



HAL
open science

Automatisation de la pose de landmarks sur des meshes 3D en morphométrie grâce à une approche de Deep Learning

Guillaumot Charlène, Sklab Youcef, Dubied Morgane, Laffont Rémi, Nicolas Navarro

► To cite this version:

Guillaumot Charlène, Sklab Youcef, Dubied Morgane, Laffont Rémi, Nicolas Navarro. Automatisation de la pose de landmarks sur des meshes 3D en morphométrie grâce à une approche de Deep Learning. Journée Deep Learning pour la Science, May 2024, Paris, France. hal-04590087

HAL Id: hal-04590087

<https://hal.science/hal-04590087>

Submitted on 28 May 2024

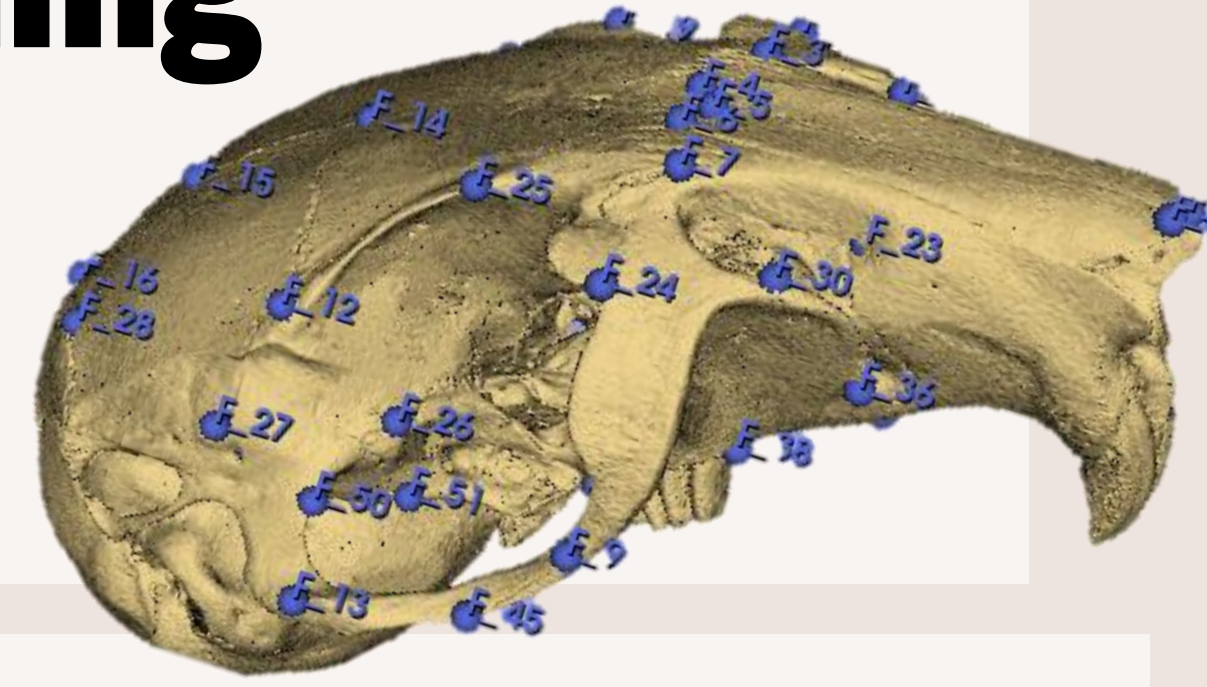
HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Automatisation de la pose de landmarks sur des meshes 3D en morphométrie grâce à une approche de Deep Learning

GUILLAUMOT Charlene¹, SKLAB Youcef², DUBIED Morgane¹, LAFFONT Rémi¹, NAVARRO Nicolas^{1,3}

1. Biogéosciences, UMR 6282 CNRS, Université de Bourgogne, 6 Boulevard Gabriel, 21000 Dijon, France
2. IRD, Sorbonne Université, UMMISCO, F-93143, Bondy, France
3. EPHE, PSL University, 75014 Paris, France

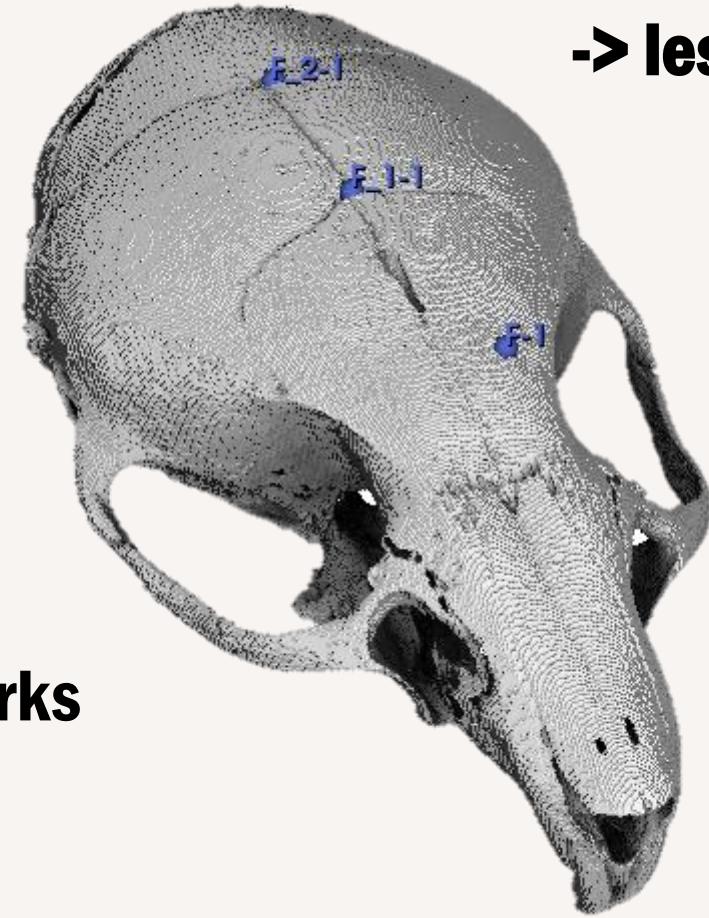


CONTEXTE

Analyse de meshes 3D en morphométrie



Implique la pose de landmarks (manuellement)

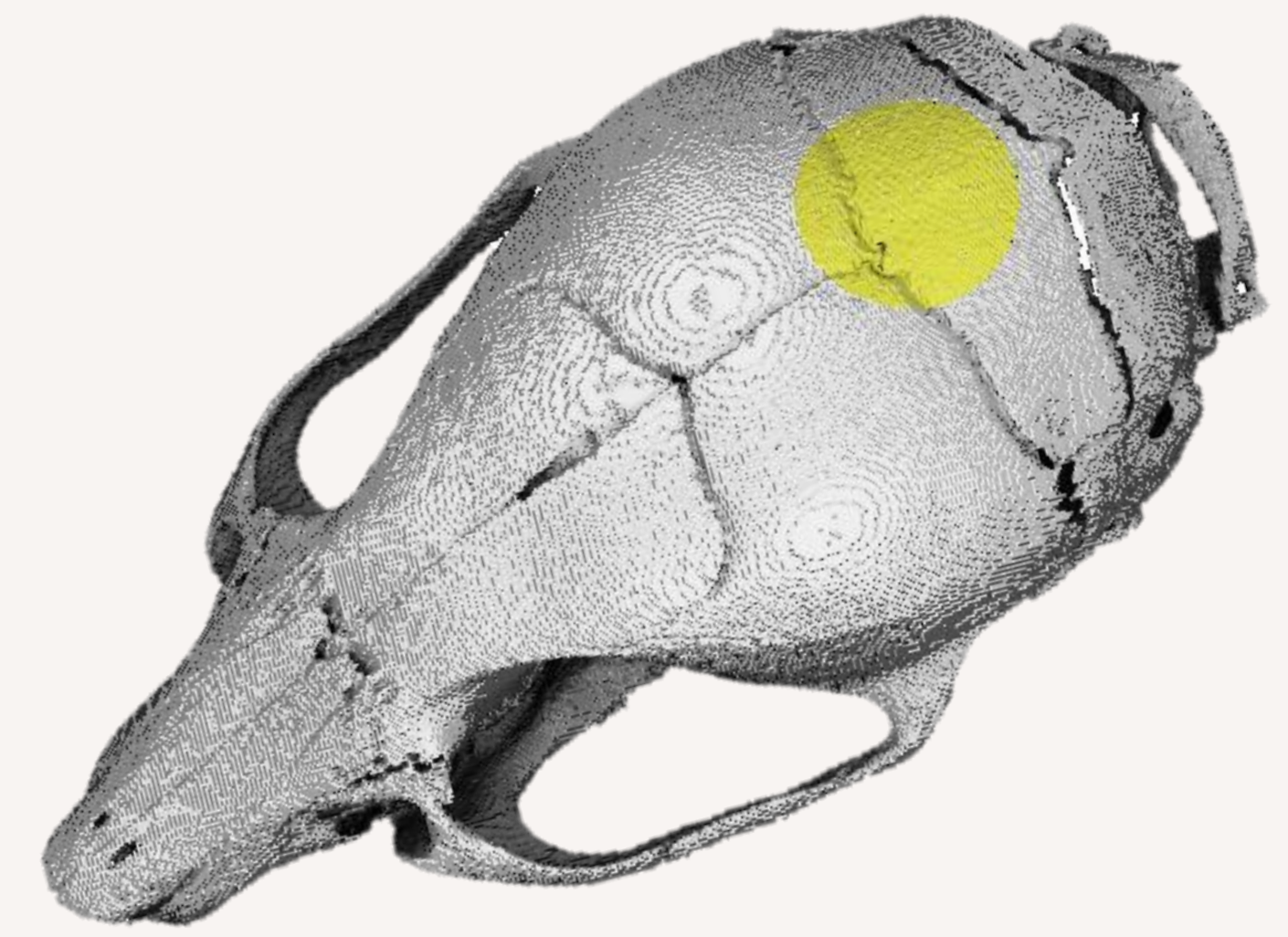


Il existe une application logiciel (ALPACA, Slicer) qui permet de faire des prédictions de position de landmarks, par la déformation d'une image source transformée en nuage de points vers une image cible avec des prédictions, -> les prédictions sont imparfaites

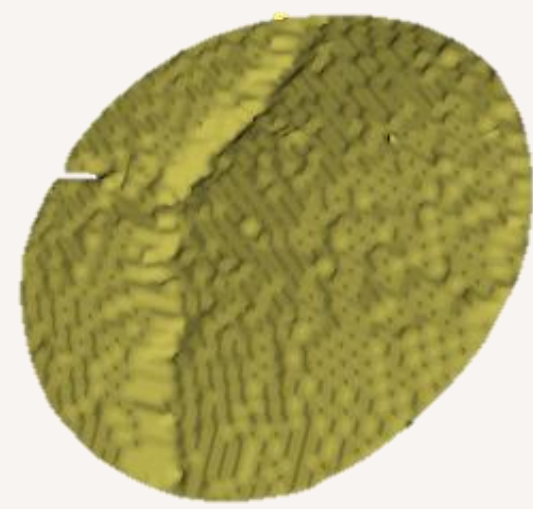
Notre objectif

Utilisation d'un réseau de neurones convolutionnel (CNN) pour l'apprentissage des zones anatomiques (ex. jonctions de lignes de suture) caractérisant l'emplacement des landmarks afin d'automatiser leur positionnement.

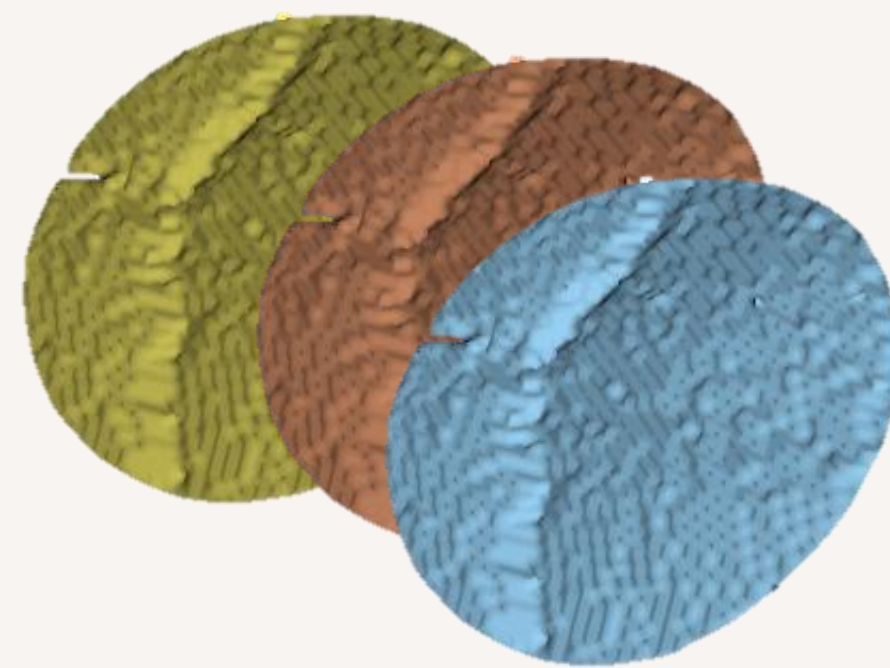
Focus sur des petites zones d'intérêt définies préalablement par le logiciel de prédiction ALPACA. On utilise le CNN pour améliorer la position de la prédiction ALPACA.



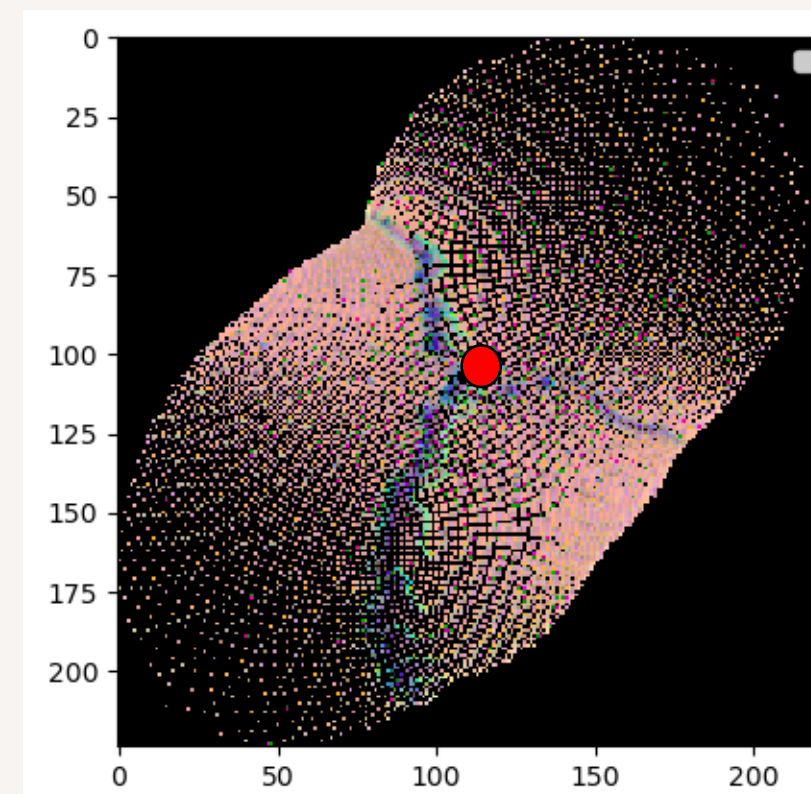
METHODE : PRODUCTION DES IMAGES D'ENTRAINEMENT



1. Extraction de la zone d'intérêt à analyser (3 rayons différents, 1 translation) => 4 petits meshes 3D par zone

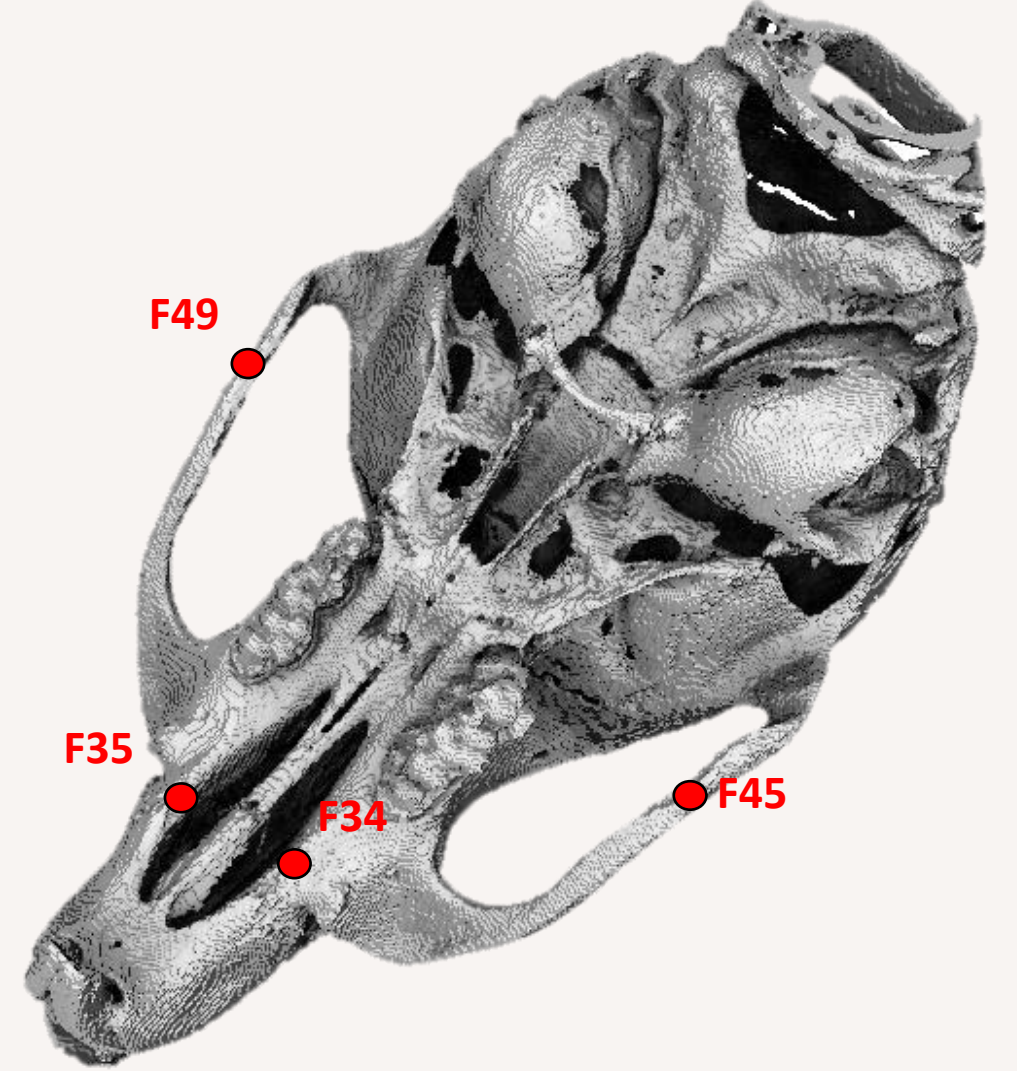
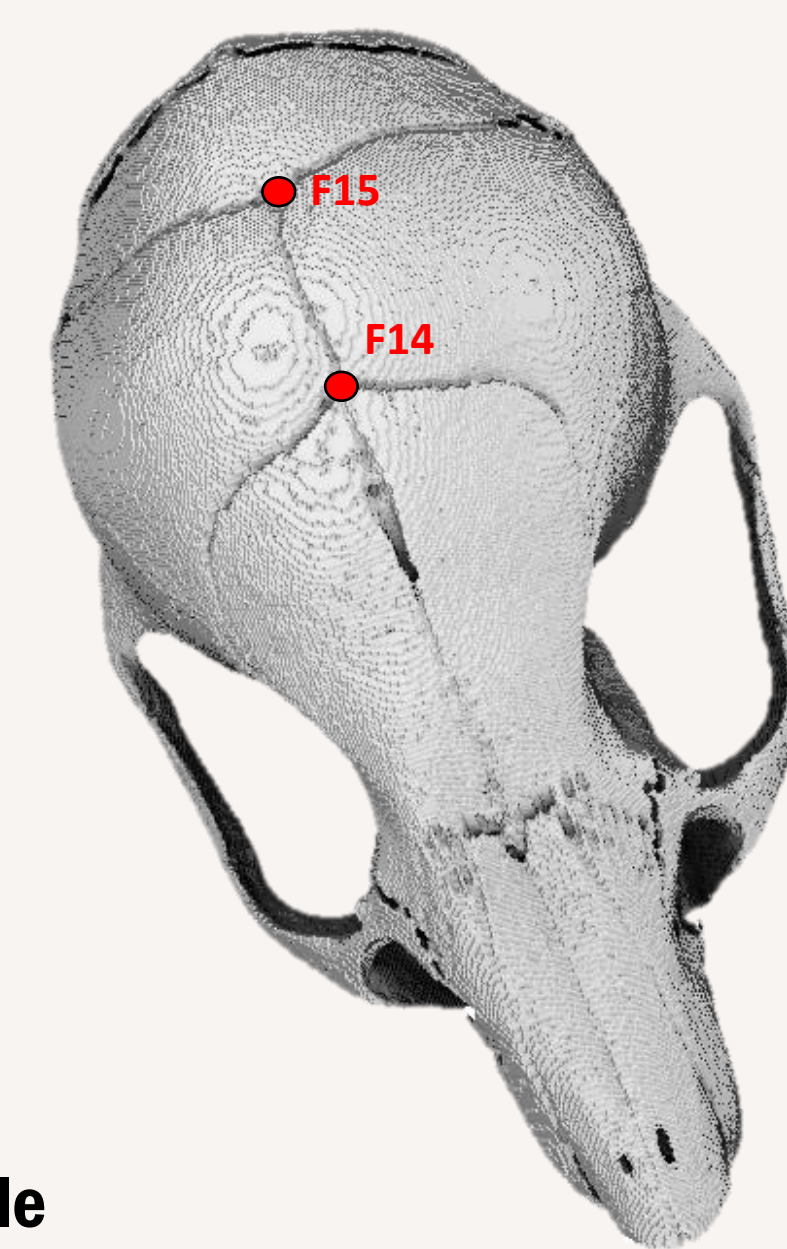


2. Coloration RGB (occlusion ambiante, obscurance volumétrique, courbure)



3. Paramétrisation dans un espace 2D et transformation en images raster RGB de taille 224x224

4. Repositionnement des landmarks « vrais » connus pour l'entraînement du modèle

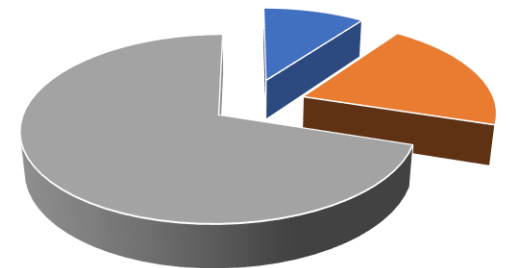
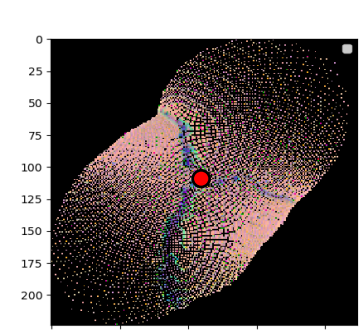


Entraînement des modèles pour 6 landmarks, avec entre 6800 et 9100 images disponibles par landmark pour l'entraînement

METHODE : DETAIL DES RESEAUX DE NEURONES

DECOUPAGE DU JEU D'IMAGES

70% training
20% validation ; 10% test



Input: image raster 224x224 et position (x,y) du landmark connu

Output: prédiction de la position (x,y) du landmark

ARCHITECTURES TESTEES

Xception
ResNet101
ViT

POIDS

Pré-entraînement « ImageNet »

METRIQUES D'EVALUATION

Loss : mean squared error (MSE)
Eval : Root MSE

TRAITEMENT DES IMAGES

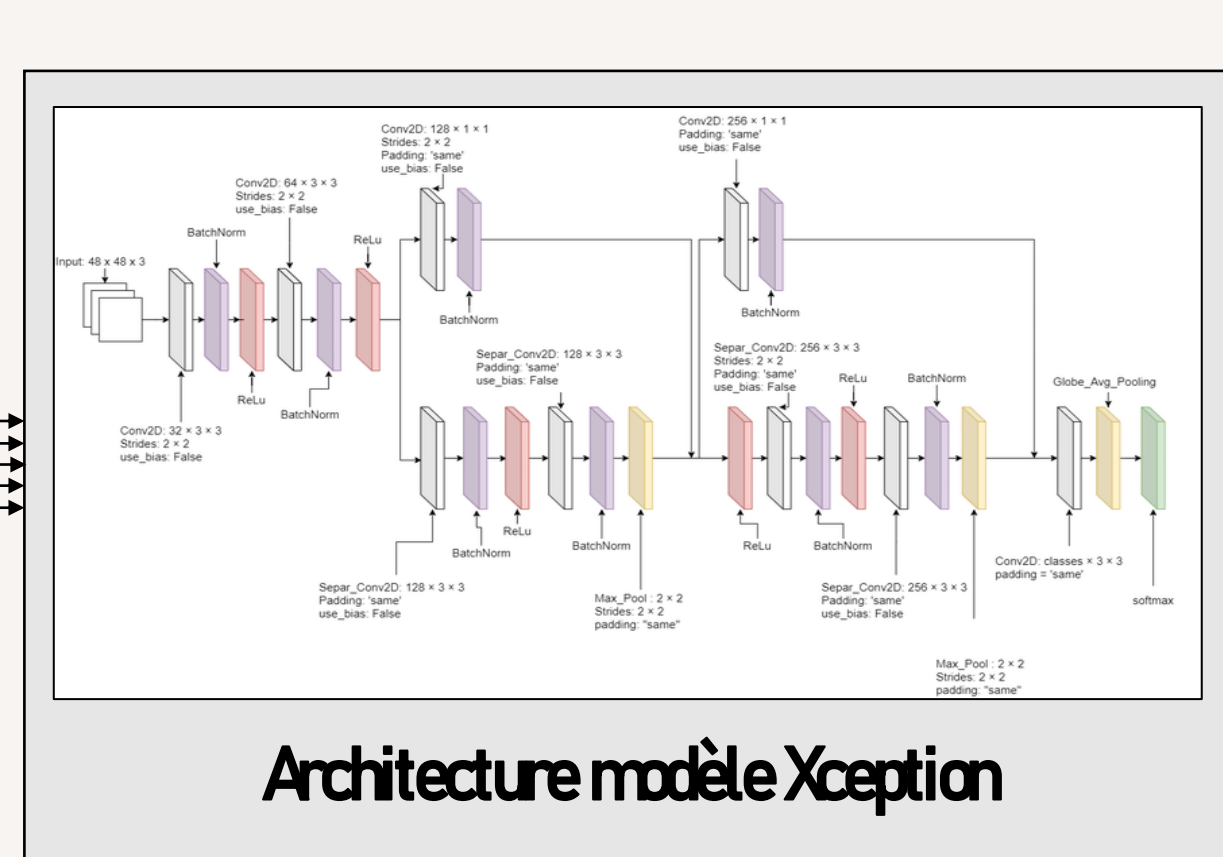
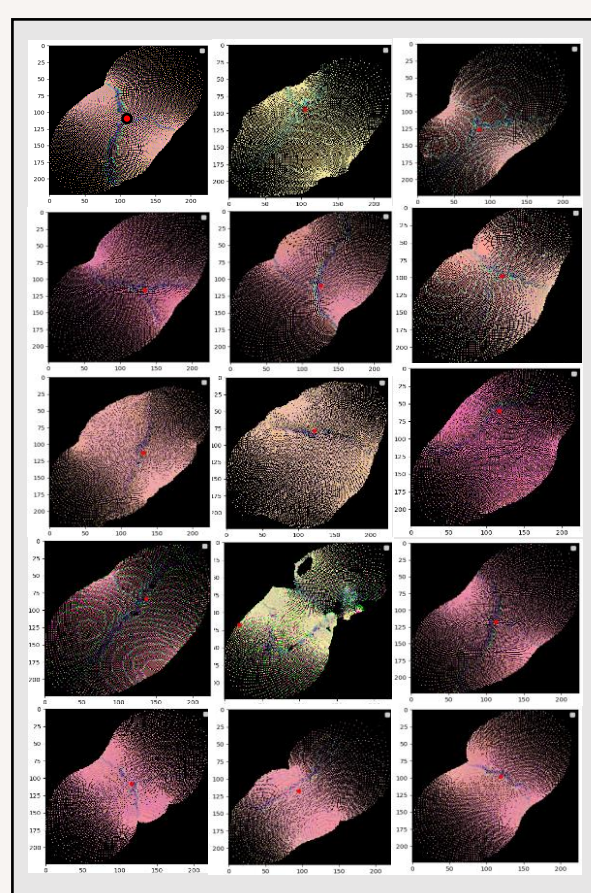
Normalisation des images [0,1]
Augmentation : rotation 30°

CALLBACKS & HYPERPARAMETRES

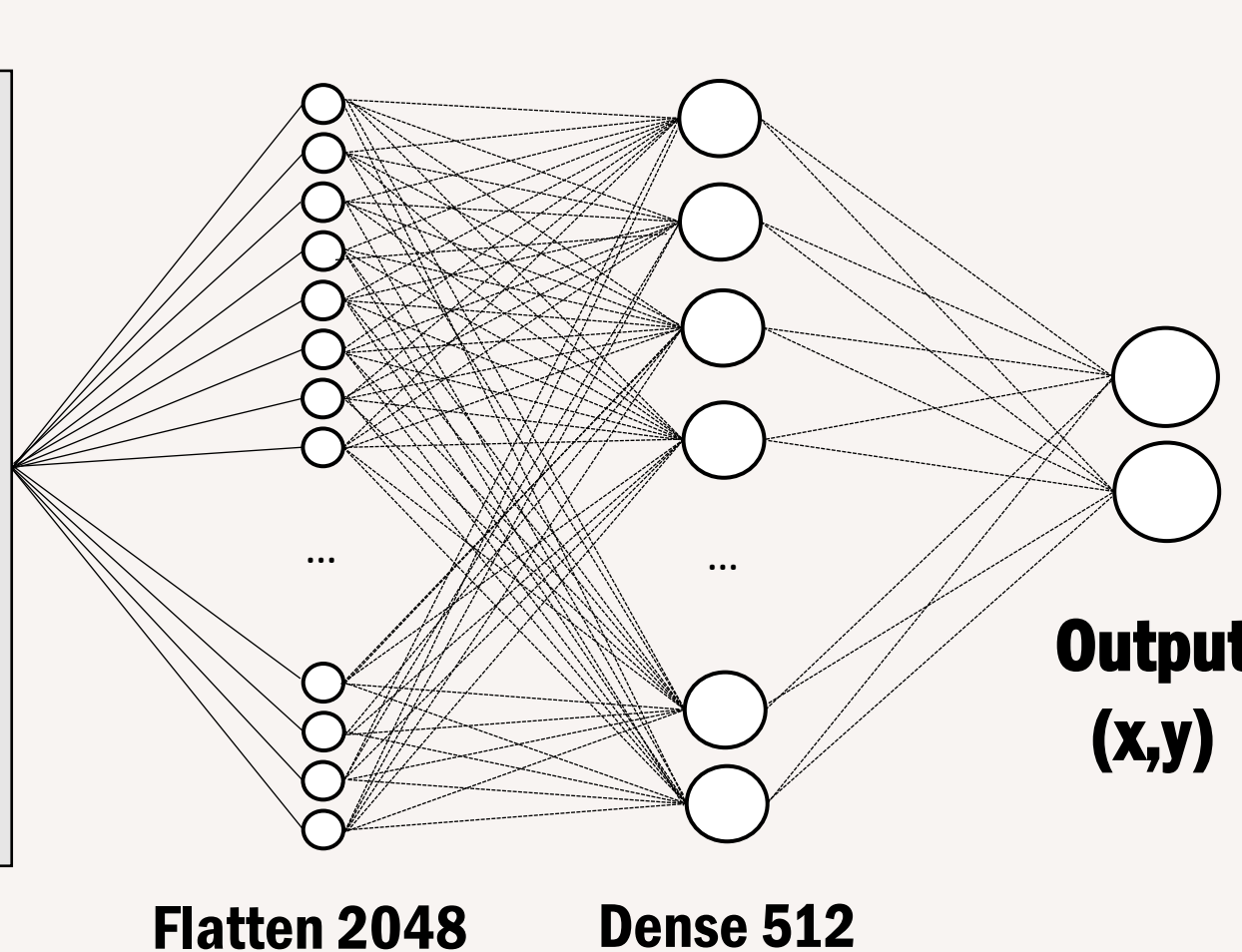
Early stopping, monitoré sur loss, patience de 15
Checkpoint pour enregistrer le meilleur modèle

Batch size de 128
Epochs: 500
Optimizer: Adam; learning rate : défaut 10⁻⁴
DropOut, batchNormalization et régularisation L1/12

ARCHITECTURE XCEPTION



Couches personnalisées



Input. Images raster 224x224 et position (x,y) du landmark connu

```
from tensorflow.keras.applications import Xception
base_model = Xception(input_shape=(224, 224, 3), include_top=False, weights='imagenet')
base_model.trainable = True # sortie du modèle chargé : block CNN taille 2048
```

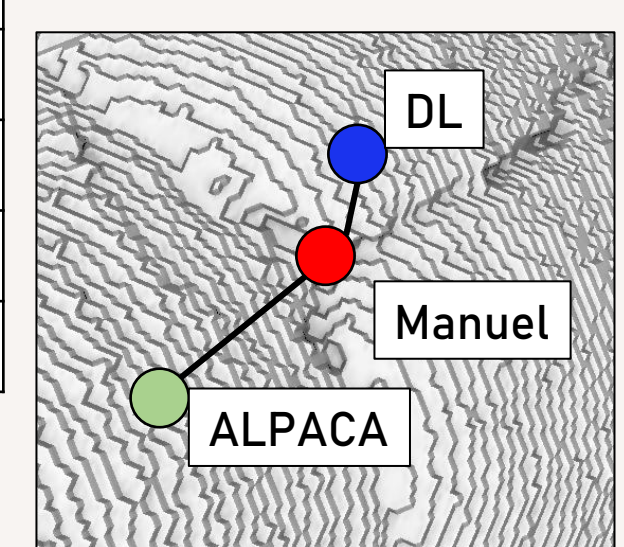
```
model = keras.Sequential([
    base_model,
    layers.MaxPooling2D((4,4)),
    Flatten(),
    layers.Dense(512, activation='relu', kernel_regularizer=l1_l2(l1=0.5, l2=0.5)),
    Dropout(0.5),
    layers.BatchNormalization(),
    layers.Dense(2, kernel_regularizer=l1_l2(l1=0.5, l2=0.5)) ])
```

RESULTATS

Landmark	Nombre d'images	Meilleur que ALPACA?*
F15	7426	94.74%
F14	9063	98.56%
F35	6826	75.81%
F34	7947	94.08%
F49	8539	88.04%
F45	9800	93.16%

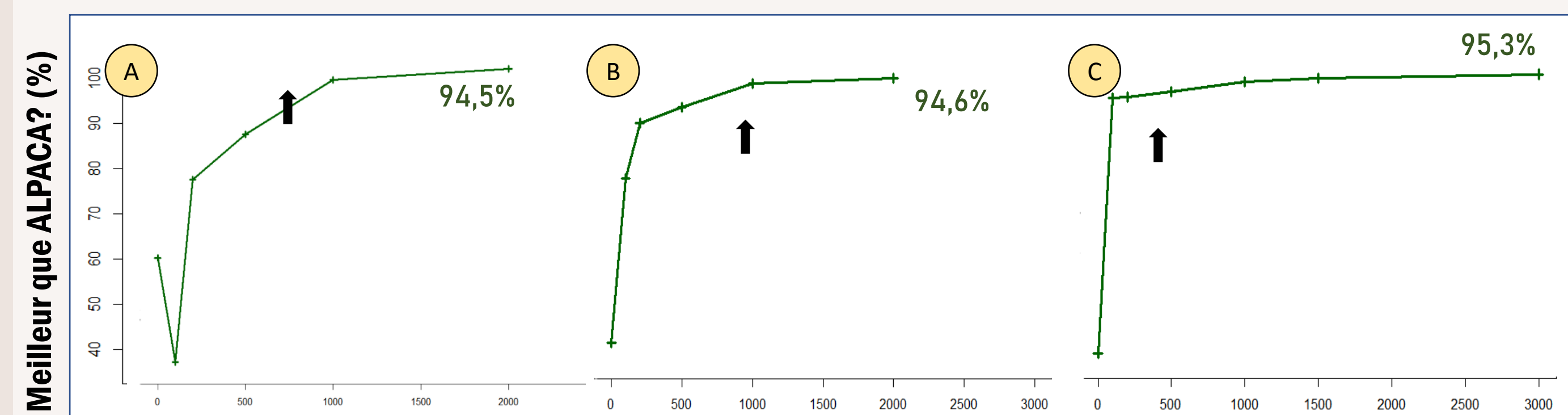
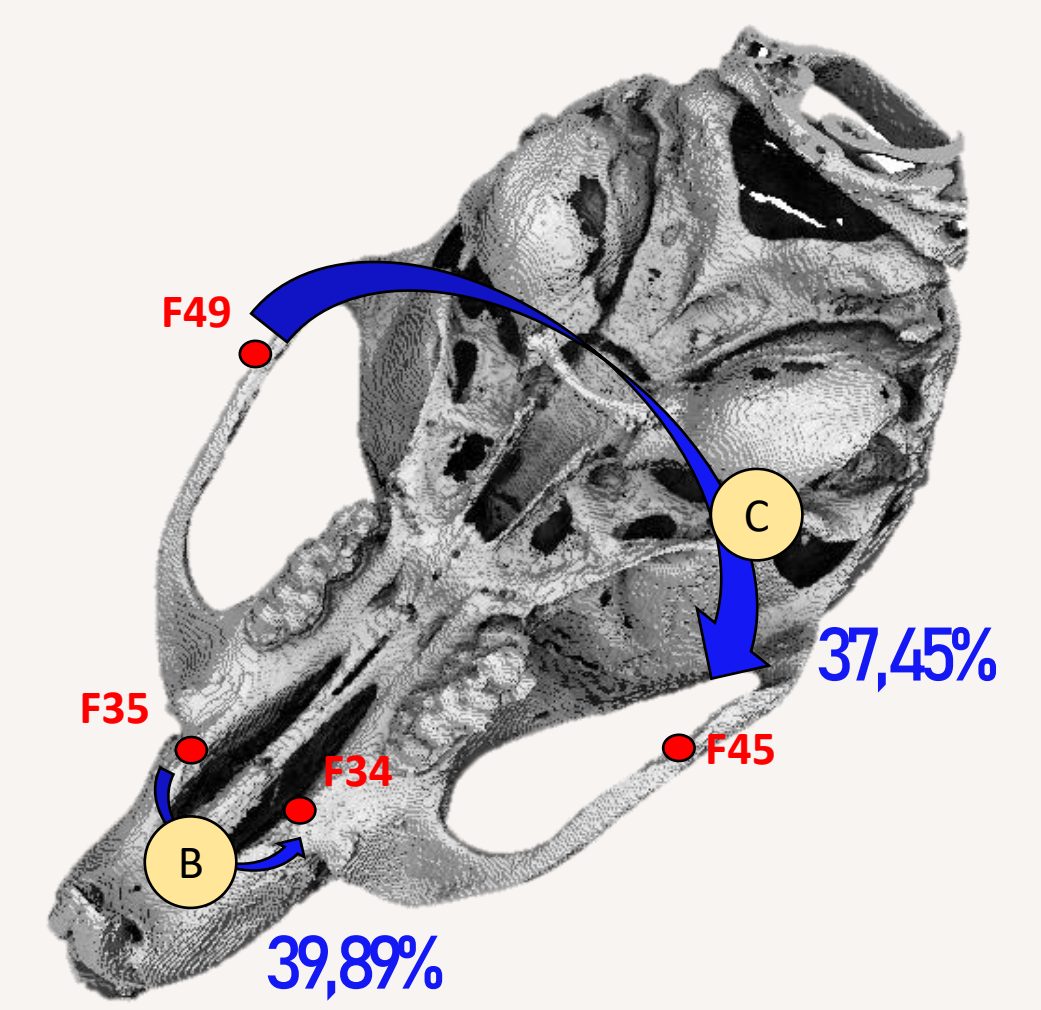
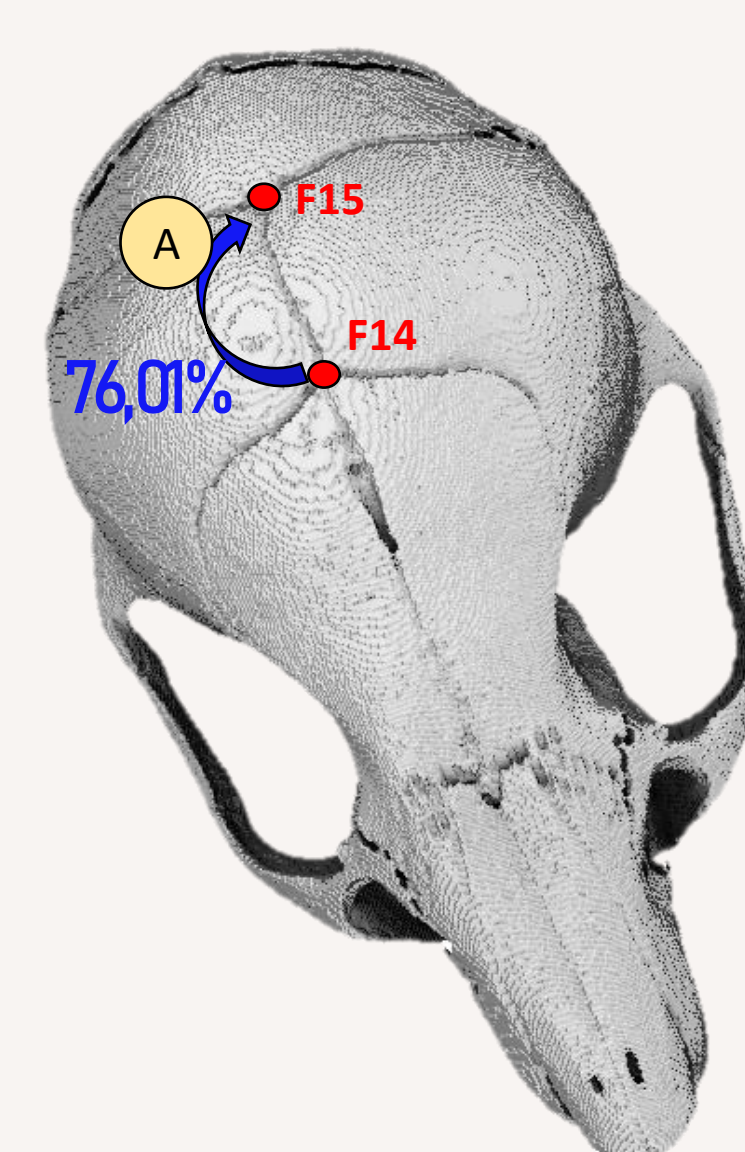
Tableau 1. Résultats des entraînements des modèles pour 6 landmarks (Xception). En cours pour ResNet101 et ViT.

* Comparaison du nb de fois où la distance au manuel est inférieure à celle de la prédiction ALPACA



TRANSFER LEARNING et FINE-TUNING

A Transfer learning du modèle de F14 sur F15 B De F35 sur F34 C De F49 sur F45



Fine-tuning : réentraînement des modèles en utilisant des images du landmark, entre 100 et 1000 images nécessaires pour atteindre des scores intéressants