https://lib.uliege.be https://matheo.uliege.be

Identification of the sex, weight, and individuals through 3D modelling of cheetah (Acinonyx jubatus) tracks taking into account their degradation over time, and photogrammetric analysis of cheetah (Acinonyx jubatus) gaits by determining a new trail measuring technique

Auteur : Delvaux, Alicia
Promoteur(s) : 2967; Lejeune, Philippe
Faculté : Gembloux Agro-Bio Tech (GxABT)
Diplôme : Master en bioingénieur : gestion des forêts et des espaces naturels, à finalité spécialisée
Année académique : 2021-2022
URI/URL : http://hdl.handle.net/2268.2/15436

Avertissement à l'attention des usagers :

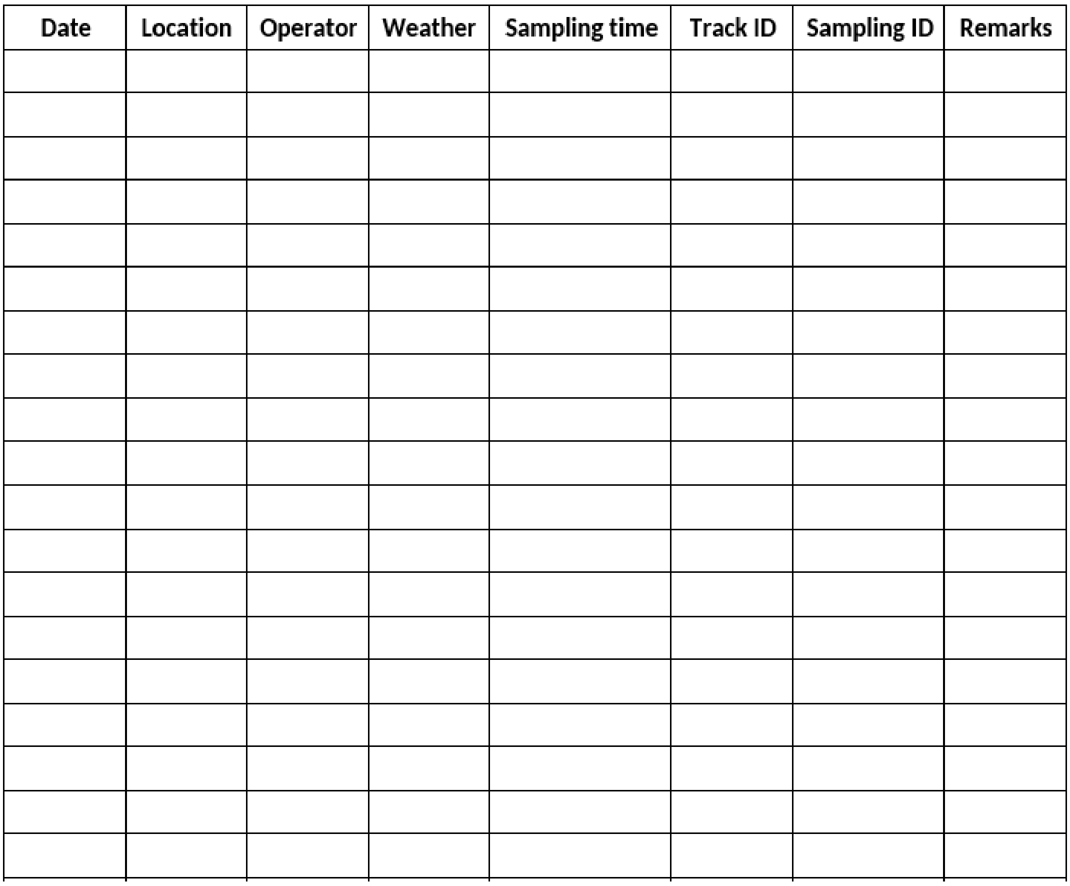
Tous les documents placés en accès ouvert sur le site le site MatheO sont protégés par le droit d'auteur. Conformément aux principes énoncés par la "Budapest Open Access Initiative"(BOAI, 2002), l'utilisateur du site peut lire, télécharger, copier, transmettre, imprimer, chercher ou faire un lien vers le texte intégral de ces documents, les disséquer pour les indexer, s'en servir de données pour un logiciel, ou s'en servir à toute autre fin légale (ou prévue par la réglementation relative au droit d'auteur). Toute utilisation du document à des fins commerciales est strictement interdite.

Par ailleurs, l'utilisateur s'engage à respecter les droits moraux de l'auteur, principalement le droit à l'intégrité de l'oeuvre et le droit de paternité et ce dans toute utilisation que l'utilisateur entreprend. Ainsi, à titre d'exemple, lorsqu'il reproduira un document par extrait ou dans son intégralité, l'utilisateur citera de manière complète les sources telles que mentionnées ci-dessus. Toute utilisation non explicitement autorisée ci-avant (telle que par exemple, la modification du document ou son résumé) nécessite l'autorisation préalable et expresse des auteurs ou de leurs ayants droit.

APPENDIX 1: Field sheets

Date	Start time	Location	Species	Individual ID	Foot position(s)	Number of tracks	Trackset ID	Trail ID	Road status (on or off road)	Surface status (prepared or not)	Slope (flat or with a slope)	Soil type (sandy, clayey, loamy)	Soil humidity (dry or wet)	End Time	Remarks

Figure 13: Field sheet used for track sampling.



Date	Location	Operator	Weather	Sampling time	Track ID	Sampling ID	Remarks

Figure 14: Field sheet used to gather degradation information. Sampling ID represents the number of times the track has been sampled (e.g. if it is being sampled for the second time, sampling ID will be 2).

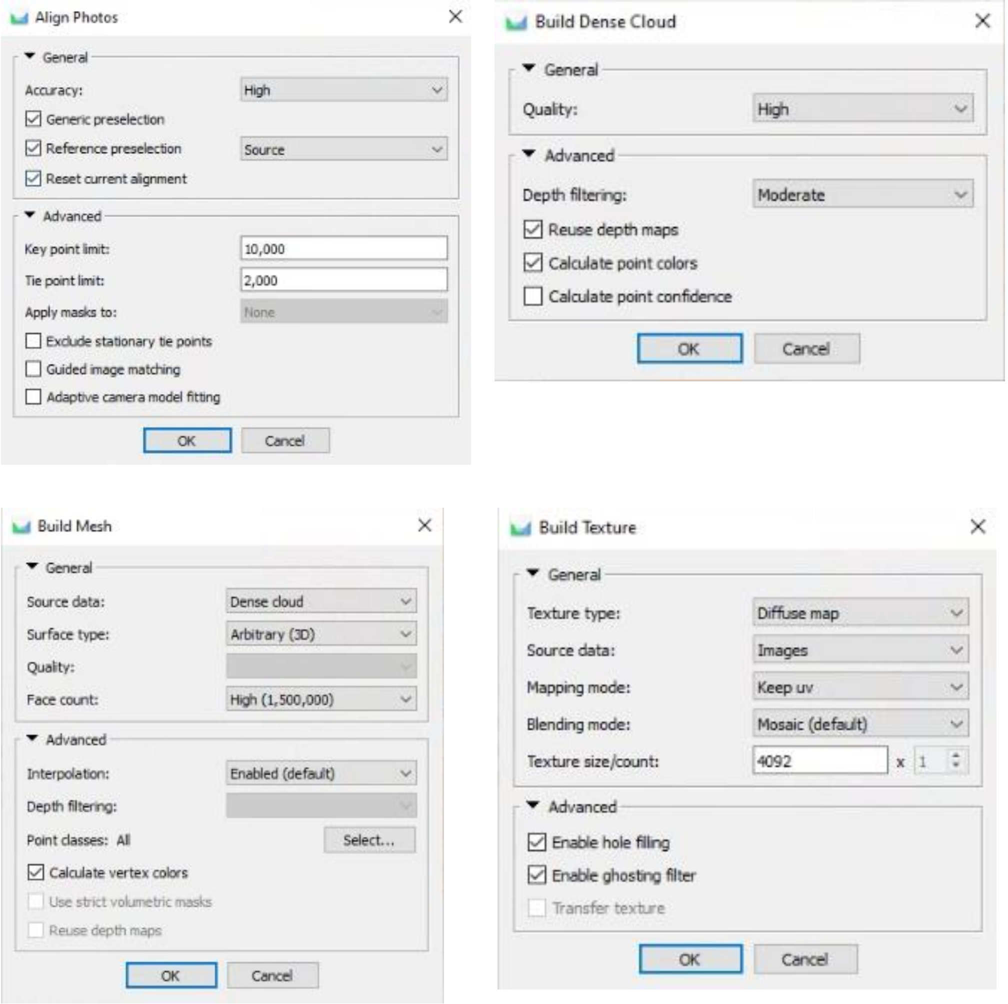
Date	Start time	Location	Species	Individual ID	Trail ID	Road status (on or off road)	Surface status (prepared or not)	Slope (flat or with a slope)	Soil type (sandy, clayey, loamy)	Soil humidity (dry or wet)	Direct measur-ements	Terrestrial photogra-mmetry	Aerial photogra-mmetry	End Time	Remarks

Figure 15: Field sheet used for trail sampling. Direct measurements were realised with a measuring tape, terrestrial photogrammetry with the camera and aerial photogrammetry with the drone.

Date	Location	Operator	Trail ID	Pace (cm)	Stride front feet (cm)	Stride hind feet (cm)	Remarks

Figure 16: Field sheet used for direct trail measurements.

APPENDIX 2: Photogrammetric parameters



Align Photos

General

Accuracy: High

☑ Generic preselection

☑ Reference preselection: Source

☑ Reset current alignment

Advanced

Key point limit: 10,000

Tie point limit: 2,000

Apply masks to: None

☐ Exclude stationary tie points

☐ Guided image matching

☐ Adaptive camera model fitting

OK Cancel

Build Dense Cloud

General

Quality: High

Advanced

Depth filtering: Moderate

☑ Reuse depth maps

☑ Calculate point colors

☐ Calculate point confidence

OK Cancel

Build Mesh

General

Source data: Dense cloud

Surface type: Arbitrary (3D)

Quality:

Face count: High (1,500,000)

Advanced

Interpolation: Enabled (default)

Depth filtering:

Point classes: All Select...

☑ Calculate vertex colors

☐ Use strict volumetric masks

☐ Reuse depth maps

OK Cancel

Build Texture

General

Texture type: Diffuse map

Source data: Images

Mapping mode: Keep uv

Blending mode: Mosaic (default)

Texture size/count: 4092 x 1

Advanced

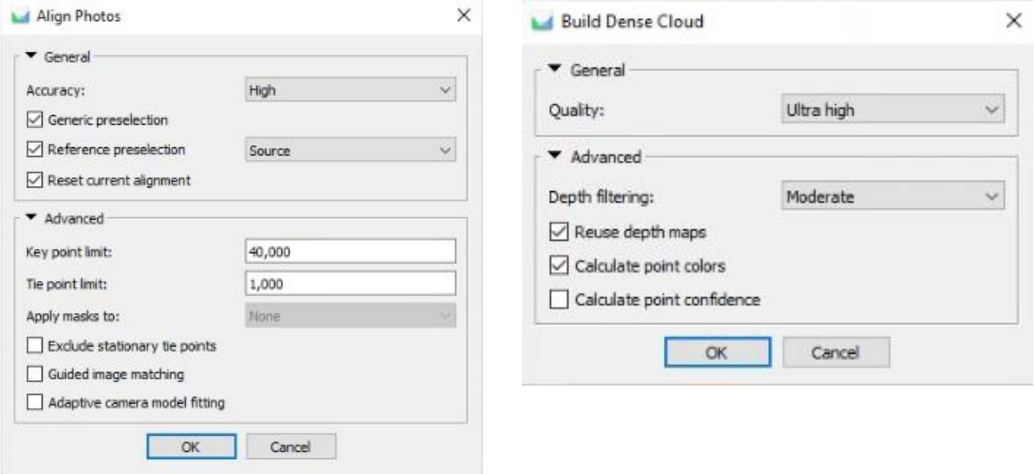
☑ Enable hole filling

☑ Enable ghosting filter

☐ Transfer texture

OK Cancel

Figure 17: Photogrammetric reconstruction parameters in Agisoft Metashape Professional used for tracks modelling.



Align Photos

General

Accuracy: High

☑ Generic preselection

☑ Reference preselection: Source

☑ Reset current alignment

Advanced

Key point limit: 40,000

Tie point limit: 1,000

Apply masks to: None

☐ Exclude stationary tie points

☐ Guided image matching

☐ Adaptive camera model fitting

OK Cancel

Build Dense Cloud

General

Quality: Ultra high

Advanced

Depth filtering: Moderate

☑ Reuse depth maps

☑ Calculate point colors

☐ Calculate point confidence

OK Cancel

Figure 18: Photogrammetric reconstruction parameters in Agisoft Metashape Professional that differ from the ones above for trails modelling.

APPENDIX 3: Automatic modelling Python code

```
Import os, sys
import Metashape
dir_project = 'D:/TFE_Alicia/'
dir_output = dir_project + 'models_smartphone/'
pictures_path = dir_project + 'SMARTPHONE/'
picture_directories = []

for entry in os.listdir(pictures_path):
    if os.path.isdir(os.path.join(pictures_path, entry)):
        picture_directories.append(entry)

#print(picture_directories)
def find_files(SMARTPHONE, pictures_path):
    return [entry.path for entry in os.scandir(SMARTPHONE) if (entry.is_file() and
os.path.splitext(entry.name)[1].lower() in pictures_path)]

def create_model(directory):
    # first, we create the list of photos from the directory
    where_the_pictures_are = pictures_path + "/" + directory + "//"
    photos = find_files(where_the_pictures_are, [".jpg", ".jpeg"])
    print("found these pictures:", photos)
    # then, we create the model
    doc = Metashape.Document()
    chunk = doc.addChunk()
    chunk.addPhotos(photos)
    print(str(len(chunk.cameras)) + " images loaded")
    chunk.matchPhotos(downscale = 1,keypoint_limit = 10000, tiepoint_limit = 2000,
generic_preselection = True, reference_preselection = True)
    chunk.alignCameras()
    chunk.detectMarkers()
    chunk.buildDepthMaps(downscale = 2, filter_mode = Metashape.ModerateFiltering)
    chunk.buildDenseCloud(keep_depth = True)
    chunk.buildModel()
    chunk.buildUV()
    chunk.buildTexture(texture_size = 4096, ghosting_filter = True)
    model_filename = dir_output + '/model_' + directory + '.ply'
    chunk.exportModel(model_filename)
    doc.save(dir_output + '/project' + directory + '.psx')
    print('Processing finished, results saved to ' + model_filename)

for directory in picture_directories:
   create_model(directory)
```

APPENDIX 4a: Fixed landmarks R code

```
library(geomorph)
library(filesstrings)
rm(list=ls())
path0 =
"C:/Users/celin/OneDrive/Documents/TFE/models_segmented/LANDMARKS_FAIL"
setwd(path0)
list_models <- list.files(path0,pattern = "_ALL", recursive = TRUE)
list_models
filename <- list.files(pattern = "_ALL.ply",full.names = FALSE)
filename <- before_last_dot(filename)
filename
filename_table <- as.data.frame(filename)
filename_table
write.csv(filename_table, file = "list_ALL.csv")
# AM1 = KOVU, AM2 = BOLT, AM3 = BULLET, AM4 = FLASH, AM5 = ODYSSEY
# AF1 = AMBER, AF2 = ATHENA, AF3 = SHILOH, AF4 = WONDER
# CODE = ID_cheetah.trailID.trackID.degradationID
AM3.43.25.1 <- read.ply(file = "43.25.1_ALL.ply", ShowSpecimen = TRUE)
digit.fixed(AM3.43.25.1, 25, index = FALSE, ptsize = 1, center = TRUE)
```

APPENDIX 4b: Surface sliders R code

```
library(geomorph)
library(MASS)
rm(list=ls())
setwd("C:/Users/celin/OneDrive/Documents/TFE/Fixed Landmarks/T0/Alicia/FL")
plylist<- list.files(pattern="_ALL.ply")
plylist
filelist<- list.files(pattern=".nts")
filelist
data<-readmulti.nts(filelist)
dim(data)
AM1.45.6.1_surf<-read.ply(plylist[20], ShowSpecimen = FALSE)
buildtemplate(AM1.45.6.1_surf, data[,,25], 130, ptsize = 1, center = TRUE)
AF2.1.3.1_surf<-read.ply(plylist[1], ShowSpecimen = FALSE)
AM1.2.3.1_surf<-read.ply(plylist[2], ShowSpecimen = FALSE)
AM1.2.7.1_surf<-read.ply(plylist[3], ShowSpecimen = FALSE)
AF1.3.7.1_surf<-read.ply(plylist[4], ShowSpecimen = FALSE)
digitsurface(AF2.1.3.1_surf, data[,,4], ptsize = 1, center = TRUE)
digitsurface(AM1.2.3.1_surf, data[,,19], ptsize = 1, center = TRUE)
digitsurface(AM1.2.7.1_surf, data[,,20], ptsize = 1, center = TRUE)
digitsurface(AF1.3.7.1_surf, data[,,1], ptsize = 1, center = TRUE)
```

This are only examples for one track (fixed landmarks) and for four tracks (surface sliders).

APPENDIX 5: LDA R code

```
library(MASS)
library(dplyr)
library(ggplot2)
library(RColorBrewer)
library(scales)
rm(list=ls())
setwd("C:/Users/celin/OneDrive/Documents/TFE/Fixed landmarks/T0/Fusion")
filelist <- list.files(pattern = ".nts")
data<-readmulti.nts(filelist)
surfslider<-as.matrix(read.csv("surfslide.csv", header = TRUE))
classifier<-read.csv ("datatrackpos_t0.csv", header=TRUE , sep= ";")
pos<-classifier$Position
sex<-classifier$Sex
id<-classifier$Name
group_weight<-classifier$Group_weight
y<-gpagen(data, ProcD = TRUE)
PCA<-gm.prcomp(y$coords)
plot(PCA, main = "PCA")
test<-as.data.frame(PCA$x)
classifier$colsex[classifier$Sex == "F"] <- "yellow"
classifier$colsex[classifier$Sex == "M"] <- "lightblue"
ggplot(test, aes(x = Comp1, y = Comp2, colour = classifier$colsex))+
geom_point(size=2)+
scale_color_brewer(name = "",aesthetics = "colour", palette = "Pastel1", guide ="legend",
                breaks=c("magenta","yellow"),labels=c("Female","Male"))+
scale_y_continuous(name="Principal Component 2 (9.49%)")+
labs(x = "Principal Component 1 (11.95%)")+
ggtitle("Principal Component analysis coloured by sexes")+
theme_bw()+
theme_light()
dataPC <- read.csv("ListePC.csv", header = FALSE, sep= ";", encoding="UTF-8")
dataPC
###SHAPE
results=NULL
results<-as.list(results)
b<-("sex ~")
for (i in 1:68) {
  a<-as.character(dataPC[i,2])
  c<-paste(b,a)
  c<-as.formula(c)
  lda<-lda(c, CV= TRUE)
  ct<- table(sex,lda$class)
  diag(prop.table(ct,1))
  results[i]<-sum(diag(prop.table(ct)))
}
```

```
results<-as.data.frame(results)
results<-t(results)
write.table(results, "shape_sex.csv", sep=";")
#### FORM
results=NULL
results<-as.list(results)
for (i in 1:68) {
  a<-as.character(dataPC[i,2])
  d<-("y$Csize+")
  c<-paste(b,d,a)
  c<-as.formula(c)
  lda<-lda(c, CV= TRUE)
  ct<- table(sex,lda$class)
  diag(prop.table(ct,1))
  results[i]<-sum(diag(prop.table(ct)))
}
results<-as.data.frame(results)
results<-t(results)
write.table(results, "form_sex.csv", sep=";")
```

This is an example for sex identification.

APPENDIX 6: Consensus track differences between each data set

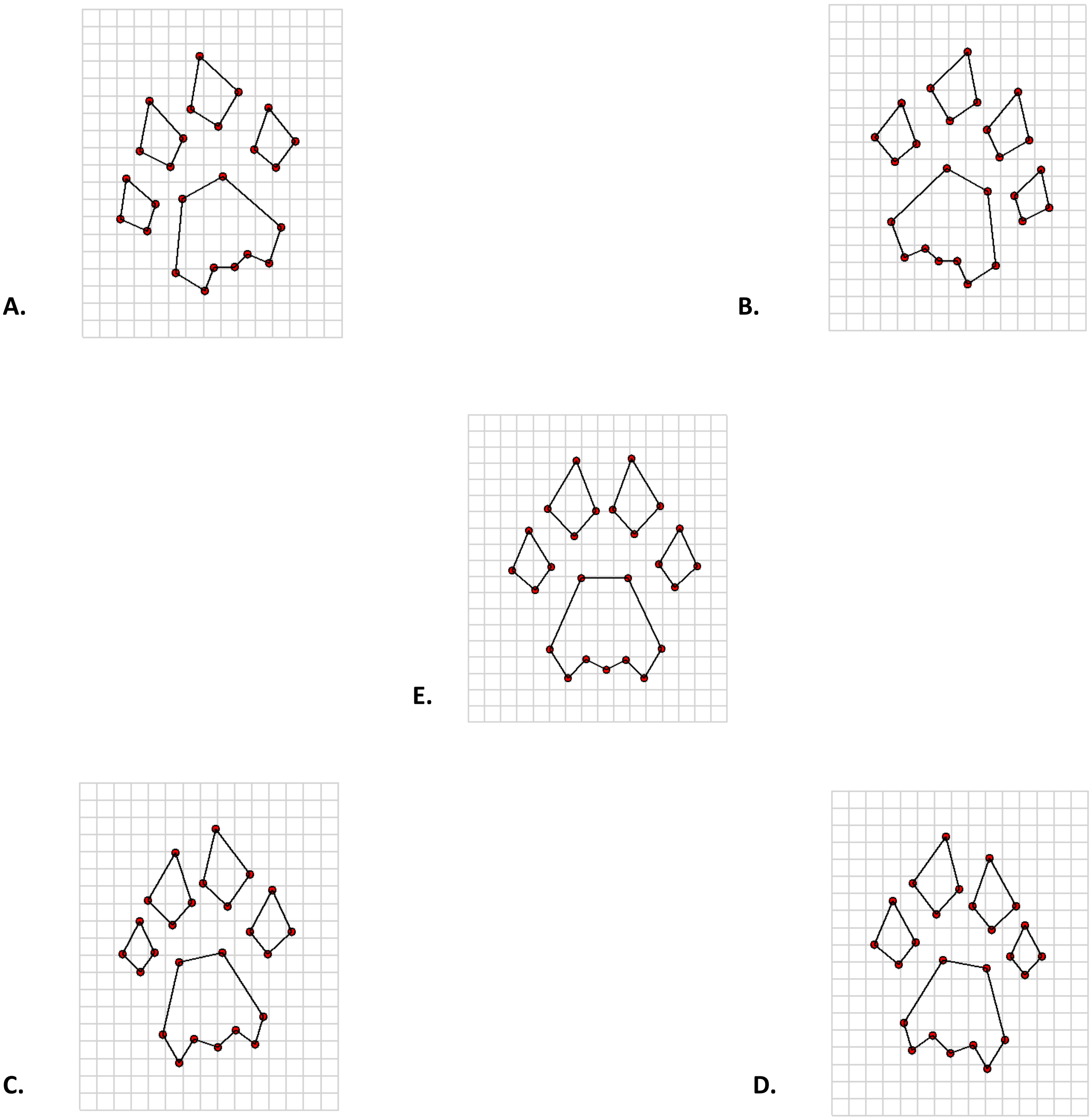


Figure 19: Consensus tracks for each case study using 3D fixed landmarks only.

- *A. Consensus track for Front Left data set (n=202).*
- *B. Consensus track for Front Right data set (n=202).*
- *C. Consensus track for Hind Left data set (n=217).*
- *D. Consensus track for Hind Right data set (n=210).*
- *E. Consensus track for whole database (n=831).*

The GPA produces a consensus track. For the entire database, this consensus appears perfectly symmetrical. It is directly visible that left-right distinction is clearer than front-hind distinction as it presents asymmetry against elongation differences. Therefore, dividing database into four data sets is justified to position characteristics and highlight other characteristics inherent to other factors.

APPENDIX 7: PC1 and PC2 variation illustrated by identity, sex and weight for position data sets

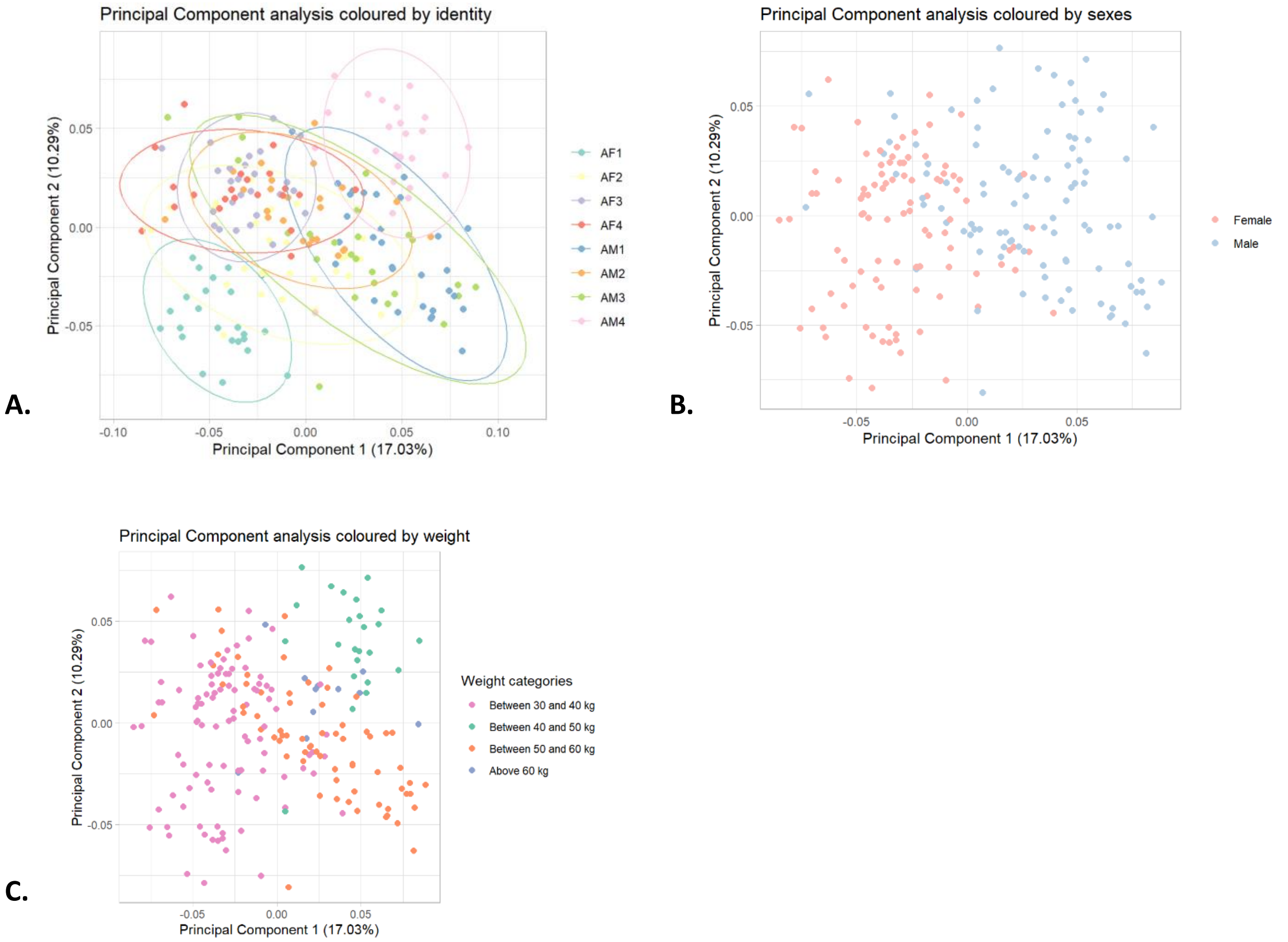
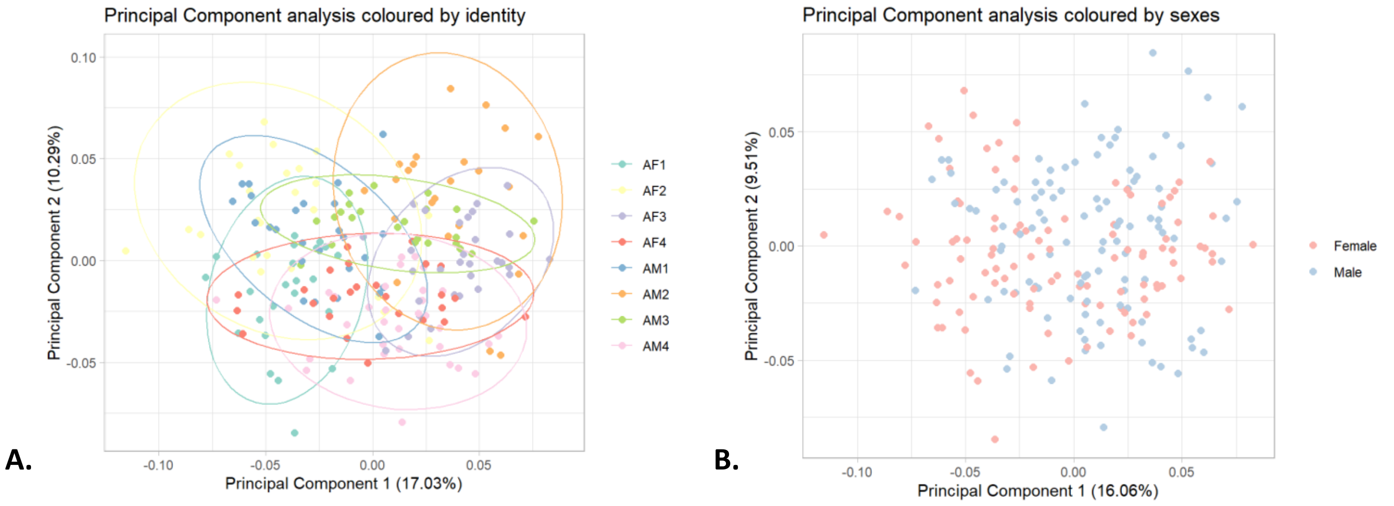


Figure 20: Principal Component Analysis (PCA) of Front Left (FL) tracks from combined database from 2019 and 2022 (n=202) with fixed landmarks only in 3D and form information coloured by A. the identity of the cheetah that produced the track (AF1 = Adult Female 1; AF2 = Adult Female 2; AF3 = Adult Female 3, AF4 = Adult Female 4; AM1 = Adult Male 1; AM2 = Adult Male 2; AM3 = Adult Male 3; AM4 = Adult Male 4), B. the sex of the cheetah that produced the track or C. the weight of the cheetah that produced the track.



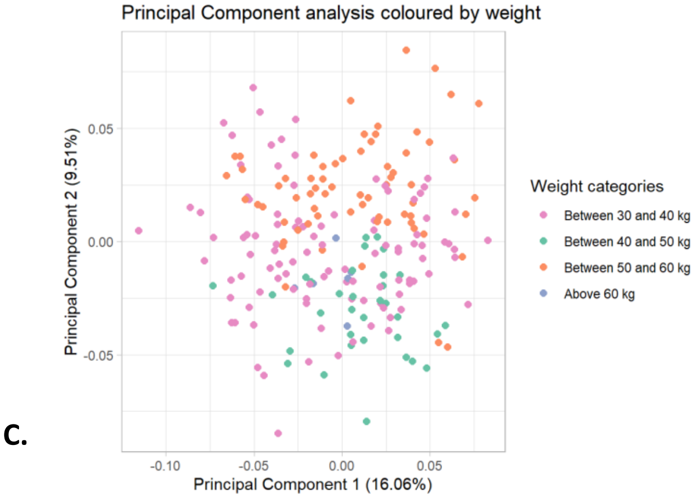


Figure 21: Principal Component Analysis (PCA) of Hind Right (HR) tracks from combined database from 2019 and 2022 (n=210) with fixed landmarks only in 3D and form information coloured by A. the identity of the cheetah that produced the track (AF1 = Adult Female 1; AF2 = Adult Female 2; AF3 = Adult Female 3, AF4 = Adult Female 4; AM1 = Adult Male 1; AM2 = Adult Male 2; AM3 = Adult Male 3; AM4 = Adult Male 4), B. the sex of the cheetah that produced the track or C. the weight of the cheetah that produced the track.

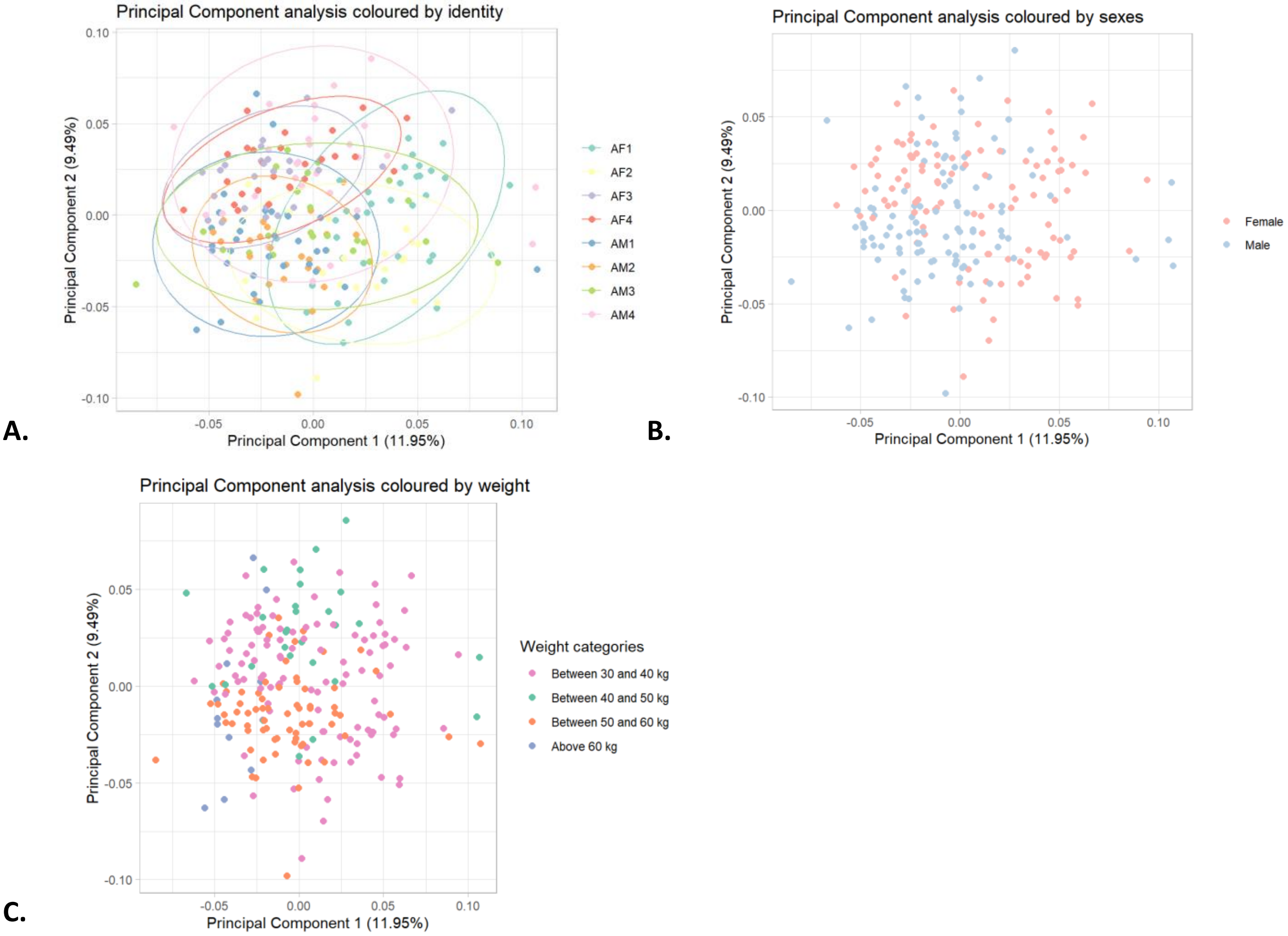
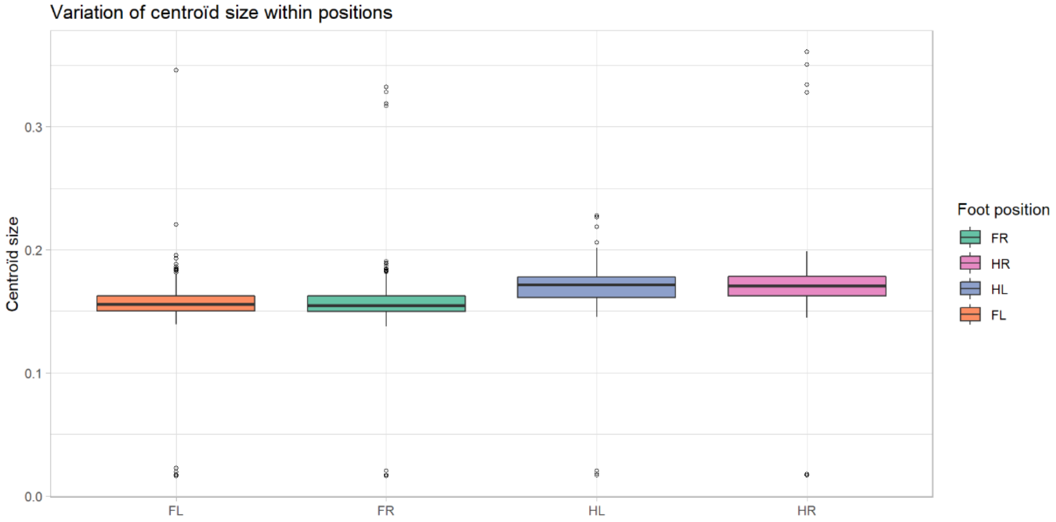
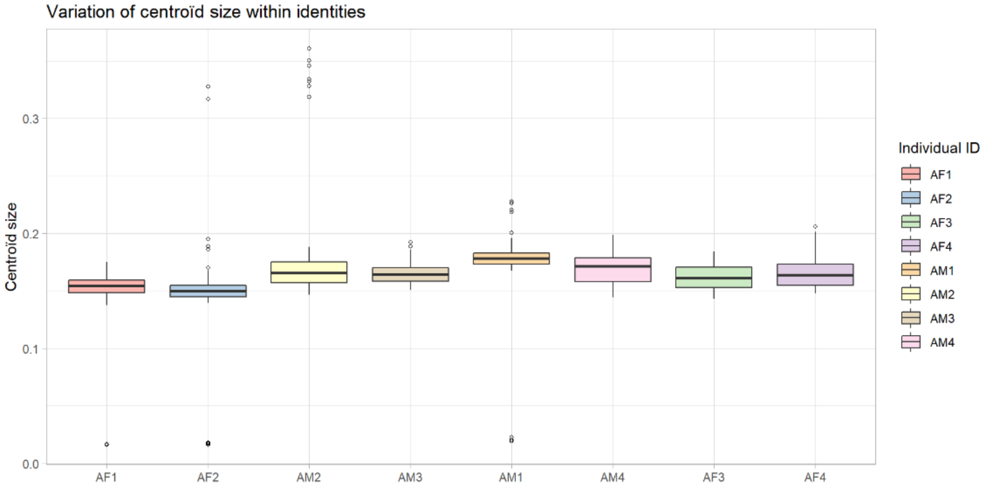


Figure 22: Principal Component Analysis (PCA) of Hind Left (HL) tracks from combined database from 2019 and 2022 (n=210) with fixed landmarks only in 3D and form information coloured by A. the identity of the cheetah that produced the track (AF1 = Adult Female 1; AF2 = Adult Female 2; AF3 = Adult Female 3, AF4 = Adult Female 4; AM1 = Adult Male 1; AM2 = Adult Male 2; AM3 = Adult Male 3; AM4 = Adult Male 4), B. the sex of the cheetah that produced the track or C. the weight of the cheetah that produced the track.

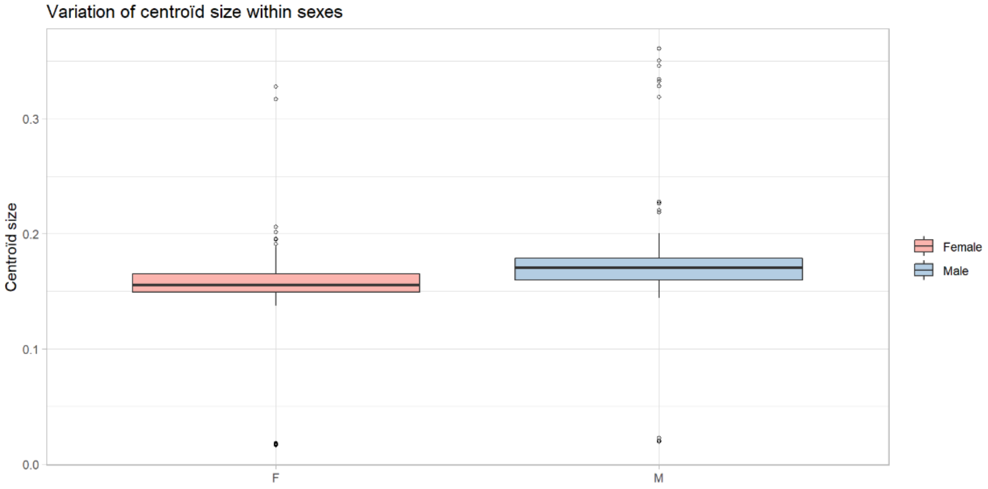
APPENDIX 8: Variation of centroïd size within factors



A.



B.



C.

D.

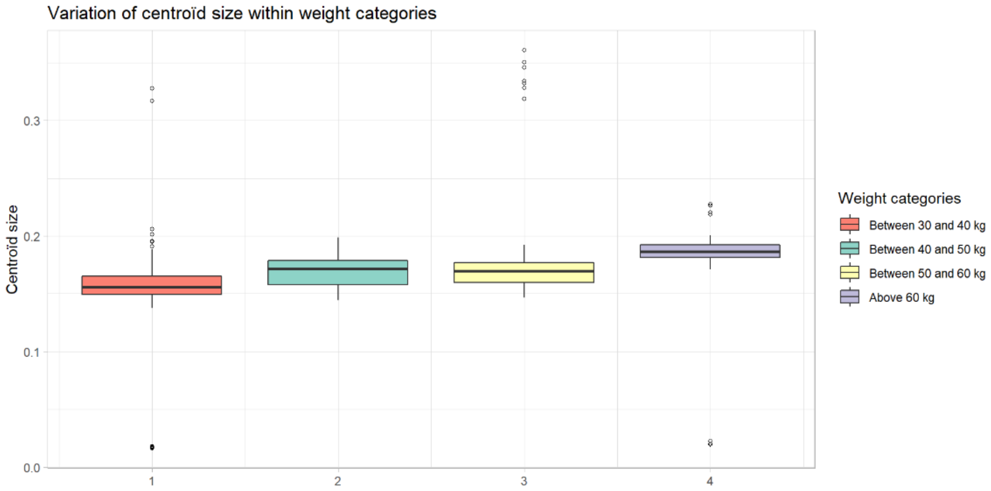


Figure 23: Box plot of centroid sizes for combined database (i.e., from 2019 and 2022, n=831) with 3D fixed landmark only grouped by A. position, B. sex, C. identity or D. weight category.

Centroïd size intervenes during LDA using form information. It is combined to shape information to increase prediction accuracy. Variations in centroïd sizes does not differ significantly but small differences are noticeable for sex and identity. Concerning position, the centroïd sizes are the same for both front positions and for both hind positions. It is explained by the elongation of tracks. Hind tracks are longer than front ones.

APPENDIX 9: Weight study

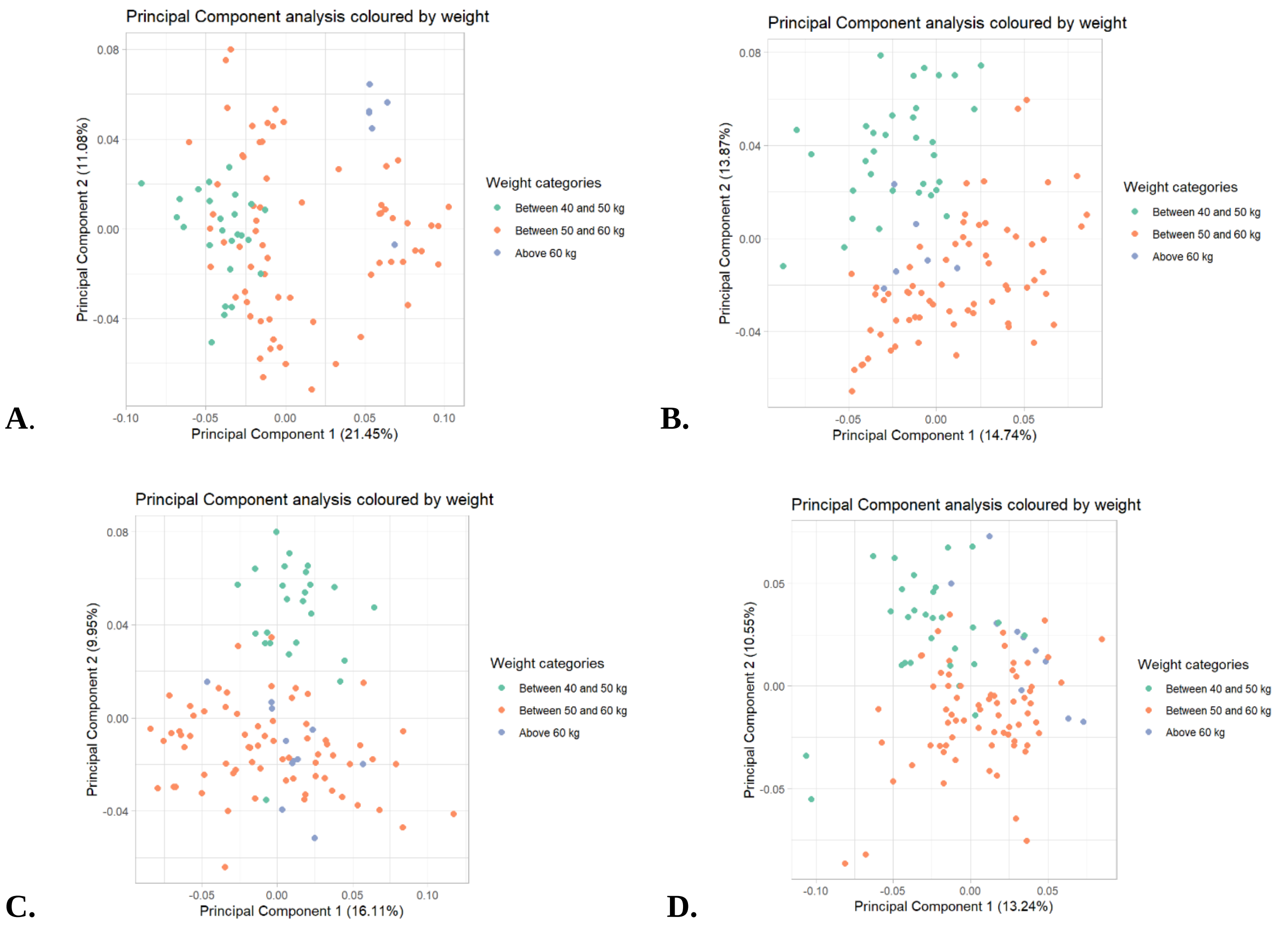


Figure 24: Principal Component Analysis (PCA) of position-divided data sets with 3D fixed landmarks only and form information coloured by weight categories. A. is for Front Left (FL, n=104) data set, B. is for Front Right (FR, n=98) data set, C. is for Hind Right (HR, n=106) data set and D. is for Hind Left (HL, n=108) data set.

For better weight evaluation, separated male weight categories were also studied because females presented little variation thus resulting in all belonging to the same weight category. Prediction accuracies were higher for position and weight-subdivided data sets.

APPENDIX 10: Trails analysis R code

```
library(utils)
library(tidyverse)
library(ggplot2)
library(ggpmisc)
library(ggpubr)
library(RColorBrewer)
rm(list=ls())
path0 = "C:/Users/celin/OneDrive/Documents/TFE/Trails_distances"
setwd(path0)
direct <- read.csv("direct_measurements.csv", sep = ";",dec = ",")
drone <- read.csv("distances_drone.csv", sep = ";", dec = ",")
cam <- read.csv("distances_trails.csv", sep = ";", dec = ",")
drone <- drone[,-c(2,3)]
#Change direct from cm to m
direct$V1 = direct$V1/100
direct$V2 = direct$V2/100
direct$V3 = direct$V3/100
Full_data <- cbind(direct,drone,cam)
names(Full_data)
names(Full_data) <- c("Trail1","Pace_direct","Stride_front_direct","Stride_hind_direct",
                      "Trail2","Pace_drone","Stride_front_drone","Stride_hind_drone",
                      "Trail3","Pace_cam","Stride_front_cam","Stride_hind_cam")
cols <- brewer.pal(n=2, name = "RdBu")
shapes <- c("1","2")
plot <- ggplot(Full_data, aes(x = Pace_direct, y = Pace_cam))+
  geom_point(aes(colour = cols[1],shape = shapes[1]), size = 2, na.rm = TRUE) +
  geom_point(aes(x = Stride_front_direct, y = Stride_front_cam, colour = cols[1], shape =
shapes[2]), size = 2, na.rm = TRUE)+
  geom_point(aes(x = Stride_hind_direct, y = Stride_hind_cam, colour = cols[1], shape =
shapes[2]), size = 2, na.rm = TRUE)+
  geom_point(aes(x = Pace_direct, y = Pace_drone, colour = cols[2], shape = shapes[1]), size
= 2, na.rm = TRUE)+
  geom_point(aes(x = Stride_front_direct, y = Stride_front_drone, colour = cols[2], shape =
shapes[2]), size = 2, na.rm = TRUE)+
  geom_point(aes(x = Stride_hind_direct, y = Stride_hind_drone, colour = cols[2], shape =
shapes[2]), size = 2, na.rm = TRUE)

plot + geom_smooth(method=lm,se=FALSE,fullrange=TRUE, size = 0.5, colour = "black") +
  scale_y_continuous(name = "Camera measurements", sec.axis = sec_axis(~., name = "Drone
measurements"))+
  labs(x = "Direct measurements")+
  scale_colour_discrete(name = "Measurement technique", breaks=c("#EF8A62",
"#F7F7F7"), labels=c("Camera", "Drone"))+
  scale_shape_discrete(name = "Type of measure", breaks = c("1","2"),
labels=c("Pace","Stride"))+
```

```
  theme(legend.title = element_text(face="bold"))+
  stat_regline_equation(label.x = 0.3, label.y = 1.4, aes(label = ..rr.label..), cex = 3)+
  theme_bw()+
  theme_light()

# Spearman test for all measures
cor_pace1 <- cor.test(Full_data$Pace_direct, Full_data$Pace_cam, method = 'spearman')
cor_pace1
cor_pace2 <- cor.test(Full_data$Pace_direct, Full_data$Pace_drone, method = 'spearman')
cor_pace2
cor_pace3 <- cor.test(Full_data$Pace_cam, Full_data$Pace_drone, method = 'spearman')
cor_pace3
cor_stride_front1 <- cor.test(Full_data$Stride_front_direct, Full_data$Stride_front_cam,
method = 'spearman')
cor_stride_front1
cor_stride_front2 <- cor.test(Full_data$Stride_front_direct, Full_data$Stride_front_drone,
method = 'spearman')
cor_stride_front2
cor_stride_front3 <- cor.test(Full_data$Stride_front_cam, Full_data$Stride_front_drone,
method = 'spearman')
cor_stride_front3
cor_stride_hind1 <- cor.test(Full_data$Stride_hind_direct, Full_data$Stride_hind_cam,
method = 'spearman')
cor_stride_hind1
cor_stride_hind2 <- cor.test(Full_data$Stride_hind_direct, Full_data$Stride_hind_drone,
method = 'spearman')
cor_stride_hind2
cor_stride_hind3 <- cor.test(Full_data$Stride_hind_cam, Full_data$Stride_hind_drone,
method = 'spearman')
cor_stride_hind3
data.frame(cor_pace1$statistic,cor_pace2$statistic,cor_pace3$statistic)
data.frame(cor_pace1$p.value,cor_pace2$p.value,cor_pace3$p.value)
line1 <- cbind(cor_pace1$estimate, cor_pace2$estimate, cor_pace3$estimate)
line2 <- cbind(cor_stride_front1$estimate, cor_stride_front2$estimate,
cor_stride_front3$estimate)
line3 <- cbind(cor_stride_hind1$estimate, cor_stride_hind2$estimate,
cor_stride_hind3$estimate)
Spearman_test <- rbind(line1,line2,line3)
write.csv(Spearman_test,"Spearman_test.csv",row.names = FALSE, col.names =
c("Correlation between direct and camera measurements", "Correlation between direct and
drone measurements", "Correlation between camera and drone measurements"))

#Mean calculations
direct_total <- read.csv("direct_total.csv", sep = ";", dec = ".")
names(direct_total) <- c("Cheetah","Sex","Trail","Pace","Stride_front","Stride_hind")

mean(direct_total$Pace,na.rm = TRUE)
```

```
sd <- sd(direct_total$Pace)
margin <- qt(0.975,df=(842-1))*sd/sqrt(842)
margin
lower_interval <- 31.53919 - margin
upper_interval <- 31.53919 + margin
# PACE = 31.53919 +- 0.4911935

mean(direct_total$Stride_front,na.rm = TRUE)
sd_sf <- sd(direct_total$Stride_front, na.rm = TRUE)
margin_sf <- qt(0.975,df=(842-1))*sd_sf/sqrt(842)
margin_sf
lower_interval_sf <- 122.3786 - margin_sf
upper_interval_sf <- 122.3786 + margin_sf
# STRIDE FRONT = 122.3786 +- 0.4879389

mean(direct_total$Stride_hind,na.rm = TRUE)
sd_sh <- sd(direct_total$Stride_hind, na.rm = TRUE)
margin_sh <- qt(0.975,df=(842-1))*sd_sh/sqrt(842)
margin_sh
lower_interval_sh <- 122.3282 - margin_sh
upper_interval_sh <- 122.3282 + margin_sh
# STRIDE HIND = 122.3282 +- 0.5134431

direct_total_F <- filter(direct_total,direct_total$Sex == "F")

mean(direct_total_F$Pace,na.rm = TRUE)
sd_F1 <- sd(direct_total_F$Pace)
margin_F1 <- qt(0.975,df=(381-1))*sd_F1/sqrt(381)
margin_F1
lower_interval_F1 <- 29.85302 - margin_F1
upper_interval_F1 <- 29.85302 + margin_F1
# PACE = 29.85302 +- 0.6351125

mean(direct_total_F$Stride_front,na.rm = TRUE)
sd_F2 <- sd(direct_total_F$Stride_front, na.rm = TRUE)
margin_F2 <- qt(0.975,df=(381-1))*sd_F2/sqrt(381)
margin_F2
lower_interval_F2 <- 119.9362 - margin_F2
upper_interval_F2 <- 119.9362 + margin_F2
# STRIDE FRONT = 119.9362 +- 0.6927735

mean(direct_total_F$Stride_hind,na.rm = TRUE)
sd_F3 <- sd(direct_total_F$Stride_hind, na.rm = TRUE)
margin_F3 <- qt(0.975,df=(381-1))*sd_F3/sqrt(381)
margin_F3
lower_interval_F3 <- 119.7015 - margin_F3
upper_interval_F3 <- 119.7015 + margin_F3
```

```
# STRIDE HIND = 119.7015 +- 0.7139147

direct_total_M <- filter(direct_total,direct_total$Sex == "M")

mean(direct_total_M$Pace,na.rm = TRUE)
sd_M1 <- sd(direct_total_M$Pace)
margin_M1 <- qt(0.975,df=(461-1))*sd_M1/sqrt(461)
margin_M1
lower_interval_M1 <- 32.93275 - margin_M1
upper_interval_M1 <- 32.93275 + margin_M1
# PACE = 32.93275 +- 0.7046104

mean(direct_total_M$Stride_front,na.rm = TRUE)
sd_M2 <- sd(direct_total_M$Stride_front, na.rm = TRUE)
margin_M2 <- qt(0.975,df=(461-1))*sd_M2/sqrt(461)
margin_M2
lower_interval_M2 <- 124.3875 - margin_M2
upper_interval_M2 <- 124.3875 + margin_M2
# STRIDE FRONT = 124.3875 +- 0.6281011

mean(direct_total_M$Stride_hind,na.rm = TRUE)
sd_M3 <- sd(direct_total_M$Stride_hind, na.rm = TRUE)
margin_M3 <- qt(0.975,df=(461-1))*sd_M3/sqrt(461)
margin_M3
lower_interval_M3 <- 124.5284 - margin_M3
upper_interval_M3 <- 124.5284 + margin_M3
# STRIDE HIND = 124.5284 +- 0.6682241
```

APPENDIX 11: Kolmogorov-Smirnov Test to assess normal distribution.

Table 7: P-values obtained for female trail distances (direct measurements) population and male trail distances (direct measurements) population by performing a Kolmogorov-Smirnov Test. Normality was determined for each measured variable (i.e., Pace, Stride Front and Stride Hind).

	Female	Male
Pace	0,363	0,00348
Stride front	0,0603	0,00357
Stride hind	0,288	0,000156

Female population is normally distributed while male population rejects the hypothesis of normal distribution. It is therefore not allowed to perform a Two-Sampled T test to compare female gait characteristics means to male gait characteristics means. A Mann-Whitney test will thus be carried out to highlight sex discrimination for gait characteristics.