

```

function(image)
{
# Set default page/screen width in characters
options(width = 80)
# Program credit acknowledgement
cat("WP.FRACTAL program Copyright 1995 Cameron L. Jones\n email:
CJONES@swin.edu.au\nVersion 1.0 Release 1 for S-Plus (MS Windows 3.1)\nWavelet Packet
Computation of 2-D objects or image data\n to estimate the Fractal Dimension from computation of the
Hurst exponent.\n")
# Image size requirement
cat("This program operates on images of size 200x300 (nrow,height x ncol,width)\n\n")
# Wavelet Packet Computation to determine the fractal dimension of 2-D objects.
cat("Enter the path and file name for your *.RAW image file:\n")
# Read input file and select the wavelet filter to use for wavelet packet transform of
# image.
fil <- readline()
fila <- read.image(nrow = 200, ncol = 300, file = fil)
myimage <- fila
cat("Input Wavelet Transform Filter to use for analysis:\n\nChoose from the following
list:\nDAUBLETs: haar(d2) d4 d6 d8 d10 d12 d14 d16 d18 d20\nSYMMLETs: s8 s10 s12 s14
s16 s18 s20\nCOIFLETs: c6 c12 c18 c24 c30\n\nYour choice is: ")
ab <- readline()
wavelet.options(wavelet = ab)
# Perform wavelet packet transform and sort wavelet packet coefficient energies
# from highest to lowest.
myimage.wpc<-wp.costs.2d(myimage)
myimage.wpbb <- best.basis(myimage.wpc, data = myimage)
wpbb.sort <- rev(sort(unclass(myimage.wpbb)^2))
# Select subset of sorted coefficient index positions and retain corresponding
# coefficient energies.
wpbb.sorta <- wpbb.sort[c(1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024,
2048, 2560, 3200, 4000, 5000, 6250, 7813, 9766, 12208, 15260,
19075, 23844, 29805, 37256, 46570, 58213, 60000, )]
win.graph()
# Examine coefficient energy values and select cut-off for preliminary graph plotting.
y1 <- wpbb.sorta
x1 <- c(1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 2560, 3200,
4000, 5000, 6250, 7813, 9766, 12208, 15260, 19075, 23844, 29805,
37256, 46570, 58213, 60000)
print(y1)
cat("Examine the list of Wavelet Packet Coefficient Energies.\nTake particular note of the
Coefficient Index Position number where\nthe energy falls below 1.\nFor binary images of 200x300
pixel size,the 'Fmin'size to\ninclude in the least squares regression is the one that returns the highest
r^2 value (i.e closest to 1)\nCoefficient Energies <500 are highlighted in RED.\n")
par(pty = "s")
plot(x1, y1, log = "xy")
label <- y1 < 1
text(x1, y1, label, col = 2, adj = 0)
cat("Enter the number of the last value\nyou want included in the regression: ")
# Read user input.
v <- readline()
mode(v) <- "numeric"
y2 <- y1[1:v]
x2 <- x1[1:v]
# Prepare final version of graphical output.
plot(x2, y2, log = "xy", xlab = "Sorted Coefficient Index Position",
ylab = "Wavelet Packet Energy Magnitude", type = "n")
points(x2, y2, pch = 18, mkh = 2)
mtext((ab), line = 0.3, adj = 0)
mtext("Wavelet Packet Transform", line = 0.3, adj = 0.345)
# Perform least squares linear regression through data and compute slope exponent.
fit <- lm(log10(y2) ~ log10(x2))
abline(fit, col = 4)
print(summary(fit))
cat("The Hurst Exponent,H is estimated from: (slope)+1=H, and the\nFractal
Dimension,D is determined from: 2-H=D.\nThe slope is the log10(x2) value.\n\n")
repeat {

```

```

        cat("Do you want to re-compute the regression using fewer data points?\n
Type YES=y , NO=n :")
        r <- readline()
        if(r == "n")
            break
        else y1 <- wpbb.sorta
        x1 <- c(1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 2560,
            3200, 4000, 5000, 6250, 7813, 9766, 12208, 15260, 19075,
            23844, 29805, 37256, 46570, 58213, 60000)
        print(y1)
        cat("Examine the list of Wavelet Packet Coefficient Energies.\nTake particular note of
the Coefficient Index Position number where\nthe energy falls below 1.\nFor binary images of 200x300
pixel size, the 'Fmin'size to\ninclude in the least squares regression is the one that returns the highest
r^2 value (i.e. closest to 1)\nCoefficient Energies <500 are highlighted in RED.\n")
        par(pty = "s")
        plot(x1, y1, log = "xy")
        label <- y1 < 1
        text(x1, y1, label, col = 2, adj = 0)
        cat("Enter the number of the last value\nyou want included in the regression:")
# Read user input.
        v <- readline()
        mode(v) <- "numeric"
        y2 <- y1[1:v]
        x2 <- x1[1:v]
# Prepare final version of graphical output.
        plot(x2, y2, log = "xy", xlab =
            "Sorted Coefficient Index Position", ylab =
            "Wavelet Packet Energy Magnitude", type = "n")
        points(x2, y2, pch = 18, mkh = 2)
        mtext((ab), line = 0.3, adj = 0)
        mtext("Wavelet Packet Transform", line = 0.3, adj = 0.345)
# Perform least squares linear regression through data and compute slope exponent.
        fit <- lm(log10(y2) ~ log10(x2))
        abline(fit, col = 4)
        print(summary(fit))
        cat("The Hurst Exponent,H is estimated from: (slope)+1=H, and the\nFractal
Dimension,D is determined from: 2-H=D.\nThe slope is the log10(x2)
value.\n\n")
        repeat {
            cat("Do you want to decompose the image using another wavelet
filter?\n Type YES=y , NO=n :")
            m <- readline()
            if(m == "n")
                break
            else cat("Input Wavelet Transform Filter to use for analysis: ")
            ab <- readline()
            wavelet.options(wavelet = ab)
# Perform wavelet packet transform and sort wavelet packet coefficient energies
# from highest to lowest.
            myimage.wpc<-wp.costs.2d(myimage)
            myimage.wpbb <- best.basis(myimage.wpc, data = myimage)
            wpbb.sort <- rev(sort(unclass(myimage.wpbb)^2))
# Select subset of sorted coefficient index positions and retain corresponding
# coefficient energies.
            wpbb.sorta <- wpbb.sort[c(1, 2, 4, 8, 16, 32, 64, 128,
                256, 512, 1024, 2048, 2560, 3200, 4000, 5000,
                6250, 7813, 9766, 12208, 15260, 19075, 23844,
                29805, 37256, 46570, 58213, 60000)]
            win.graph()
# Examine coefficient energy values and select cut-off for preliminary graph plotting.
            y1 <- wpbb.sorta
            x1 <- c(1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024,
                2048, 2560, 3200, 4000, 5000, 6250, 7813, 9766,
                12208, 15260, 19075, 23844, 29805, 37256, 46570,
                58213, 60000)
            print(y1)

```

```

        cat("Examine the list of Wavelet Packet Coefficient Energies.\nTake
particular note of the Coefficient Index Position number where\nthe energy falls below 1.\nFor binary
images of 200x300 pixel size, the 'Fmin'size to\ninclude in the least squares regression is the one that
returns the highest r^2 value (i.e closest to 1) \nCoefficient Energies <500 are highlighted in RED.\n")
        par(pty = "s")
        plot(x1, y1, log = "xy")
        label <- y1 < 1
        text(x1, y1, label, col = 2, adj = 0)
        cat("Enter the number of the last value\nyou want included in the
regression: ")

# Read user input.
        v <- readline()
        mode(v) <- "numeric"
        y2 <- y1[1:v]
        x2 <- x1[1:v]

# Prepare final version of graphical output.
        plot(x2, y2, log = "xy", xlab =
            "Sorted Coefficient Index Position", ylab =
            "Wavelet Packet Energy Magnitude", type = "n")
        points(x2, y2, pch = 18, mkh = 2)
        mtext((ab), line = 0.3, adj = 0)
        mtext("Wavelet Packet Transform", line = 0.3, adj =
            0.345)

# Perform least squares linear regression through data and compute slope exponent.
        fit <- lm(log10(y2) ~ log10(x2))
        abline(fit, col = 4)
        print(summary(fit))
        cat("The Hurst Exponent,H is estimated from: (slope)+1=H, and
the\nFractal Dimension,D is determined from: 2-H=D.\nThe slope is the

log10(x2) value.\n\n")
    }
}

```