Geometric Morphometrics Manual ENG

Version 1.0

Joana Reis Vasconcelos
PhD in Biological Sciences
Universidad de Las Palmas de Gran Canaria
https://jreisvasconcelos.github.io

July 30, 2025

Contents

List of Figures

List of Tables

How to Cite This Manual

If you use this manual or code in your work, please cite it as:

Vasconcelos, J. (2025). *Geometric Morphometrics Manual Using Stereomorph and Morpho in R*. Zenodo. https://doi.org/10.5281/zenodo.16617767

Workflow Overview

This schematic outlines the key steps in a typical geometric morphometric analysis, from specimen imaging to statistical interpretation and export of results.

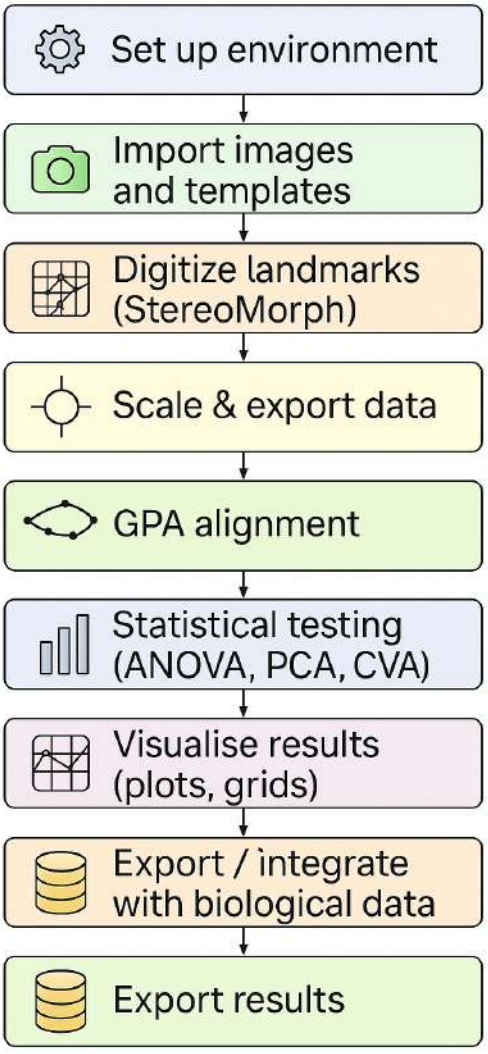

Figure 1: Overview of the morphometric analysis workflow.

1 Introduction

This manual provides a step-by-step guide for performing two-dimensional (2D) geometric morphometric analyses using the R environment. It is designed for students and researchers working with landmark- and curve-based shape analyses of biological specimens. The primary focus is on marine taxa, with worked examples using two species: the fish *Trachurus trachurus* (landmark-based analysis) and the mollusc *Phorcus sauciatus* (combining landmarks and curves).

The workflows presented here integrate multiple R packages, including *StereoMorph* [1, 2] for digitisation, *geomorph* [3, 4] and *Morpho* [5] for statistical shape analysis and thin-plate spline deformation grids, *candisc* [6, 7]for canonical variate analysis (CVA), and ggplot2 for high-quality visualisation.

Beyond its methodological focus, this manual highlights the wide range of applications of geometric morphometrics in biology. Shape analyses can reveal meaningful morphological variation that is often linked to environmental or genetic drivers. In population biology and fisheries science, these techniques are especially useful for identifying population structure, assessing local adaptation, or defining management units based on phenotypic differentiation. For instance, shape variation in body or otolith contours has been effectively used to distinguish between fish populations across environmental gradients, supporting stock identification and sustainable fisheries management [8].

2 Photographic Setup and Specimen Preparation

Standardised image acquisition is essential for geometric morphometric workflows. High-quality, consistent photographs ensure the accuracy of subsequent landmark digitisation and shape analysis.

Photographic Stand

All photographs in this manual were taken using a Kaiser RS1 copy stand system with an adjustable LED lighting source and smartphone holder (Figure 2). This setup provides a perpendicular, stable positioning of the camera relative to the specimen.

Specimen Preparation

To optimise fin and body visibility, fresh or thawed specimens should be pinned to a neutral, non-reflective surface (e.g. polystyrene), with fins fully extended and the specimen positioned in a left lateral view — that is, the left side facing upward and the mouth oriented to the left (Figure 3). The fish should lie in a straight, horizontal line to avoid body curvature, which can compromise landmark accuracy. Pinning ensures consistent landmark placement and minimises deformation. All specimens should be photographed from the same distance and at the same resolution to maintain comparability.

Figure 2: Kaiser RS1 copy stand with mounted smartphone and lateral illumination setup.

Figure 3: Pinned specimen with fins spread using coloured pins on a polystyrene base.

3 Getting Started

Before starting, ensure you have R installed on your computer. For macOS users, also install XQuartz from `https://www.xquartz.org/` — this is required for some graphical interfaces used by the StereoMorph package. This manual relies on several R packages for geometric morphometric analysis, statistical testing, and data visualisation. Install and load the StereoMorph package by running:

```
# Install StereoMorph if not already installed
install.packages("StereoMorph")

# Load the package
library(StereoMorph)
```

Repeat the same steps to install and load the other required packages:

- `geomorph`
- `Morpho`
- `candisc`
- `ggplot2`

For example:

```
install.packages("geomorph")
library(geomorph)
```

⋆ **Tip:** You only need to run the installation commands once. Loading `library(...)` is required in each session.

4 Setting the Working Directory

Set your working directory to the folder that contains your image files and landmark templates:

```
setwd("your/path/to/GeometricManual")
# Update this path to match your setup
```

5 Landmarking with *Trachurus trachurus*

We will use a 17-landmark scheme defined by Vasconcelos et al. [8], illustrated in Figure 4.

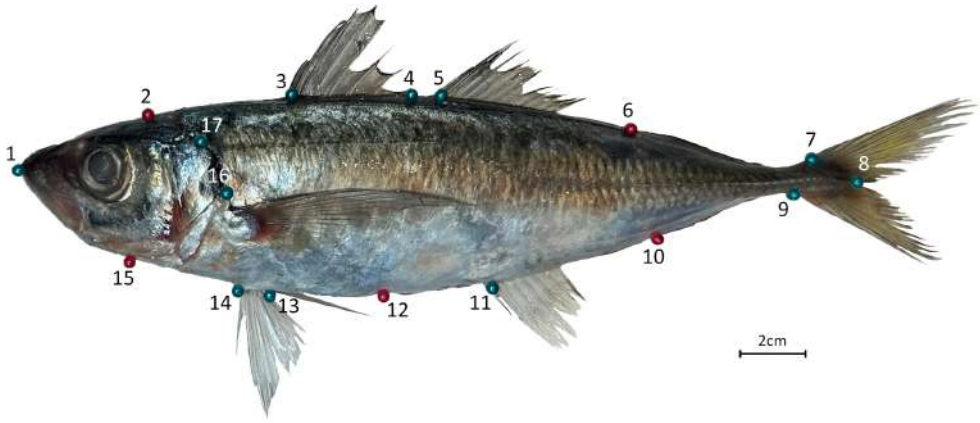

Figure 4: Landmark scheme used for *Trachurus trachurus* (from [8])

6 Preparing the Landmark File

Before launching the `StereoMorph` digitisation interface, you must prepare a plain text file containing the list of landmarks you will digitise. This file can be named, for example, `landmarks.txt`, and should be saved in the same folder where your images are stored.

You can use simple labels such as:

```
LM1
LM2
LM3
...
LM17
```

Alternatively, you may use descriptive anatomical labels. These labels will appear in the digitising interface to help you identify each landmark. For example:

```
Anterior tip of the mouth
Dorsal margin of the eye
Midpoint between the anterior tip of the mouth and the origin
of the first dorsal fin
```

```
Base of the first dorsal fin
...
```

Table 1 shows the set of anatomical definitions used in this study, adapted from [8].

Table 1: Definitions of anatomical landmarks used for *Trachurus trachurus* (adapted from [8])

Landmark	Definition
LM1	Anterior tip of the mouth
LM2	Midpoint between the anterior tip of the mouth and the origin of the first dorsal fin
LM3	Base of the first dorsal fin
LM4	Tip of the first dorsal fin
LM5	Base of the second dorsal fin
LM6	Midpoint between the base of the second dorsal fin and the anterior attachment of the dorsal membrane from the caudal fin
LM7	Anterior attachment point of the dorsal membrane from the caudal fin
LM8	Rear end of the vertebral column
LM9	Anterior attachment point of the ventral membrane from the caudal fin
LM10	Midpoint between the anterior attachment of ventral membrane from caudal fin and the origin of the anal fin
LM11	Base of the anal fin
LM12	Midpoint between the origin of the anal fin and the insertion end of the pelvic fin
LM13	End of insertion of the pelvic fin
LM14	Base of the pelvic fin
LM15	Most posterior point of the maxilla
LM16	Base of insertion of the pectoral fin
LM17	Base of insertion of the operculum

Make sure to save this file in the same directory as your image files. You will load it when launching the digitisation interface.

7 Digitising Landmarks

To begin digitising landmarks, use the `digitizeImages()` function from the `StereoMorph` package:

```
digitizeImages(image.file = "Fish",
               landmarks.file = "landmarks.txt")
```

This command launches a Shiny-based graphical interface for interactive landmark digitisation.

On the top-right side of the interface, you will see four tabs: `Settings`, `Landmarks`, `Curves`, and `Scaling` (Figure 5).

To begin digitising:

- Click the `Landmarks` tab.
- Select the first landmark (e.g., `LM1`) — it will appear bold.
- Click on the image to place the selected landmark.
- Repeat this process for all remaining landmarks in the list (Figure 6).

Note: You can double-click any placed landmark on the image to reselect it and drag to adjust its position. This allows you to correct placement errors without restarting the process.

8 Setting the Scale

Once all landmarks have been placed (Figure 6), navigate to the `Scaling` tab (Figure 7) in the `StereoMorph` interface.

To set the scale (Figure 7):

- Click on the image to place two points along the known scale bar (e.g., a segment representing 1 cm).
- Enter the real-world distance (e.g., 1) in the scale input box.

This step is essential to ensure that the landmark coordinates are properly scaled for geometric morphometric analyses.

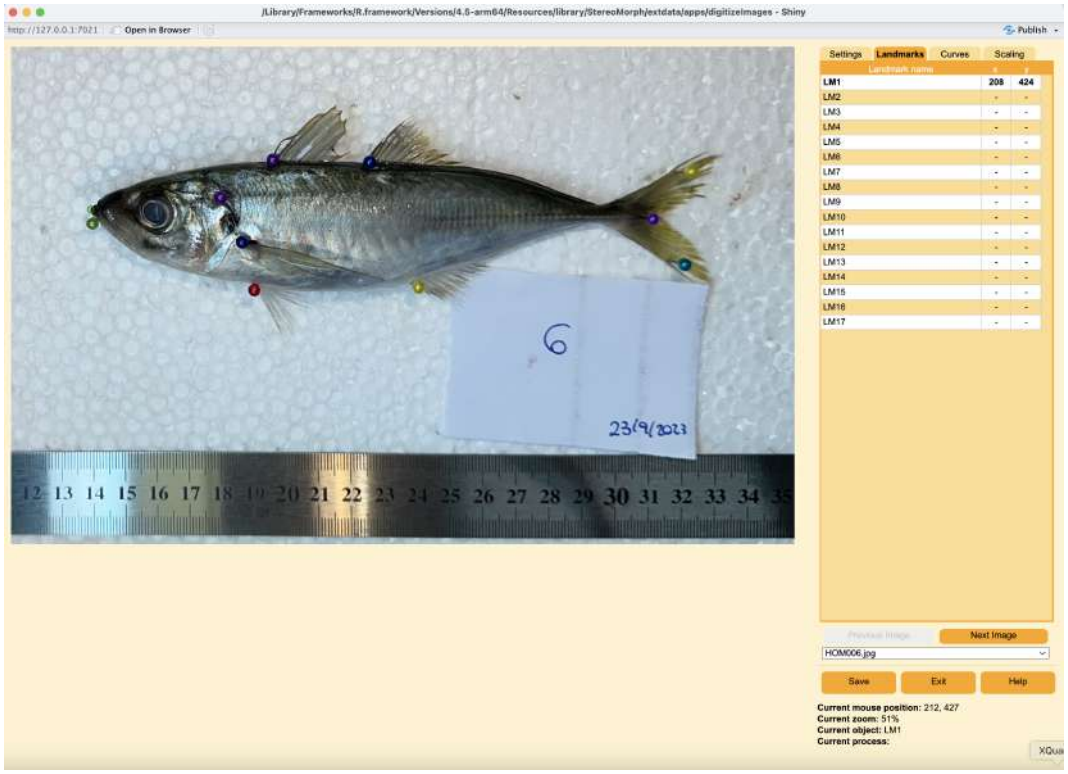

Figure 5: *StereoMorph* interface showing the selection of anatomical landmarks during digitisation.

9 Extracting and Exporting Landmark Data

After completing the digitisation and scaling steps, you can extract and export the landmark data for analysis using the `readShapes()`, `writeLMToTPS()` and `readland.tps()` functions:

```
# Read all shape files from the output folder
read_shapes <- readShapes("Shapes")

# Export pixel coordinates to TPS format
writeLMToTPS("Shapes", "Shapes.tps",
             in.pixels = TRUE,
             flip.y = TRUE,
             flip.x = FALSE)
```

Figure 6: Completed digitisation process with all landmark coordinates listed and visible.

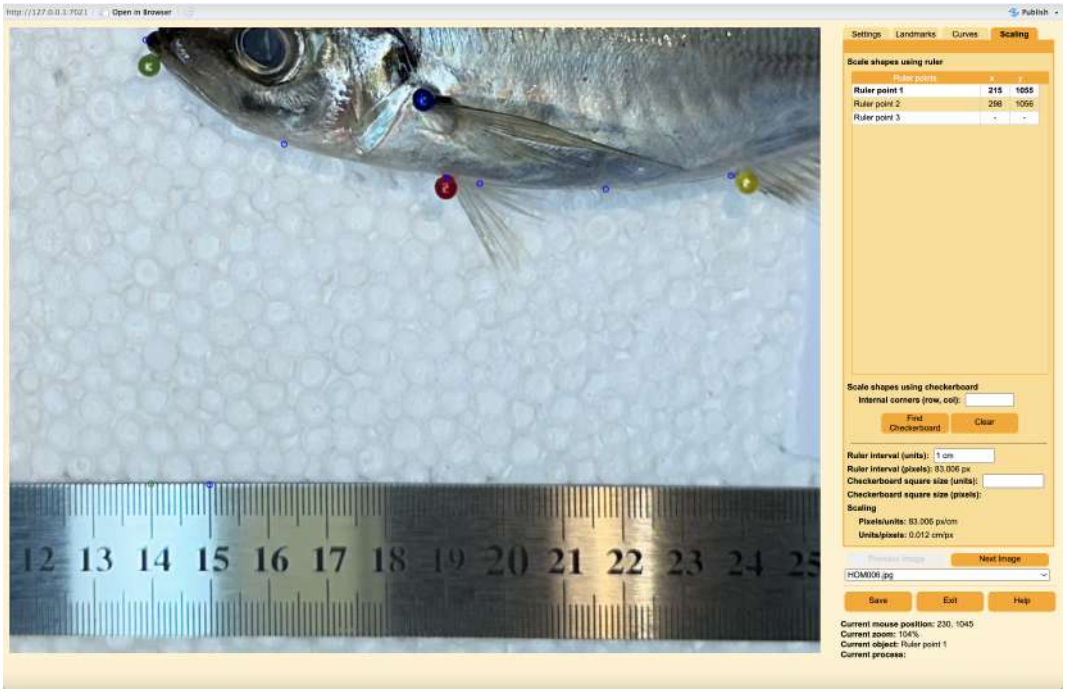

Figure 7: Calibration using a known distance (e.g., 1 cm) in the Scaling tab to convert pixel coordinates to real-world units.

```
# Read TPS file using the geomorph package
myData <- readland.tps("Shapes.tps",
                       specID = "ID",
                       readcurves = FALSE,
                       warnmsg = TRUE)
```

The function `writeLMToTPS()` exports the digitised landmark coordinates to a `.tps` file, which is compatible with downstream analysis in `geomorph`. The options `flip.y = TRUE` and `flip.x = FALSE` adjust the orientation of coordinates to match R's Cartesian system.

Use `readland.tps()` to import the TPS file and store the landmark data as a 3D array, with each slice representing one specimen.

10 Integrating with Biological Data

After preparing and exporting your landmark data (e.g., from a `.tps` file), you may want to combine it with biological metadata such as specimen ID, species, sex, or sampling location. This information is typically stored in a spreadsheet format (e.g., an Excel file).

Use the following code to import and prepare your metadata:

```
# Load biological dataset
BDTrac <- read_excel("Data_Trachurus_trachurus.xlsx")

# Preview structure
str(BDTrac)

# Format variables appropriately
BDTrac[1:3] <- lapply(BDTrac[1:3], as.factor)
BDTrac[4:6] <- lapply(BDTrac[4:6], as.numeric)
```

This example assumes that the first three columns represent categorical variables (e.g., Pic code, Species, Capture location), and the next three columns represent continuous data (e.g., number, total length, weight, etc).

11 Generalized Procrustes Analysis (GPA)

To make landmark configurations comparable across specimens, we apply Generalized Procrustes Analysis (GPA), which standardises shapes by removing the effects of translation, rotation, and scale.

Use the `gpagen()` function from the `geomorph` package:

```
myGPA <- gpagen(myData, PrinAxes = FALSE, ProcD = TRUE)
summary(myGPA)
```

This will return a list containing Procrustes-aligned coordinates for each specimen.

- `myGPA$coords` holds the aligned landmark coordinates.
- `myGPA$Csize` gives the centroid size of each specimen.

We now create a data frame combining shape, centroid size, and biological metadata (e.g. sampling site and total length):

```
Mdf <- geomorph.data.frame(
  shape = myGPA$coords,
  cs = myGPA$Csize,
  site = BDTrac$Capture_Location,
  Csize = log(myGPA$Csize),
  size = BDTrac$TL_cm,
  Size = log(BDTrac$TL_cm)
)
```

12 Visualising Shape Variation

To visualise shape variation among specimens, we use the `plotAllSpecimens()` function from the `geomorph` package. This function overlays all individual landmark configurations and can optionally include the consensus (mean) shape.

```
# Plot with consensus (mean) shape
plotAllSpecimens(Mdf$shape, mean = TRUE)

# Plot without consensus shape
```

```
plotAllSpecimens(Mdf$shape, mean = FALSE)
```

Figure 8 shows the output with the consensus shape included, while Figure 9 illustrates the configuration without the mean shape.

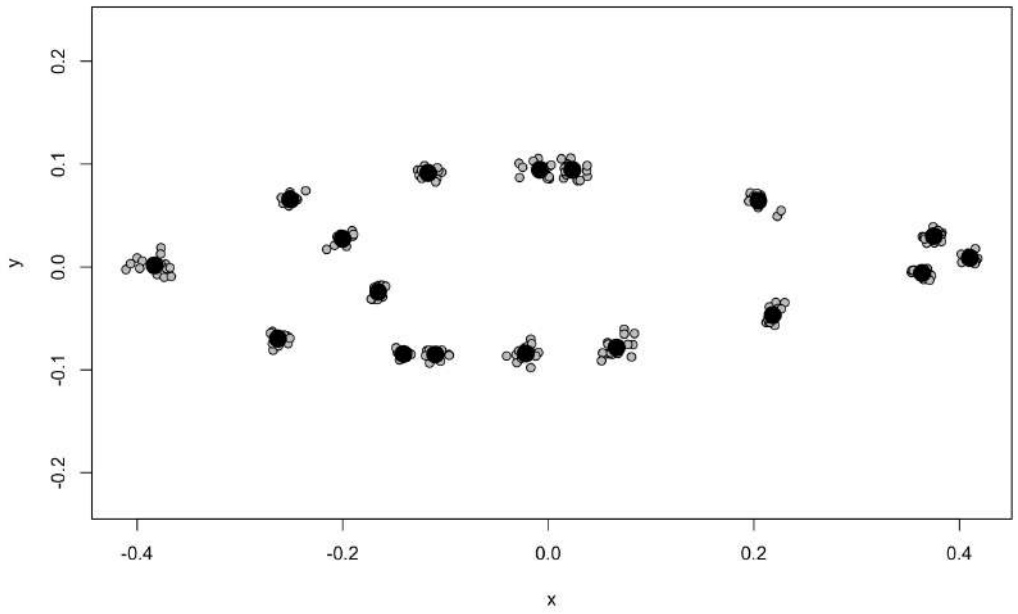

Figure 8: Overlay of all specimens' landmark configurations with the consensus (mean) shape displayed in bold.

13 Evaluating Centroid Size Distribution

It is important to assess whether centroid size is normally distributed before applying statistical tests. This can be done by visualising histograms. In small datasets, a log transformation may improve normality. In larger datasets, it is recommended to construct histograms separately for each level of a grouping variable (e.g., site).

```
# Histogram of raw centroid size
hist(myGPAM$Csize,
     xlab = "Centroid Size",
     ylab = "Number of fish",
```

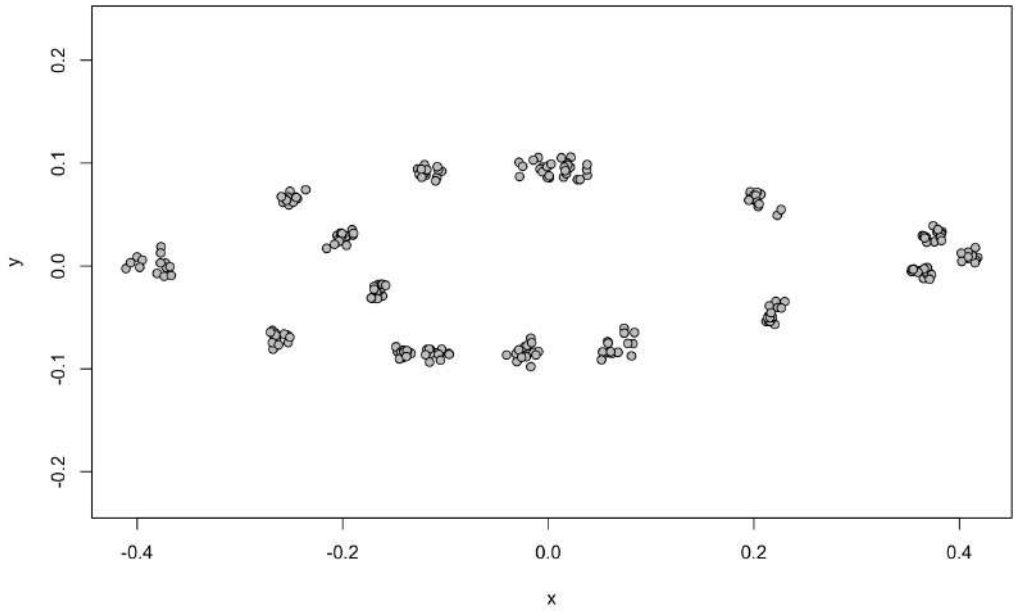

Figure 9: Overlay of all specimens' landmark configurations without the consensus shape.

```
     main = "")

# Histogram of log-transformed centroid size
hist(log(myGPAM$Csize),
     xlab = "Log transformed centroid size",
     ylab = "Number of fish",
     main = "")
```

If the data appear right-skewed, a log transformation is generally recommended. This is particularly important in biological datasets, where size variation often increases with body size. Log transformation stabilises the variance, making the assumptions of subsequent models more robust.

14 Testing Shape Differences: Procrustes ANOVA

To test whether shape differences are associated with biological or experimental factors, you can perform a Procrustes ANOVA using the `procD.lm()`

function from the `geomorph` package. This function allows you to test for the influence of centroid size, location, or other covariates on shape.

```
modelC <- procD.lm(Mdf$shape ~ Mdf$Csize, iter = 999)
anova(modelC)
```

In this example, the model tests whether shape variation is significantly related to centroid size. The argument `iter = 999` defines the number of permutations used in the test.

15 Optional: Removing Allometric Effects

If the Procrustes ANOVA shows that centroid size significantly affects shape, this allometric effect should be removed before testing other biological or environmental factors.

This is done by extracting the residuals from a Procrustes regression model and reconstructing the shape matrix without the size component.

```
# Fit Procrustes regression model
modelM1Allometry <- procD.lm(Mdf$shape ~ Mdf$Csize,
                             logsz = TRUE, iter = 499)
summary(modelM1Allometry)
plot(modelM1Allometry)

# Fit models with and without size effect
modelM1ANOVA <- procD.lm(shape ~ Csize, data = Mdf, RRPP = TRUE)
modelM1NULL  <- procD.lm(shape ~ 1, data = Mdf, RRPP = TRUE)

# Extract residuals and fitted means
shape.resid1 <- arrayspecs(modelM1ANOVA$residuals,
                           p = dim(myGPAM$coords)[1],
                           k = dim(myGPAM$coords)[2])

shape.mean1 <- arrayspecs(modelM1NULL$fitted,
                          p = dim(myGPAM$coords)[1],
                          k = dim(myGPAM$coords)[2])

# Reconstruct adjusted shape matrix (size-free)
```

```
adj.shape1 <- shape.mean1 + shape.resid1

# Build new data frame with adjusted shapes
Mdf2 <- geomorph.data.frame(shape = adj.shape1,
                            cs = myGPAM$Csize,
                           site = BDTrac$Capture_Location,
                            Csize = log(myGPAM$Csize),
                            size = BDTrac$TL_cm,
                            Size = log(BDTrac$TL_cm))
```

Note: Use this adjusted shape data for subsequent analyses when shape is significantly influenced by size (i.e., allometry is present).

16 Testing Shape Differences Between Groups

Once the effect of allometry has been removed, you can test for shape differences between groups (e.g., capture locations). Use the `procD.lm()` function on the adjusted shape matrix.

```
# Test shape differences between sites using size-adjusted shapes
modelInteraction <- procD.lm(Mdf$shape ~ Mdf$site, iter = 999)
anova(modelInteraction)
```

This analysis performs a Procrustes ANOVA with permutation to assess whether groups differ significantly in shape space.

17 Principal Component Analysis (PCA)

We use PCA to summarise shape variation across specimens using the `gm.prcomp()` function from `geomorph`. The first few PCs often explain the majority of variation in shape and can be plotted to visualise group separation.

```
# Perform PCA on GPA-aligned shape data
PCA <- gm.prcomp(Mdf$shape)
summary(PCA)
```

You can colour points by group (e.g., Capture Location) and plot them using base R or `ggplot2`. First, add the PCA scores to the biological dataset:

```
# Merge scores with biological data
pc_scores <- PCA$x
pc_scores1 <- cbind(BDTrac, pc_scores)

# Define colours
location_colors <- c("Coruna" = "#B57BA2",
                     "Gulf of Cadiz" = "#78A8CE")

# Calculate group centroids
centroids <- pc_scores1 %>%
  group_by(Capture_Location) %>%
  summarize(Comp1 = mean(PC1), Comp2 = mean(PC2)) %>%
  as.data.frame()
```

Base R plot (Figure 10):

```
plot(PCA, pch = 21,
    col = location_colors[as.character(BDTrac$Capture_Location)],
     cex = 2, alpha = 0.5)

points(centroids$Comp1, centroids$Comp2, pch = 16,
     col = location_colors[names(table(BDTrac$Capture_Location))],
       cex = 3)
legend("topright", legend = unique(BDTrac$Capture_Location),
       col = unique(location_colors), pch = 16)
```

ggplot2 version (Figure 11):

```
ggplot(pc_scores1, aes(x = PC1, y = PC2)) +
  geom_point(aes(color = Capture_Location),
             size = 7, alpha = 0.5) +
  geom_point(data = centroids,
             aes(x = Comp1, y = Comp2,
             color = as.character(Capture_Location)),
             shape = 16, size = 8) +
  xlab("PC1 (50.18%)") + ylab("PC2 (15.73%)") +
  scale_color_manual(values = location_colors) +
  theme_bw() +
```

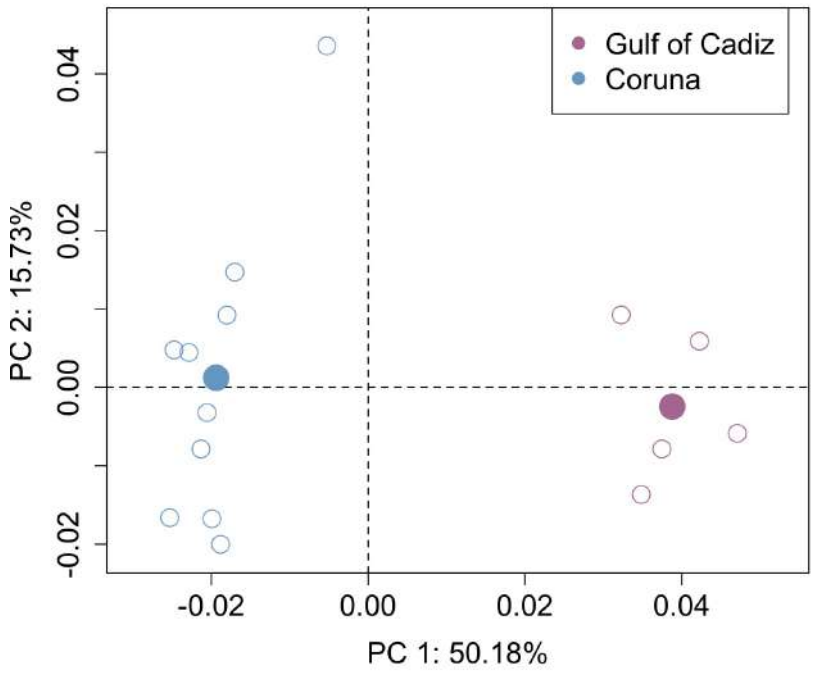

Figure 10: Principal Component Analysis (PCA) plot of GPA-aligned shape data for *Trachurus trachurus*, generated using Base R. Individuals are coloured by capture location: Gulf of Cadiz (purple) and Coruna (blue). Filled circles represent group centroids. PC1 and PC2 account for 50.18% and 15.73% of shape variation, respectively.

```
  theme(legend.position = "bottom",
        legend.title = element_blank(),
        legend.text = element_text(size = 14),
       axis.title = element_text(size = 16, face = "bold"),
        axis.text = element_text(size = 14)) +
  geom_hline(yintercept = 0,
             linetype = "dashed",
             color = "gray") +
  geom_vline(xintercept = 0,
             linetype = "dashed",
             color = "gray")
```

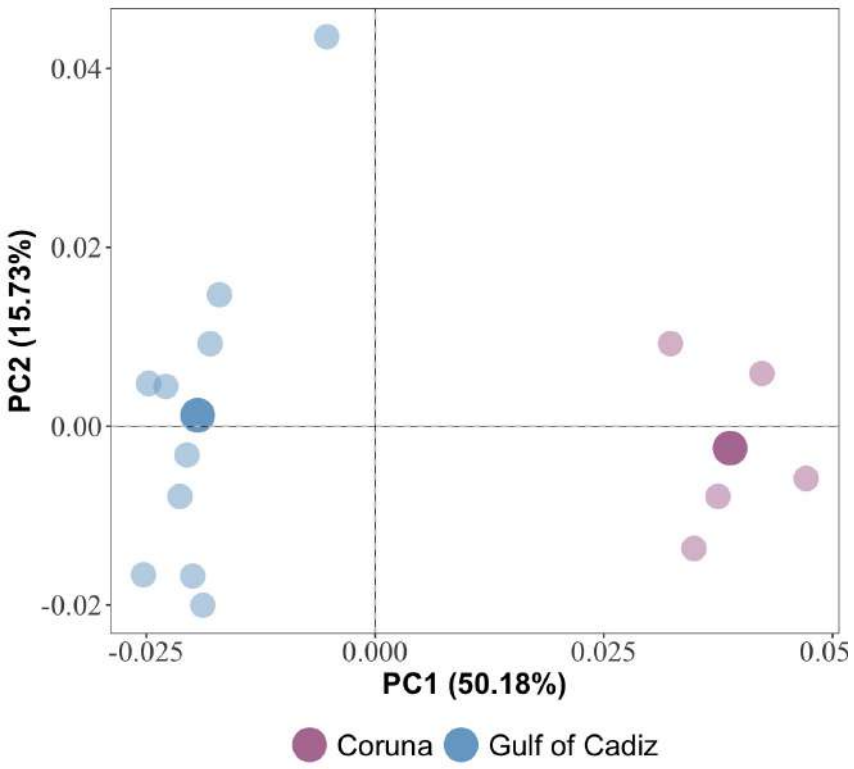

Figure 11: Principal Component Analysis (PCA) plot of GPA-aligned shape data for *Trachurus trachurus*, generated using `ggplot2`. Individuals are coloured by capture location: Coruna (purple) and Gulf of Cadiz (blue). Larger filled circles indicate group centroids. PC1 and PC2 account for 50.18% and 15.73% of shape variation, respectively.

18 Canonical Variate Analysis (CVA)

Canonical variate analysis (CVA) is a multivariate method that maximises group separation and helps assess classification accuracy. To avoid overfitting, the analysis should be restricted to a limited number of principal components (PCs) that explain most of the shape variation (typically 90–95%).

To determine how many PCs to retain, examine the cumulative variance explained in the PCA (see Section 17). Only PCs that account for a substantial proportion of shape variation should be included in the CVA.

```
library(candisc)
```

```
mod <- lm(cbind(Comp1, Comp2, Comp3, Comp4, Comp5, Comp6, Comp7)
~ Capture_Location, data = pc_scores1)
anova(mod)  # Check degrees of freedom

output.candis <- candisc(mod, term = "Capture_Location")
summary(output.candis)
plot(output.candis)

# Change color per locality
plot(output.candis,
     cex.axis=1.7,
     cex.lab=1.7,
     col=location_colors)
```

The resulting canonical variate plot from the `candisc` package is shown in Figure 12. This plot illustrates the separation between groups along the first two canonical axes.

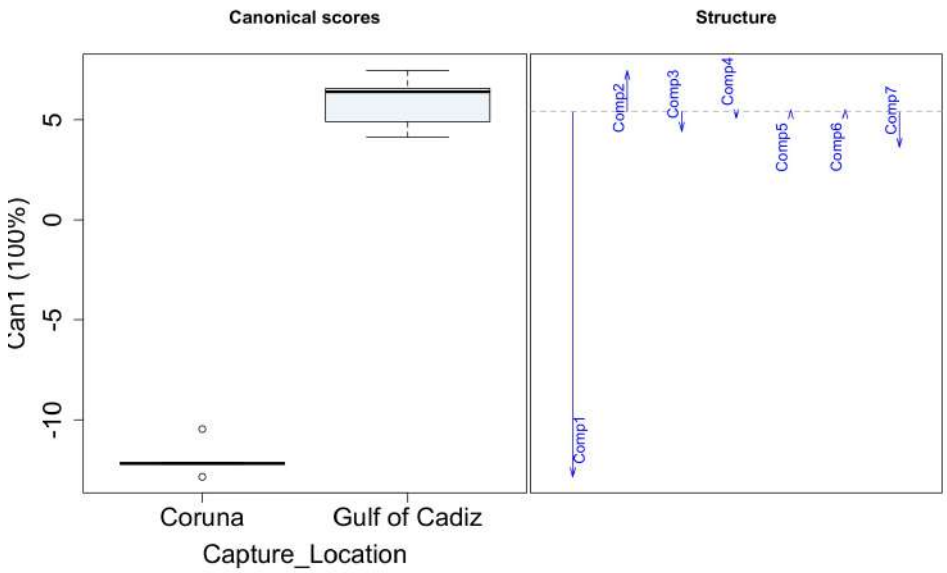

Figure 12: CVA plot using the `candisc` package. Individual specimens are projected onto canonical axes that maximise group separation.

Note: The `candisc` package is useful for visualising group separation but does *not* provide classification performance metrics such as accuracy or

confusion matrices. To assess classification accuracy, you can use the `Morpho` package, which includes cross-validation .

Important: Avoid using all PCs. Your model will fail if the number of dependent variables (PCs) exceeds the residual degrees of freedom. This happens when the number of individuals is smaller than the number of PCs plus the number of groups.

Alternatively, CVA can be performed using the `Morpho` package:

```
library(Morpho)

PCAdata <- data.frame(pc_scores1)
groups <- as.factor(PCAdata$Capture_Location)

cv <- CVA(PCAdata[7:20], PCAdata$Capture_Location,
          cv=TRUE, prior = c(0.3, 0.7))
cv
```

Note: Columns `7:20` correspond to the retained principal components, from `Comp1` to `Comp14`, selected to represent most of the shape variation.

Figure 13 shows an equivalent canonical variate plot produced with the `Morpho` package, which additionally reports classification metrics such as cross−validated accuracy.

The output of the cv object generated by the `Morpho::CVA()` function includes cross-validated classification results by frequency and percentage, overall accuracy, and the Kappa statistic. This output is shown in Figure 14.

19 Thin-Plate Spline Visualisation

Thin-plate spline (TPS) grids are commonly used to visualise shape differences between specimens or groups. This technique interpolates landmark displacements across a regular grid, revealing how shapes deform relative to one another. In `Morpho`, this can be done using the `deformGrid2d()` function.

To begin, calculate the relative warps:

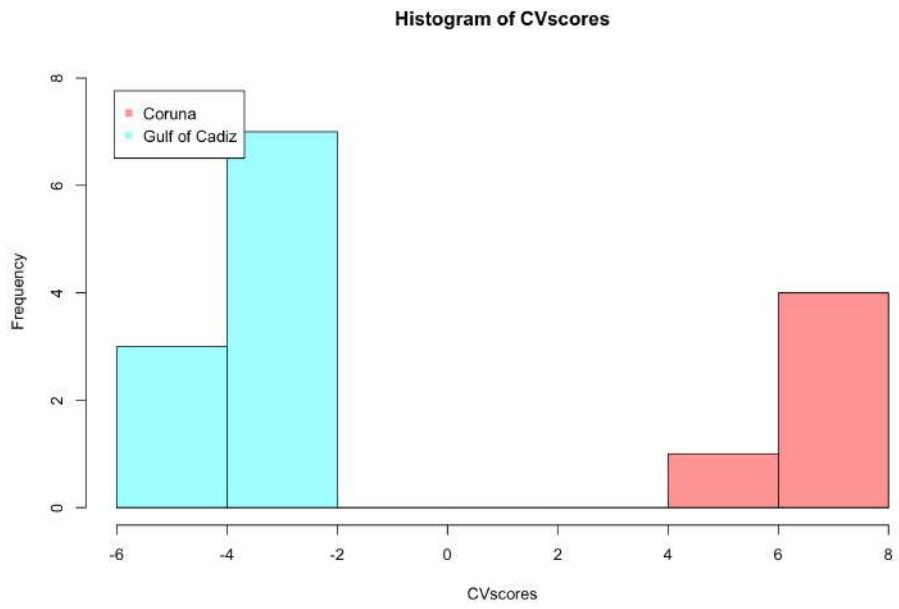

Figure 13: CVA plot using the `Morpho` package. Canonical axes reflect optimal shape separation across capture locations.

```
library(Morpho)

RW <- relWarps(Mdf$shape, scale = TRUE, CSinit = TRUE,
               alpha = 0, orp = TRUE, noalign = TRUE)
RW
```

This produces the relative warp scores and the consensus (mean) shape (`RW$mshape`) to use in deformation visualisation.

To compare shape extremes or group means (Figure 15), use:

```
# Example without deformation grid
deformGrid2d(RW$mshape, PCA$shapes$shapes.comp2$min,
             ngrid = 0, pch = 19, cex1 = 2, cex2 = 2,
             col1 = "red", col2 = 4, gridcol = "black")

lineplot(RW$mshape, point = c(1,10:17,2:7,1,7:10),
         col = "red", lwd = 2)
lineplot(PCA$shapes$shapes.comp2$min,
```

```
> cv
cross-validated classification results in frequencies

               Coruna Gulf of Cadiz
 Coruna             5             0
 Gulf of Cadiz      1             9


cross-validated classification result in %

               Coruna Gulf of Cadiz
 Coruna           100             0
 Gulf of Cadiz     10            90


overall classification accuracy: 93.33333 %

 Kappa statistic: 0.85714
>
```

Figure 14: Cross-validated classification results from the `Morpho::CVA()` function. The confusion matrices indicate classification performance by frequency and percentage, with an overall classification accuracy of 93.3 percent and a Kappa statistic of 0.857.

```
point = c(1,10:17,2:7,1,7:10),
col = 4, lwd = 2)
```

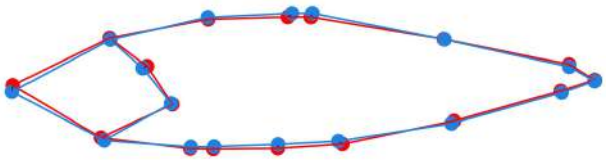

Figure 15: Thin-plate spline comparison without deformation grid. Shape differences between mean shape and the minimum along PC2 are illustrated with landmark connections.

To include the deformation grid (Figure 16), increase `ngrid`:

```
# Example with deformation grid
deformGrid2d(RW$mshape, PCA$shapes$shapes.comp2$min,
             ngrid = 10, pch = 19, cex1 = 2, cex2 = 2,
             col1 = "red", col2 = "#78A8CE",
             gridcol = "black")

lineplot(RW$mshape, point = c(1,10:17,2:7,1,7:10),
         col = "red", lwd = 2)
lineplot(PCA$shapes$shapes.comp2$min,
         point = c(1,10:17,2:7,1,7:10),
         col = "#78A8CE", lwd = 2)
```

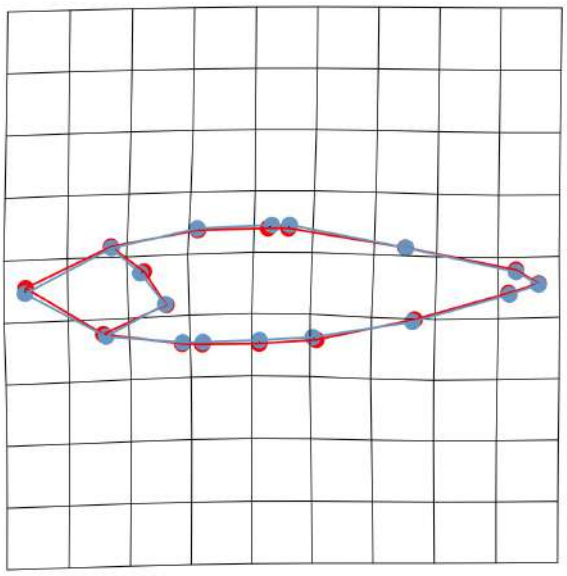

Figure 16: Thin-plate spline visualisation with a deformation grid (`ngrid = 10`). The red shape corresponds to the consensus (mean), while blue represents the minimum shape along PC2.

20 Access to Code and Example Data

You can access the complete R code, sample data, and image templates used in this manual from the following GitHub repository:

`https://github.com/jreisvasconcelos/GeometricMorphometricsManual`

Feel free to clone, adapt, or cite the repository when using this workflow in your own studies.

Help Material and R Packages Used

StereoMorph
Allows precise 2D and 3D landmark digitisation through a Shiny-based interface.
CRAN: https://cran.r-project.org/package=StereoMorph

readxl
Enables direct import of Excel files into R.
CRAN: https://cran.r-project.org/package=readxl

geomorph
Provides functions for statistical analysis of shape, including GPA, PCA, and Procrustes ANOVA.
CRAN: https://cran.r-project.org/package=geomorph

Morpho
Provides tools for manipulating 3D shape data, performing thin-plate spline transformations, and visualising deformation grids.
CRAN: https://cran.r-project.org/package=Morpho

candisc
Performs canonical discriminant analysis on multivariate linear models.
CRAN: https://cran.r-project.org/package=candisc

ggplot2
Enables advanced graphical visualisation of morphometric data.
CRAN: https://cran.r-project.org/package=ggplot2

References

[1] Olsen, A. M., & Westneat, M. W. (2015). StereoMorph: an R package for the collection of 3D landmarks and curves using a stereo camera set-up. Methods in Ecology and Evolution 6(3), 351–356. https://doi.org/10.1111/2041-210X.12326.

[2] Olsen, A. M. & Haber, A. (2022). StereoMorph: Stereo Camera Calibration and Reconstruction. Version 1.6.5. https://CRAN.R-project.org/package=StereoMorph.

[3] Baken, E., Collyer M., Kaliontzopoulou A., Adams D. (2021). geomorph v4.0 and gmShiny: enhanced analytics and a new graphical interface for a comprehensive morphometric experience. Methods in Ecology and Evolution, 12, 2355-2363.

[4] Adams, D., Collyer, M., Kaliontzopoulou, A., Baken, E. (2025). Geomorph: Software for geometric morphometric analyses. R package version 4.0.10. https://cran.r-project.org/package=geomorph.

[5] Schlager S (2017). Morpho and Rvcg - Shape Analysis in R. In Zheng G, Li S, Szekely G (eds.), Statistical Shape and Deformation Analysis, 217-256. Academic Press. ISBN 9780128104934.

[6] Friendly, M. (2007). HE plots for Multivariate General Linear Models. Journal of Computational and Graphical Statistics, 16(2), 421-444. doi:10.1198/106186007X208407 https://doi.org/10.1198/106186007X208407.

[7] Friendly, M., Fox, J. (2024). candisc: Visualizing Generalized Canonical Discriminant and Canonical Correlation Analysis. R package version 0.9.0, https://CRAN.R-project.org/package=heplots.

[8] Vasconcelos, J., Cirera, M., Vieira, A.R., Otero-Ferrer, J.L., Tuset, V.M. (2025a). Application of shape analysis for the identification of pelagic fish stocks. *Hydrobiologia*, 852, 2847–2869. https://doi.org/10.1007/s10750-025-05798-1

Index

Glossary

Alignment The process of translating, rotating, and scaling landmark coordinates so that they can be compared across specimens.

candisc An R package used for generalized canonical discriminant analysis, often applied to visualise group separation in morphometric studies.

Centroid Size A measure of size calculated as the square root of the summed squared distances of each landmark from the centroid of a configuration.

Curve A representation of the contour of a shape, usually captured with semi-landmarks or curves.

Digitising The process of recording landmark coordinates from an image using software tools.

geomorph An R package for geometric morphometric shape analysis in 2D and 3D, including tools for GPA, shape variation modelling, and hypothesis testing.

GPA (Generalized Procrustes Analysis) A superimposition method that removes differences in position, orientation, and scale between specimens.

Landmark A biologically meaningful point that is geometrically homologous across all specimens.

Morpho An R package providing tools for geometric morphometric analysis, including functions for Procrustes superimposition, visualisation, and classification (e.g., CVA).

PCA (Principal Component Analysis) A multivariate statistical method used to reduce the dimensionality of shape data and visualise variation.

Procrustes Distance A measure of shape difference after GPA alignment.

StereoMorph An R package used for digitising landmarks and managing shape data.

Thin-Plate Spline A method for visualising shape differences as deformations of a reference grid.

About this Manual

This manual, titled *Geometric Morphometrics Manual ENG*, was developed by Joana Reis Vasconcelos (PhD), Universidad de Las Palmas de Gran Canaria, to support students and researchers in applying geometric morphometric techniques using R.

For updates, example code, and downloads, please visit:

`https://jreisvasconcelos.github.io`

`https://github.com/jreisvasconcelos/GeometricMorphometricsManual/`

First published: July 2025
Version 1.0