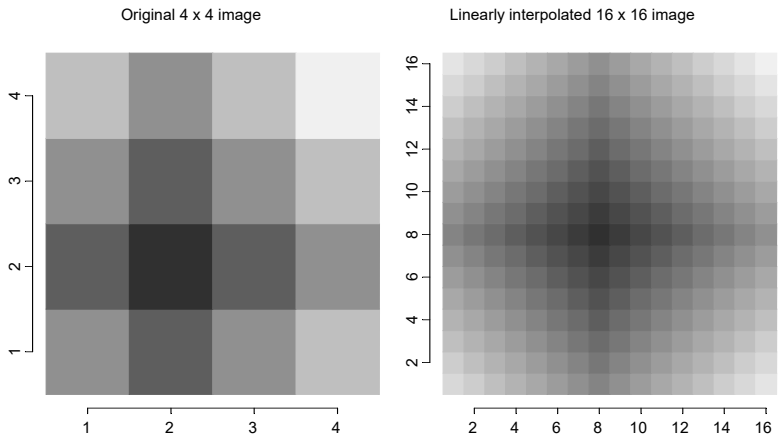

FIGURE 12.19. The original 4×4 image and 16×16 image linearly interpolated by formula (12.29).

12.7.6 Computational aspects

Two issues complicate the minimization of the unweighted or weighted sum of squares of residuals: (a) we deal with discrete minimization (indices are integers), and (b) the function S may have several local minima. If images have different sizes, an interpolation may be applied to reduce the discrete nature (Gonzalez and Woods, 2002). To exclude false minima, several starting points should be used to see whether the minimization converges to the same minimum, see Appendix 13.3 for

a general discussion. Below we discuss how to minimize the sum of squares without derivatives.

12.7.7 Derivative-free algorithm for image registration

In the continuous optimization problem $F(\mathbf{x})$, where $\mathbf{x}$ is a vector with continuous elements, the derivative plays the central role. Indeed, many maximization algorithms can be expressed via the update formula $\mathbf{x}_{s+1} = \mathbf{x}_s + \lambda_s \mathbf{H}_s^{-1}\mathbf{g}_s$, where $\mathbf{H}_s$ is a positive definite matrix, $\mathbf{g}_s = \partial F(\mathbf{x} = \mathbf{x}_s)/\partial \mathbf{x}$ is the derivative, λ_s is the step length (typically, $\lambda_s = 1$), and $s = 0, 1, \ldots$. Sometimes, computation of the derivative is very complex or even not feasible. Furthermore, in some instances, the derivative does not exist. For those who code in FORTRAN, there is a programming technique that generates a subroutine for derivative computation, *ADIFOR* (Bischof et al., 1992).

Now we discuss a derivative-free optimization technique, or more specifically, a derivative-free approach to the sum of squares minimization in nonlinear regression suggested originally by Ortega and Rheinboldt (1970). Later, Ralston and Jennrich (1978) reinvented the algorithm and called it *DUD*. This algorithm may be useful when the argument is not continuous but takes integer values as in the image registration criteria discussed above. An important by-product result is that this algorithm generates the covariance matrix for the registration parameters so that various statistical hypotheses may be tested; for example, if the transformation is rigid. Parameters ν and μ may be eliminated, but then the criterion function becomes nonsquare, (12.22). Otherwise, one can alternate between S and S_w minimization, when ν and μ are held, finding ν and μ from linear least squares after the transformation parameters are determined.

The derivative-free algorithm DUD is described as follows. We want to minimize the sum of squares (SS) of residuals,

$$S(\boldsymbol{\beta}) = \sum_{i=1}^{n} (y_i - f_i(\boldsymbol{\beta}))^2, \tag{12.30}$$

where $\boldsymbol{\beta}$ is the m-dimensional parameter vector and $f_i = f_i(\boldsymbol{\beta})$ is the regression function; some computational detail was discussed in Section 6.1. For example, for image registration (12.21), y is the gray level, M_1, and f is the gray level, M_2.

The Gauss–Newton algorithm usually works well and has the form

$$\boldsymbol{\beta}_{s+1} = \boldsymbol{\beta}_s + (\mathbf{G}_s'\mathbf{G}_s)^{-1}\mathbf{G}_s'(\mathbf{y} - \mathbf{f}_s), \tag{12.31}$$

where $\mathbf{G}_s = \partial \mathbf{f}/\partial \boldsymbol{\beta}$ is an $n \times m$ matrix evaluated at $\boldsymbol{\beta} = \boldsymbol{\beta}_s$ and $\mathbf{f} = (f_1, ..., f_n)'$, $\mathbf{y} = (y_1, ..., y_n)'$. Can we avoid derivative computation, or, more specifically, can we approximate matrix $\mathbf{G}$ by computing values of function f? Indeed, a finite-difference approach does just that using the approximation $\partial f_i/\partial \boldsymbol{\beta} \approx (f_i(\boldsymbol{\beta} + \Delta\beta \mathbf{1}) - f_i(\boldsymbol{\beta}))/\Delta\beta$, where $\mathbf{1}$ is the $m \times 1$ vector of $1'$s and $\Delta\beta$ is a scalar. When the components of $\boldsymbol{\beta}$ are integers, the finite-difference approach is, perhaps, the only way to assess the derivative, or, more precisely, the relative change (because the derivative is not defined). The DUD algorithm suggests an economical way to approximate the derivatives based on previous computations of $\mathbf{f}$. Indeed, let the values of the regression vector

function be known at $P+1$ points, $\boldsymbol{\beta}_0, \boldsymbol{\beta}_1, ..., \boldsymbol{\beta}_P$, and we want to approximate matrix $\mathbf{G}$ at $\boldsymbol{\beta}_0$. It is assumed that $P \geq m$ and the $P+1$ vectors in the m-dimensional space are in *general position,* meaning that vectors $\{\boldsymbol{\beta}_p - \boldsymbol{\beta}_0, p = 1, ..., P\}$ have rank m. From the definition of the derivative, we have

$$\mathbf{f}_p - \mathbf{f}_0 \simeq \mathbf{G}(\boldsymbol{\beta}_p - \boldsymbol{\beta}_0), \tag{12.32}$$

where $\mathbf{f}_p = \mathbf{f}(\boldsymbol{\beta}_p)$, for $p = 1, ..., P$. The linear system (12.32) for $\mathbf{G}$ has nm unknowns and nP equations. If $P = m$, this system may be solved exactly for $\mathbf{G}$. Otherwise, it is overspecified, so, generally, we find $\mathbf{G}$ using the weighted sum of squares criterion,

$$\operatorname{tr}(\mathbf{GA} - \mathbf{F})'(\mathbf{GA} - \mathbf{F})\boldsymbol{\Omega}^{-1} \Rightarrow \min_{\mathbf{G}}, \tag{12.33}$$

where $\mathbf{A}$ is an $m \times P$ matrix with the pth column $\boldsymbol{\beta}_p - \boldsymbol{\beta}_0$, and $\mathbf{F}$ is an $n \times P$ matrix with the pth column $\mathbf{f}_p - \mathbf{f}_0$. The $P \times P$ weight matrix $\boldsymbol{\Omega}^{-1}$ is positive definite. For example, $\boldsymbol{\Omega}$ may be a diagonal matrix with the pth diagonal element equal to $\|\mathbf{f}_p - \mathbf{f}_0\|^2$. Then points closer to $\mathbf{f}_0$ will have more influence on the derivative approximation. Solving this quadratic optimization problem, we find the LS estimate for the derivative of the regression function based on $P+1$ function values:

$$\mathbf{G} = \mathbf{F}\boldsymbol{\Omega}^{-1}\mathbf{A}'(\mathbf{A}\boldsymbol{\Omega}^{-1}\mathbf{A}')^{-1}. \tag{12.34}$$

Matrix $\mathbf{A}$ has full rank, and therefore matrix $\mathbf{AA}'$ is nonsingular. In fact, formula (12.34) is quite general and may be applied for derivative approximation of any nonlinear function, not necessarily in the nonlinear regression framework. In a special case, when $P = m$, matrix $\mathbf{A}$ is square and then $\mathbf{G} = \mathbf{FA}^{-1}$. This formula has a clear finite-difference flavor, just expressed in a matrix form. Thus, for $P = m$, weighting is irrelevant.

Now, coming back to the SS minimization, substituting (12.34) into the Gauss–Newton update formula (12.31), we finally obtain the DUD algorithm,

$$\boldsymbol{\beta}_{s+1} = \boldsymbol{\beta}_s + (\mathbf{A}_s\boldsymbol{\Omega}^{-1}\mathbf{A}_s')(\mathbf{A}_s\boldsymbol{\Omega}^{-1}\mathbf{F}_s'\mathbf{F}_s\boldsymbol{\Omega}^{-1}\mathbf{A}_s')^{-1}\mathbf{A}_s\boldsymbol{\Omega}^{-1}\mathbf{F}_s'(\mathbf{y} - \mathbf{f}_{s0}).$$

Several variations of the DUD algorithm exist. First, one can incorporate the step length to provide that the value of the SS drop from iteration to iteration. Second, one may use all previous iteration points or just P, closest to the current beta vector. Third, for a special case, when $P = m$ and $\boldsymbol{\Omega} = \mathbf{I}$, matrix $\mathbf{A}$ becomes an $m \times m$ nonsingular matrix and the derivative matrix (12.34) is approximated as $\mathbf{G} \simeq \mathbf{FA}^{-1}$. Then formula (12.34) simplifies to

$$\boldsymbol{\beta}_{s+1} = \boldsymbol{\beta}_s + \mathbf{A}_s(\mathbf{F}_s'\mathbf{F}_s)^{-1}\mathbf{F}_s'(\mathbf{y} - \mathbf{f}_{s0}).$$

Clearly, $\mathbf{G}$, approximated by (12.34), may be treated as its continuous counterpart, and therefore, $s^2(\mathbf{G}'\mathbf{G})^{-1}$ serves as a covariance matrix estimate for $\boldsymbol{\beta}$, where s^2 is the minimum SS divided by the degrees of freedom, $n - m$.

12.7.8 Example: clock alignment

In image registration, the choice of the starting point for the affine parameters, $\boldsymbol{\beta}_0$, is very important. It is a good idea to determine this vector from the respective

landmarks on both images. We illustrate this technique on clock registration; see Figure 12.16. Aligning two 2D figures requires at least three landmarks, assuming that all six parameters are unknown. If only a rigid transformation is allowed ($\beta_2 = \beta_6$ and $\beta_3 = -\beta_5$), two landmarks would be enough. Generally, more landmarks are better. If there are K landmarks on the first and second images, $\{(x_k, y_k), k = 1, ..., K\}$ and $\{(v_k, u_k), k = 1, ..., K\}$, we approximate $2K$ equations,

$$x_k \simeq \beta_1 + \beta_2 v_k + \beta_3 u_k, \quad y_k \simeq \beta_4 + \beta_5 v_k + \beta_6 u_k,$$

by least squares (if $K = 3$, this system can be solved exactly). When the target and source images are not aligned, the SE = 32.7, and after landmark alignment, SE = 10.7.

The `R` function that plots four clock images in Figure 12.16 and uses landmarks for image alignment is shown below. The function `matR` provides image reflection about the y-axis for correct display. Landmark point coordinates for the original and target images are in arrays `cL1` and `cL2`, respectively. The affine transformation parameters are found from the linear least squares that minimizes the Euclidean norm between the landmarks on the original and target images. The indices should be rounded to avoid image display issues.

```
clockFIG=function()
{
dump("clockFIG", "c:\\MixedModels\\Chapter12\\clockFIG.r")
matR=function(M) #matrix reflection about y-axis
{
nr=nrow(M);nc=ncol(M)
MR=M
for(i in 1:nc) MR[,nc-i+1]=M[,i]
return(MR)
}
c1 <- scan("c:\\MixedModels\\Chapter12\\clock1.pgm",what="")
nr1 <- as.numeric(c1[2]); nc1 <- as.numeric(c1[3])
M1 <- matrix(as.numeric(c1[5:length(c1)]), nrow = nr1, ncol = nc1)
c2 <- scan("c:\\MixedModels\\Chapter12\\clock2.pgm",what="")
nr2 <- as.numeric(c2[2]); nc2 <- as.numeric(c2[3])
M2 <- matrix(as.numeric(c2[5:length(c2)]),nrow=nr2,ncol=nc2)
cL1 <- matrix(c(327, 148, 376, 256, 269, 344, 128,
          280, 191, 101, 252, 111), nrow = 2)
cL2 <- matrix(c(302, 154, 349, 258, 245, 343, 108,
          283, 169, 108, 228, 116), nrow = 2)
cL1 <- cL1[, 1:5]; cL2 <- cL2[, 1:5]
nLc1 <- ncol(cL1)
```

```
 for(i in 1:nLc1) {
 i1 <- 1 + 2 * (i - 1)
 i2 <- 2 * i
 X[i1, 1] <- 1
 X[i1, 2:3] <- cL1[1:2, i]
 X[i2, 4] <- 1
 X[i2, 5:6] <- cL1[1:2, i]
 }
 xx <- t(X) %*% X
 xy <- t(X) %*% as.vector(cL2)
 beta <- b <- solve(xx) %*% xy
 par(mfrow = c(2, 2), mar = c(1, 1, 3, 1))
 image(1:nr1, 1:nc1, matR(M1), xlab = "", ylab = "", axes = F,
                  col=gray(0:255/255))
 mtext(side = 3, "Source image", line = 0.25, cex = 1.25)
 for(i in 1:nLc1)
 points(cL1[1+2*(i-1)], nc1-cL1[2*i]+1, pch = 16, cex = 1.25)
 image(1:nr2, 1:nc2, matR(M2), xlab = "", ylab = "", axes = F,
                  col=gray(0:255/255))
 mtext(side = 3, "Target image", line = 0.25, cex = 1.25)
 for(i in 1:nLc1)
 points(cL2[1+2*(i-1)], nc1-cL2[2*i]+1, pch = 16, cex = 1.25)
 del <- matrix(nrow = nr1, ncol = nc1)
 r1 <- min(nr1, nr2)
 r2 <- min(nc1, nc2)
 image(1:r1, 1:nc1, matR(M1[1:r1,1:r2]-M2[1:r1,1:r2]),xlab ="",
                       ylab = "", axes = F,col=gray(0:255/255))
 mtext(side = 3, "Difference before alignment",line=0.25,cex=1.25)
 for(i in 1:nr1)
 for(j in 1:nc1) {
 p1 <- round(b[1] + b[2] * i + b[3] * j)
 q1 <- round(b[4] + b[5] * i + b[6] * j)
 if(q1 > 0 & q1 <= nc2 & p1 > 0 & p1 <= nr2)
 del[i, j] <- M1[i, j] - M2[p1, q1]
 }
 image(1:nr1, 1:nc1, matR(del), xlab = "", ylab = "", axes = F,
                                  col=gray(0:255/255))
 mtext(side=3, "Difference after alignment",line=0.25,cex=1.25)
}
```

Problems for Section 12.7

1. Pick the center of the clock as an additional landmark and redo the image alignment (use the `clock` program). Does it improve the image alignment?

2. Apply linear interpolation to the image used in the function `histgr` from Section 12.5 (modify the function `imageLI` for this purpose). Do linear interpolation and histogram equalization commute? (Is the order of operation important?)

3. See the effect of a slight quadratic alignment transformation by modifying the function `clockROT`.

4*. Write an `R` function which implements the DUD algorithm. Test your function against `nls`.

5. Prove that matrix (12.34) is the solution of minimization problem (12.33).

12.8 Ensemble of structured images

Now we develop a statistical model for an ensemble of structured images following the line of the shape model developed in the previous chapter. Thus, the issue of image registration becomes central. It is assumed that the ensemble consists of N independent images $\{\mathbf{M}_i, i = 1, ..., N\}$ of the same object(s), subject(s), scene, etc. It is allowed to have partial images (showing only part of the scene), but the majority of images should have commonality. Images may have different size, magnification, and viewpoint, so that they are registered up to an affine transformation. Our goal is to reconstruct the true object(s), subject(s), scene, or the *true* image.

Two kinds of assumptions on image-specific transformations may be taken, fixed and random. The first assumption leads to a generalization of the Procrustes model described in Section 11.6, and the second assumption leads to a nonlinear mixed effects model as a generalization of the shape model of Section 11.6.3. To start, we assume that images have the same mean and scale intensity; at the end we relax this assumption.

12.8.1 Fixed affine transformations

Assuming that the images, up to an unknown fixed transformation, differ from the true image by a random error with constant variance, we come to the statistical model,

$$\widetilde{M}_i(\beta_{i1} + \beta_{i2}p + \beta_{i3}q, \beta_{i4} + \beta_{i5}p + \beta_{i6}q) = M(p, q) + \varepsilon(p, q). \tag{12.35}$$

Since images may have different sizes we use the interpolated images indicated by a tilde; see Section 12.7.5. Also, in (12.35), we let

$$p = 1, ..., \overline{P} = \max\{P_i\}, \qquad q = 1, ..., \overline{Q} = \max\{Q_i\}. \tag{12.36}$$

In this model, we treat $\boldsymbol{\beta}_i = (\beta_{i1}, ..., \beta_{i6})'$ as an unknown affine image-specific parameter vector subject to estimation.

The task is to recover the true image M. If errors $\{\varepsilon\}$ are normally distributed, the minimization of the mean-squared error (MSE) is equivalent to maximum likelihood,

$$\frac{1}{|\mathcal{M}|}\sum_{i=1}^{N}\sum_{(p,q)\in\mathcal{M}}\left[\widetilde{M}_i(\beta_{i1}+\beta_{i2}p+\beta_{i3}q,\beta_{i4}+\beta_{i5}p+\beta_{i6}q)-M(p,q)\right]^2, \quad (12.37)$$

where $\mathcal{M}$ is the index set such that the sum of squares is well defined:

$$\mathcal{M}=\{(p,q):1\le\beta_{i1}+\beta_{i2}p+\beta_{i3}q\le P_i, 1\le\beta_{i4}+\beta_{i5}p+\beta_{i6}q\le Q_i\}$$

for all $i=1,...,N$, where (p,q) are from (12.36) and $|\mathcal{M}|$ denotes the number of elements in the set. Apparently, the images must have a common intersection because otherwise the set $\mathcal{M}$ is null. Obviously, if the affine parameters are fixed, the mean image is simply equal to the ensemble average,

$$\overline{M}(p,q)=\frac{1}{N}\sum_{i=1}^{N}\widetilde{M}_i(\beta_{i1}+\beta_{i2}p+\beta_{i3}q,\beta_{i4}+\beta_{i5}p+\beta_{i6}q). \quad (12.38)$$

There are two ways to minimize (12.37): (1) alternate between N separate MSE minimizations over $\boldsymbol{\beta}_i$ and then substitute M with the mean (12.38), or (2) minimize (12.37) over $6N$ parameters simultaneously with $\overline{M}$ in place of M.

12.8.2 Random affine transformations

If, due to sampling design, images are taken at random angles and have a random size, a model with random affine parameters may be adequate,

$$\boldsymbol{\beta}_i=\boldsymbol{\beta}+\mathbf{b}_i, \quad \mathbf{b}_i\sim\mathcal{N}\left(\boldsymbol{\beta},\sigma^2\mathbf{D}\right), \quad i=1,...N, \quad (12.39)$$

where $\boldsymbol{\beta}$ is a known vector, say $\boldsymbol{\beta}=(0,1,0,0,0,1)'$. Matrix $\mathbf{D}$ is the scaled covariance matrix of the random effects $\mathbf{b}_i$ and determines how "free" $\boldsymbol{\beta}_i$ are. The complete model is written in hierarchical fashion: the first-stage model is the same, (12.35), and the second-stage model is (12.39). Combining these into one model and using matrix notation, we obtain a nonlinear mixed effects (NLME) model,

$$\widetilde{\mathbf{M}}_i(\boldsymbol{\beta}+\mathbf{b}_i)=\mathbf{M}+\boldsymbol{\varepsilon}_i. \quad (12.40)$$

Following the line of argumentation for the random effects shape model of Section 11.6.3, model (12.40) requires a large number of images with a relatively small matrix $\mathbf{D}$. For example, if images are rotated up to 2π, the model with fixed transformations may be more adequate.

Several methods of NLME models are discussed in Chapter 8. Laplace approximation minimizes the penalized MSE,

$$\frac{1}{|\mathcal{M}|}\sum_{i=1}^{N}\left\{\sum_{(p,q)\in\mathcal{M}}\left[\widetilde{M}_i(b_{i1}+b_{i2}p+b_{i3}q,b_{i4}+b_{i5}p+b_{i6}q)-\overline{M}(p,q)\right]^2\right.$$
$$\left.+(\mathbf{b}_i-\boldsymbol{\beta})'\mathbf{D}^{-1}(\mathbf{b}_i-\boldsymbol{\beta})'\right\}. \quad (12.41)$$

If $\mathbf{D} \to \mathbf{0}$, the second term dominates in (12.41), resulting in $\mathbf{b}_i = \boldsymbol{\beta}$: All images are the same up to a random error ε. If $\mathbf{D}$ becomes large, the second term vanishes and we arrive at a model with fixed affine parameters, (12.37) and (12.38).

If images have different scale intensity, independent of transformation, we can assume that $\nu_i \sim \mathcal{N}(1, \sigma_\nu^2)$ and incorporate it into (12.40). The mean intensity parameter is unneeded because M is unknown.

Problems for Section 12.8

1. Does the fixed affine image transformation reduce to image alignment from the previous section when $N = 1$?

2*. Develop statistical models for fixed and random rigid transformation of images of different size.

3*. Develop a maximum likelihood estimation of the average true image with images observed up to a random rigid transformation.

12.9 Modeling spatial correlation

So far we have assumed that components of the error term $\boldsymbol{\varepsilon}$ in the image models, such as (12.27) or (12.39), are independent. Clearly, this is a very simplifying assumption because in real images neighboring pixels usually correlate. Much research has been done in the area of statistics to address spatial correlation (Ripley, 1981; Cressie, 1991). Two dominant statistical models for spatial correlation are simultaneous and conditional spatial autocorrelation (SAR and CAR). In imaging, the stochastic distribution on the plane is called a random field, and the most popular stochastic model is called a *Markov Random Field (MRF).* The latter is a variation of spatial autoregression and, similarly to (12.26), can be defined as

$$\varepsilon(p,q) = \sum_{j=-J}^{J} \sum_{k=-K}^{K} \alpha_{jk} \varepsilon(p+j, q+k) + \eta(p,q), \tag{12.42}$$

where $\{\eta(p,q)\}$ are iid uncorrelated random variables with zero mean, and $\{\alpha_{jk}, j = -J, -J+1, ..., J,\ k = -K, -K+1, ..., K\}$ are $(2J+1)(2K+1)$ fixed parameters. We refer the reader to a collection of papers edited by Chellappa and Jain (1993) with various applications of the MRF theory to image models.

An important observation is that one can express ε through η from (12.42) via a linear operator. Consequently, if $\{\eta(p,q)\}$ has normal distribution, components of matrix $\boldsymbol{\varepsilon}$ also have joint multivariate normal distributions with the covariance matrix defined by $\{\alpha_{jk}\}$. Hence, instead of modeling (12.42), one can model the covariance matrix of $\boldsymbol{\varepsilon}$. This approach may be more computationally attractive for the estimation of spatial correlation parameters, such as α in (12.42). The desirable correlation structure is simple enough to derive an estimation procedure and complex enough to describe a variety of possible spatial correlations.

A convenient way to generate a normally distributed random (field) matrix with correlated entries is to pre- and postmultiply a matrix with iid elements by fixed matrices. Indeed, let $\boldsymbol{\eta} = \{\eta(p,q), p = 1, ..., P, q = 1, ..., Q\}$ be a $P \times Q$ matrix with iid normally distributed elements with zero mean and variance σ^2. Let $\mathbf{V}_L$ and

$\mathbf{V}_R$ be $P \times P$ and $Q \times Q$ fixed matrices dependent on some parameter vector, $\boldsymbol{\theta}$. Define

$$\boldsymbol{\varepsilon} = \mathbf{V}_L \boldsymbol{\eta} \mathbf{V}_R, \tag{12.43}$$

a $P \times Q$ random matrix. The elements of this matrix have zero mean and correlate. Matrices $\mathbf{V}_L$ and $\mathbf{V}_R$ should fulfill the following requirements: (1) matrices $\mathbf{V}'_R\mathbf{V}_R$ and $\mathbf{V}_L\mathbf{V}'_L$ are invertible, (2) at $\boldsymbol{\theta} = \mathbf{0}$ these matrices turn into identity matrices. The first assumption excludes matrix deficiency and will be justified later. The second assumption means that $\{\boldsymbol{\varepsilon}\}$ become iid as a special case of (12.43). Now we derive the distribution and the covariance matrix implied by (12.43) expressed in terms of the $(PQ)\times 1$ vector, $\mathrm{vec}(\boldsymbol{\varepsilon})$. Using the properties of the Kronecker product, we obtain

$$\mathrm{vec}(\boldsymbol{\varepsilon}) = \mathrm{vec}(\mathbf{V}_L \boldsymbol{\eta} \mathbf{V}_R) = (\mathbf{V}'_R \otimes \mathbf{V}_L)\mathrm{vec}(\boldsymbol{\eta}).$$

But $\mathrm{vec}(\boldsymbol{\eta}) \sim \mathcal{N}\left(\mathbf{0}, \sigma^2 \mathbf{I}_{PQ}\right)$, so we find that

$$\mathrm{vec}(\boldsymbol{\varepsilon}) \sim \mathcal{N}\left(\mathbf{0}, \sigma^2(\mathbf{V}'_R\mathbf{V}_R \otimes \mathbf{V}_L\mathbf{V}'_L)\right). \tag{12.44}$$

Now it is clear why the first requirement guarantees that matrix $\boldsymbol{\varepsilon}$ has a nondegenerate distribution.

Having found the observation matrix $\boldsymbol{\varepsilon}$ as the difference between the image data and estimated mean, we estimate the parameters of matrices $\mathbf{V}_L$ and $\mathbf{V}_R$ by maximum likelihood. Since the distribution is normal, the log-likelihood, up to a constant, takes the form

$$\begin{aligned} l &= -0.5\left\{(PQ)\ln\sigma^2 + \ln|\mathbf{V}'_R\mathbf{V}_R \otimes \mathbf{V}_L\mathbf{V}'_L| \right. \\ &\quad \left. + \sigma^{-2}\mathrm{vec}'(\boldsymbol{\varepsilon})(\mathbf{V}'_R\mathbf{V}_R \otimes \mathbf{V}_L\mathbf{V}'_L)^{-1}\mathrm{vec}(\boldsymbol{\varepsilon})\right\}. \end{aligned}$$

Using properties of the Kronecker product, we simplify

$$\begin{aligned} \ln|\mathbf{V}'_R\mathbf{V}_R \otimes \mathbf{V}_L\mathbf{V}'_L| &= \ln|\mathbf{V}'_R\mathbf{V}_R| + \ln|\mathbf{V}_L\mathbf{V}'_L|, \\ (\mathbf{V}'_R\mathbf{V}_R \otimes \mathbf{V}_L\mathbf{V}'_L)^{-1} &= (\mathbf{V}'_R\mathbf{V}_R)^{-1} \otimes (\mathbf{V}_L\mathbf{V}'_L)^{-1} \end{aligned}$$

and

$$\begin{aligned} & \mathrm{vec}'(\boldsymbol{\varepsilon})(\mathbf{V}'_R\mathbf{V}_R \otimes \mathbf{V}_L\mathbf{V}'_L)^{-1}\mathrm{vec}(\boldsymbol{\varepsilon}) \\ = \; & \mathrm{vec}'(\boldsymbol{\varepsilon})[(\mathbf{V}'_R\mathbf{V}_R)^{-1} \otimes (\mathbf{V}_L\mathbf{V}'_L)^{-1}]\mathrm{vec}(\boldsymbol{\varepsilon}) \\ = \; & \mathrm{tr}((\mathbf{V}'_R\mathbf{V}_R)^{-1}\boldsymbol{\varepsilon}'(\mathbf{V}_L\mathbf{V}'_L)^{-1}\boldsymbol{\varepsilon}). \end{aligned}$$

Thus, function l can be written as

$$l = -0.5\left\{n\ln\sigma^2 + \ln|\mathbf{Q}_R| + \ln|\mathbf{Q}_L| + \sigma^{-2}\mathrm{tr}(\mathbf{Q}_R^{-1}\boldsymbol{\varepsilon}'\mathbf{Q}_L^{-1}\boldsymbol{\varepsilon})\right\}, \tag{12.45}$$

where $n = PQ$ is the number of image pixels, and

$$\mathbf{Q}_R = \mathbf{V}'_R\mathbf{V}_R, \quad \mathbf{Q}_L = \mathbf{V}_L\mathbf{V}'_L.$$

One can easily express σ^2 through $\mathbf{Q}_R$ and $\mathbf{Q}_L$ as

$$\sigma^2 = n^{-1}\mathrm{tr}(\mathbf{Q}_R^{-1}\boldsymbol{\varepsilon}'\mathbf{Q}_L^{-1}\boldsymbol{\varepsilon}). \tag{12.46}$$

As is seen from (12.45), the likelihood is easy to express in terms of the $P\times P$ and $Q\times Q$ matrices $\mathbf{Q}_L$ and $\mathbf{Q}_R$, so we can model those, not the original $\mathbf{V}_L$ and $\mathbf{V}_R$. One can interpret $\mathbf{Q}_L$ and $\mathbf{Q}_R$ as standard deviations and $\mathbf{V}_L$ and $\mathbf{V}_R$ as variances. The log-likelihood function (12.45) is in general form and it requires further specification for matrices $\mathbf{Q}_L$ and $\mathbf{Q}_R$ to estimate parameters of the covariance spatial matrix. Several covariance models can be suggested for $\mathbf{Q}_L$ and $\mathbf{Q}_R$. Below we develop an idea based on the Toeplitz structure used before to model time series in Section 4.3.4 and shape analysis in Section 11.6.3.

12.9.1 Toeplitz correlation structure

A parsimonious way to model matrices $\mathbf{Q}_L$ and $\mathbf{Q}_R$ is to assume that they have a Toeplitz structure, meaning that the elements on the diagonal parallel to the main diagonal are the same. In terms of the elementary Toeplitz matrix, $\mathbf{Q}$ can be expressed as a linear combination with the coefficients subject to estimation:

$$\mathbf{Q}=\mathbf{I}+\sum_{k=1}^{K}\theta_k\mathbf{T}_{k'}. \tag{12.47}$$

To model various correlation lags, we use the index function $k'=k'(k)$. For instance, if the first and the third lags are used, we have $\mathbf{Q}=\mathbf{I}+\theta_1\mathbf{T}_1+\theta_2\mathbf{T}_3$, so that $k'(1)=1$ and $k'(2)=3$. In (12.47), $\mathbf{T}_{k'}$ is the k'th elementary $P\times P$ Toeplitz matrix, $\{\theta_k\}$ are unknown parameters, and K is the correlation depth. Examples of the elementary Toeplitz matrix are (4.115). Several isotropic random fields ($\mathbf{Q}=\mathbf{Q}_L=\mathbf{Q}_R$) with Toeplitz spatial correlation (12.47) for different depth, K, and $\theta_k=0.9^k$, $k'=k$, are shown in Figure 12.20.

Vertically dependent random fields

To illustrate, we consider the vertically dependent random fields, that is when the columns of matrix $\boldsymbol{\varepsilon}$ are independent ($\boldsymbol{\varepsilon}=\mathbf{V}_L\boldsymbol{\eta}$ and $\mathbf{V}_R=\mathbf{I}$), but the rows correlate according to a Toeplitz structure (12.47). If $\boldsymbol{\varepsilon}_i$ denotes the ith column of matrix $\boldsymbol{\varepsilon}$, we have iid $\boldsymbol{\varepsilon}_i\sim\mathcal{N}(\mathbf{0},\sigma^2\mathbf{Q})$, where $\mathbf{Q}$ is defined by (12.47), $i=1,...,Q$. Two methods of estimation for $\{\theta_k,k=1,...,K\}$ are suggested below.

Variance least squares

According to this method, we estimate the variance parameters σ^2 and $\boldsymbol{\theta}$ by minimizing the sum of squares of the difference between the empirical and theoretical covariance matrices, Section 3.12. Since the empirical covariance matrix is $\boldsymbol{\varepsilon}_i\boldsymbol{\varepsilon}_i'$ and the theoretical matrix is $\sigma^2(\mathbf{I}+\sum\theta_k\mathbf{T}_{k'})$, the variance least squares (VLS) estimate minimizes the function

$$S(\sigma^2,\tau_1,...,\tau_K)=\sum_{i=1}^{Q}\mathrm{tr}\left(\boldsymbol{\varepsilon}_i\boldsymbol{\varepsilon}_i'-\sigma^2\mathbf{I}_P-\sum_{k=1}^{K}\tau_k\mathbf{T}_{k'}\right)^2,$$

where $\tau_k=\sigma^2\theta_k$. Differentiating S with respect to σ^2 and noticing that $\mathrm{tr}(\mathbf{T}_{k'})=0$, we obtain $\sigma^2=\mathrm{tr}(\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}')/n$. Differentiating with respect to τ_k, we obtain K linear equations which can be solved for $\tau_1,...,\tau_K$ as in Section 4.3.

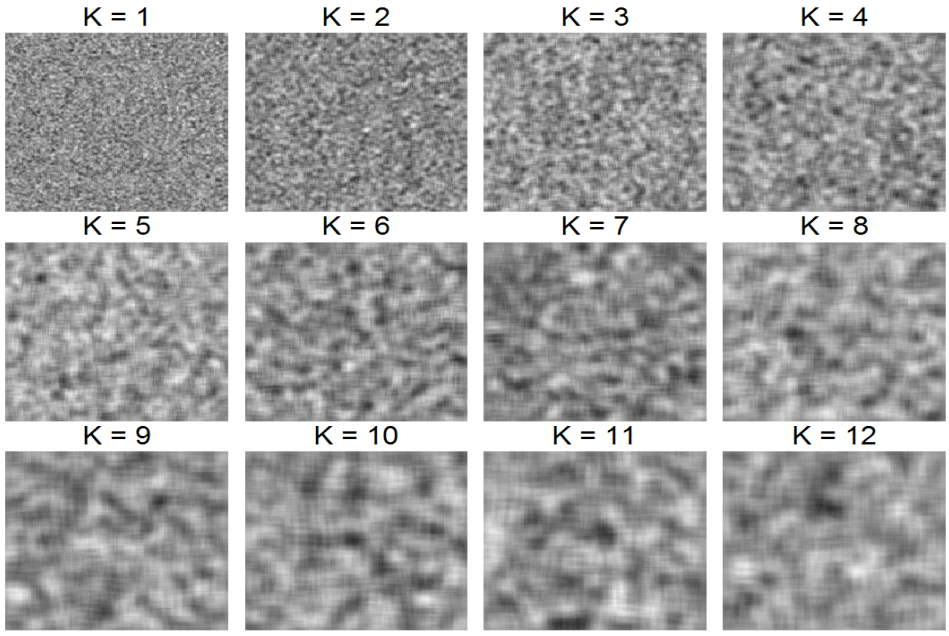

FIGURE 12.20. Random isotropic fields with Toeplitz spatial correlations for different depth, K, and $\theta_k = 0.9^k$.

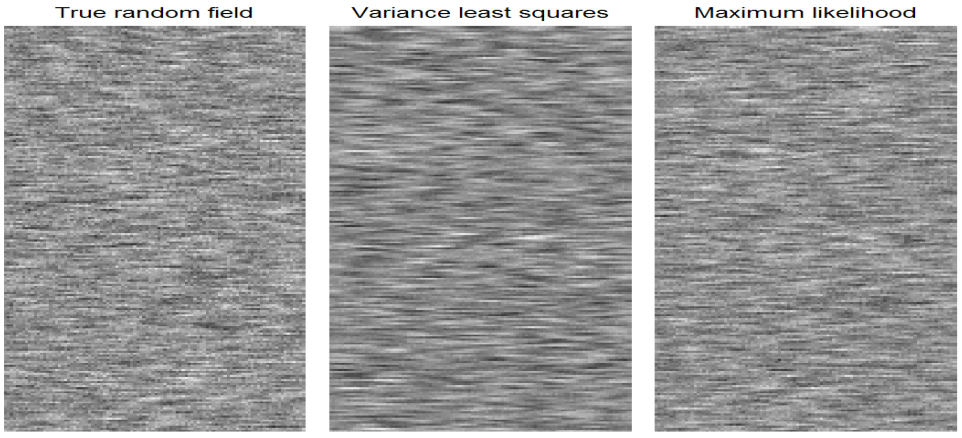

FIGURE 12.21. Three vertically dependent random fields. The first is the true random field with the left Toeplitz correlation structure. The second and third are generated using the variance least squares (VLS) and maximum likelihood (ML) estimates.

Maximum likelihood

Since $\{\boldsymbol{\varepsilon}_i, i = 1, ..., Q\}$ are normally distributed and uncorrelated, the log-likelihood function, up to a constant term, takes the form

$$l = -\frac{1}{2}\left\{ n \ln \sigma^2 + Q \ln \left| \mathbf{I} + \sum \theta_k \mathbf{T}_{k'} \right| + \frac{1}{\sigma^2} \sum_{i=1}^{Q} \boldsymbol{\varepsilon}_i' \left(\mathbf{I} + \sum \theta_k \mathbf{T}_{k'} \right)^{-1} \boldsymbol{\varepsilon}_i \right\}.$$

Differentiating with respect to σ^2, we obtain $\sigma^2 = \text{tr}(\boldsymbol{\varepsilon}\mathbf{Q}^{-1}\boldsymbol{\varepsilon}')/n$. The derivatives with respect to θ_k are

$$\frac{\partial l}{\partial \theta_k} = -0.5 \left\{ Q tr(\mathbf{Q}^{-1}\mathbf{T}_{k'}) - \sigma^{-2}\text{tr}(\boldsymbol{\varepsilon}'\mathbf{Q}^{-1}\mathbf{T}_{k'}\mathbf{Q}^{-1}\boldsymbol{\varepsilon}_i) \right\}, \; k = 1, ..., K,$$

and the second derivatives ($j = 1, ..., K$) are

$$\frac{\partial^2 l}{\partial \theta_k \partial \theta_j} = -0.5 \left\{ -Q tr(\mathbf{Q}^{-1}\mathbf{T}_{j'}\mathbf{Q}^{-1}\mathbf{T}_{k'}) \right.$$
$$\left. + \sigma^{-2}\text{tr}(\boldsymbol{\varepsilon}'\mathbf{Q}^{-1}\mathbf{T}_{j'}\mathbf{Q}^{-1}\mathbf{T}_{k'}\mathbf{Q}^{-1}\boldsymbol{\varepsilon}) + \sigma^{-2}\text{tr}(\boldsymbol{\varepsilon}'\mathbf{Q}^{-1}\mathbf{T}_{k'}\mathbf{Q}^{-1}\mathbf{T}_{j'}\mathbf{Q}^{-1}\boldsymbol{\varepsilon}) \right\}. \quad (12.48)$$

If $\mathbf{H}$ is the $K \times K$ matrix with the elements $\{-\partial^2 l/\partial\theta_k\partial\theta_j\}$, the Newton's iterations yield

$$\boldsymbol{\theta}_{s+1} = \boldsymbol{\theta}_s + \lambda_s \mathbf{H}^{-1} \left(\frac{\partial l}{\partial \boldsymbol{\theta}} \right), \quad s = 0, 1, ...,$$

where λ_s is the step length to ensure that the l value increases from iteration to iteration. Matrix $\mathbf{H}^{-1}$ is the asymptotic covariance matrix for $\boldsymbol{\theta}$ with the variances on the diagonal. The value $Z_k = \widehat{\theta}_k / SE(\widehat{\theta}_k)$ is a characteristic of statistical significance and serves as guidance for the choice of the right correlation structure.

Example

The two methods of estimation are applied to a vertically dependent random field generated with $K = 5$ and $\theta_k = 0.35 - 0.05k$ and $\sigma^2 = 0.5$, see Figure 12.21. The true values, VLS and ML estimates, are shown in Figure 12.22. As the reader can see, maximum likelihood yields better estimates; also, the random field generated in Figure 12.21 looks closer to the true random field.

12.9.2 Simultaneous estimation of variance and transform parameters

Spatial correlation modifies previous models for image registration and ensemble of images. In particular, the MSE criterion (12.21) or (12.37) will be replaced by the weighted MSE.

To illustrate, we simply assume that we have a uniform image $M(p, q)=\mu + \varepsilon(p, q)$ or in matrix form, $\mathbf{M} = \mu \mathbf{1}_P \mathbf{1}_Q' + \boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon}$ is the random field defined by (12.43). The parameter of interest is the intensity μ. The log-likelihood function for this image model is given by (12.45) with $\boldsymbol{\varepsilon}$ replaced by $\mathbf{M} - \mu \mathbf{1}_P \mathbf{1}_Q'$. Let us assume that the matrices $\mathbf{Q}_L$ and $\mathbf{Q}_R$ are known. Then the MLE for μ minimizes

$$S(\mu) = \text{tr}[\mathbf{Q}_R^{-1}(\mathbf{M} - \mu \mathbf{1}_P \mathbf{1}_Q')' \mathbf{Q}_L^{-1} (\mathbf{M} - \mu \mathbf{1}_P \mathbf{1}_Q')].$$

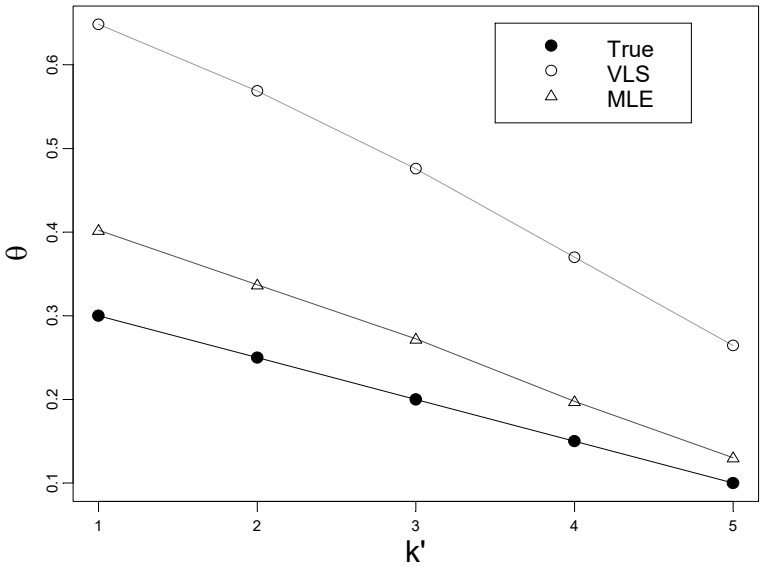

FIGURE 12.22. Estimation by variance least squares (VLS) and maximum likelihood (ML). For this example, the ML estimate is closer to the true θ.

Taking the derivative of S with respect to μ, we find the solution as the generalized least squares estimator,

$$\widehat{\mu}_{ML} = \frac{\mathbf{1}_P' \mathbf{Q}_L^{-1} \mathbf{M} \mathbf{Q}_R^{-1} \mathbf{1}_Q}{(\mathbf{1}_P' \mathbf{Q}_L^{-1} \mathbf{1}_P)(\mathbf{1}_Q' \mathbf{Q}_R^{-1} \mathbf{1}_Q)} = \frac{\sum \left(\mathbf{Q}_L^{-1} \mathbf{M} \mathbf{Q}_R^{-1} \right)_{pq}}{\sum (\mathbf{Q}_L^{-1})_{pq} \sum (\mathbf{Q}_R^{-1})_{pq}}. \tag{12.49}$$

Notice that even a simple model for intensity level leads to a weighted mean. For a proper estimation, matrices $\mathbf{Q}_L$ and $\mathbf{Q}_R$ must be estimated previously and then formula (12.49) applied. In the maximum likelihood approach, μ and these matrices are estimated simultaneously. Consequently, if two images are aligned, the MSE criterion should be replaced by the weighted MSE of the form (12.23), where $w(p, q)$ is the (p, q)th element of the inverse covariance matrix.

Problems for Section 12.9

1. Does the log-likelihood function (12.45) have a maximum; that is, is model (12.43) specified correctly? Does this model turn into a correctly specified model with additional restriction $\text{tr}(\mathbf{Q}_R) = \text{tr}(\mathbf{Q}_L) = 1$? Maximize l, given by equation (12.45), over $\mathbf{Q}_R$ and $\mathbf{Q}_L$ under these rectrictions.

2. The `R` function `randmat` plots 36 simulated images with various spatial correlation structure. Identify what correlation structure is used. What correlation model described in this section was used to create those images?

3. Find the information matrix for the log-likelihood by taking the expectation of the Hessian (12.48).

4*. There are three PGM images of the pine bark in the directory `"c:\\MixedModels\\Chapter12\\bark\\`. Write an `R` function that plots the images. Do they look like

vertically dependent random fields in Figure 12.21? Can these images be modeled via the left Toeplitz correlation structure? Compute correlation coefficients between rows and pick those that produce maximum values. These values may be used to construct adequate Toeplitz matrices.

12.10 Summary points

- Mathematically, a gray image is a $P \times Q$ matrix with integer entries that take values from 0 (black) to 255 (white). A color image in the RGB format can be equivalently represented as three grayscale images: Red, Green and Blue.

- *Image processing* is a well-established discipline with a variety of techniques to enhance and restore one image at a time. However, statistical aspects of image estimation and testing are underdeveloped. For example, least squares is used as a criterion of image discrepancy but not as a statistical method of estimation of a statistical model. Consequently, the least squares estimate (LSE) is rarely accompanied by its standard error as a characteristic of how the LSE is sensitive to the data. Statistical hypothesis testing for images is not developed either. For example, there is no analogy of a t-test for images when two sample images are compared.

- Classic statistics deals with numbers; statistical image analysis deals with matrices of numbers. The revolution in digital imaging poses challenging problems to statistical science. A marriage of image processing and statistics creates a new discipline, *Statistical Image Analysis*. This chapter lays out the foundation of this discipline using a model-based approach. In the process, many techniques of image processing receive a theoretical justification, such as histogram equalization and the Karhunen–Loeve transformation.

- We classify images into two groups: structured and unstructured. Structured are images of an object (objects) or easy-to-recognize scene. Unstructured are images without content, such as fabric (textures) or microscopic images. Consequently, for structured images, image registration typically occurs when objects on the first images are aligned with the same objects on the second image. Unstructured images are analyzed using gray level distributions and histograms. Thus, we say that two unstructured images are the same if they have the same gray level distribution.

- The histogram is a standard tool of image processing and is typically used for image enhancement. We apply the cumulative distribution function to compare unstructured images. An advantage of the distribution function is that several distribution functions can be plotted on one graph—a convenient graphical tool for image comparison. A nonparametric Kolmogorov–Smirnov criterion was used to test whether two images are the same.

- We have developed a multinomial distribution for grayscale image. The maximum likelihood estimate for the probability that the gray level of a pixel takes value g is the histogram value $h_g, g = 0, 1, ..., 255$. Two χ^2-tests and one

likelihood ratio parametric test were developed to test whether two images have the same gray level distribution.

- Entropy is a milestone of information theory. The entropy of a binary message, as a sequence of 1s and 0s, is zero if the sequence contains only 0 (or 1). The entropy is maximum if the probability of 1 appearing is $\frac{1}{2}$. Based on the multinomial distribution, we have introduced the notion of image entropy. If pixels have the same gray level (blank image), the image entropy is zero.

- We have introduced Entropy Per Pixel (EPP) as a unit for image information, $EPP = -\sum h_g \log_2 h_g$ bits. The absolute maximum of EPP is 8 bits and it is attained when each gray level has the same probability of occurrence, 1/256. Hence, a popular image enhancement technique known as *histogram equalization* maximizes EPP. Another application of the EPP concept is used to reduce a gray image to a four-gray level image—the thresholds must be quartiles of the distribution function.

- Typically, we deal with an ensemble of unstructured images. Two statistical models based on the multinomial distribution were developed in the framework of mixed model methodology: fixed- and random-shift models. These are analogs of the fixed and random intercept models studied in Sections 2.4 and 7.2, respectively. Logit transformation reduces a nonlinear model to the linear mixed model extensively studied in Chapters 2 through 4. Especially effective is a two-stage estimation method to analyze an ensemble of images. In particular, we have demonstrated how to use this method to compare two samples of images as a generalization of the standard t-test.

- Image alignment and registration is essential for content-dependent image comparison; for example, when a pixel-by-pixel difference is to be taken. Four types of model registration may be suggested: landmark-based, affine, nonlinear, and random (stochastic). In the landmark model, images are aligned such that the landmark points from two images coincide or are as close as can be solved by least squares.

- The affine registration model is the easiest and reduces to the mean squared error minimization over six affine coefficients (parameters) for a 2D image. If the transformation is rigid, it reduces to four parameters. If only rotation is required and the size remains the same, we come to a minimization under a quadratic constraint. Nonlinear registration is typically used a polynomial of a low degree and again reduces to the unconstrained MSE minimization. Random registration is the most complicated and can be accomplished via the nonlinear mixed effects model estimation of Chapter 8.

- A precise image registration requires image interpolation. The easiest is the linear interpolation, although some more elaborate methods, such as B-splines, exist. The derivative-free algorithm for MSE minimization may be preferable because we deal with a discrete function and the derivatives do not exist. The theory of linear statistical hypothesis testing may be applied to affine parameters; for example, to test whether the transformation is rigid.

- If several images of the same object or scene are available, the images must be aligned first to derive the mean image. Two statistical models for the affine parameters can be suggested: fixed and random. The former model assumes that the parameters are fixed and unknown and therefore can be found from individual MSE minimization. The latter model is more complex and assumes that the affine parameters are random with population-averaged values. This model leads to a nonlinear mixed effects model, studied extensively in Chapter 8. In particular, after Laplace approximation, the parameters are found from a penalized MSE.

- A realistic assumption in content-dependent image analysis is to assume spatial correlation that implies treating an image as a random field. Several models for spatial correlation may be suggested. A parsimonious model uses a Toeplitz matrix and can describe complex statistical intra-image dependencies. Methods of estimation developed in the earlier chapters, such as variance least squares or maximum likelihood, apply readily. Spatial correlation complicates image alignment and registration because the variance and transform parameters must be estimated simultaneously. In particular, instead of the mean squared error, one should use the weighted mean-squared error.

- Statistics should play a more important role in image science—from image processing to image reconstruction. For example, little work has been done in applying powerful statistical hypothesis testing to image comparison. Today, image analysis is method-driven. To make further advances, it should be model-driven. A good example of a model-driven image reconstruction is the PET model based on the Poisson distribution. Statistical image modeling not only yields an efficient fitting method but also generates the covariance matrix and/or the likelihood value needed for statistical significance testing, model selection, and verification. We strongly believe that a statistical model-based image analysis will bring image science to the next level.