

Rapport du projet lié à l'UE : Conception d'un Projet de  
Recherche et de Développement

# MASTER 1

Cursus : Bio- Informatique

---

## Outil de visualisation de Landmark chez les Scarabées

---

*Réalisé par :*

Maram AL HASSAN  
Cléo BERREHAIL  
Ayse ERGUN  
Hanane HASSAINIA

*Sous la direction de :*

Marie BEURTON AIMAR (labri)

*Client :*

Van Linh LE (labri)

23 mai 2020

## PREFACE

Dans le cadre de notre première année de Master en Bioinformatique, notre équipe, composée de Maram Al Hassan, Cléo Berrehail, Ayse Ergun et Hanane Hassainia, a eu l'occasion de travailler sur un réel projet durant quatre mois. Ce qui nous a permis de mettre en application toutes les notions et connaissances apprises au cours de cette année.

Ayant toutes les quatre un intérêt pour le développement Web, nous nous sommes naturellement dirigées vers ce projet afin de nous confronter aux enjeux de ce domaine et d'en avoir une compréhension plus approfondie. Nous nous sommes donc engagées à répondre aux attentes de notre client.

Ainsi, dans ce rapport seront détaillées toutes les étapes de l'analyse, de la conception ainsi que de la réalisation de notre projet.

Dans un premier temps, nous présenterons le contexte et motivation ainsi qu'un point sur l'état de l'art et la présentation des données nécessaires au fonctionnement de l'algorithme. Une partie sera également attribuée aux cahiers des charges présentant les objectifs. Dans un second temps, nous élaborerons un plan pour la conception du projet, notamment sur l'organisation du site, ainsi que le développement des fonctions javascript et l'interface graphique. Pour finir, nous détaillerons les étapes nécessaires à la réalisation du projet, la présentation de la structure finale du site et l'explication des différentes fonctions.

Ce projet étant conséquent, de nouvelles notions seront abordées pour répondre aux besoins de ce projet. Ceci sera, pour finir, détailler dans une dernière partie, dans laquelle seront abordés les apports et compétences acquises grâce au travail fourni à cette conception.

# Table des matières

<b>1</b>	<b>Analyse du Projet</b>	<b>1</b>
1.1	Contexte et motivation . . . . .	1
1.2	État de l’art . . . . .	2
1.3	Présentation des données . . . . .	2
1.3.1	Les données d’entrée . . . . .	2
1.3.2	Les fichiers .TPS . . . . .	3
1.3.3	Les fichiers CSV . . . . .	3
1.3.4	Les images de scarabées . . . . .	4
1.4	Besoin et Objectifs . . . . .	4
1.4.1	Besoins fonctionnels . . . . .	5
1.4.2	Besoins non fonctionnels . . . . .	6
1.5	Les données de sortie . . . . .	6
<b>2</b>	<b>Conception du Projet</b>	<b>7</b>
2.1	Organisation du logiciel . . . . .	7
2.2	Outils de développements . . . . .	8
2.2.1	Onglet Landmark . . . . .	8
2.2.2	Onglet Statistic . . . . .	9
2.3	Interface Graphique . . . . .	11
<b>3</b>	<b>Réalisation du Projet</b>	<b>14</b>
3.1	Importation des données . . . . .	14
3.2	Visualisation landmark - gestion des données . . . . .	15
3.3	Visualisation Statistique . . . . .	16
3.4	Les données de sortie . . . . .	19
3.4.1	Sortie de l’image traitée . . . . .	19
3.4.2	Sortie statistiques . . . . .	19
3.5	Construction page Web . . . . .	20
3.5.1	Entête . . . . .	20
3.5.2	Bas de page . . . . .	21
3.5.3	Corps de page de la page des landmarks . . . . .	21

3.5.4	Corps de page de l'onglet des statistiques . . . . .	23
3.5.5	Corps de page de l'onglet 'Home' . . . . .	25
3.5.6	Dimensions et proportions de la page web . . . . .	25
<b>4</b>	<b>Perspectives d'améliorations</b>	<b>26</b>
4.1	Importation de répertoire . . . . .	26
4.2	Création d'un compte personnel . . . . .	26
4.3	Rendre l'application totalement fonctionnelle sur mobile . . . . .	27
	<b>Conclusion</b>	<b>28</b>
	<b>Bibliographie</b>	<b>29</b>

# Chapitre 1

## Analyse du Projet

### 1.1 Contexte et motivation

En biologie l'analyse d'image est une étape indispensable lorsqu'on fait de la comparaison de phénotype entre individus. Un phénotype est l'ensemble des traits observables d'un organisme, comme la taille, la couleur et bien d'autre. C'est donc avec ces caractéristiques physiques qu'on est en mesure de caractériser les individus d'une même espèce. De nos jours, l'obtention d'une grande quantité de data n'est plus un problème. En effet, avec les outils de "photographies", l'obtention d'image de très bonne qualité pour un grand nombre d'individus est possible. En revanche, des complications apparaissent lors de l'analyse des images du fait de leur grand nombre. C'est là que rentre en jeu la bio-informatique. Le nombre conséquent de data conduit à la création d'algorithmes analysant automatiquement et facilement les fichiers choisis.

Il existe plusieurs méthodes d'analyses phénotypiques. Parmi ces méthodes, on retrouve l'étude des repères morphologiques (landmark en anglais). Les landmarks sont des points placés sur une image qui peuvent être utilisés pour identifier ou classer des individus.

Pour chaque individu, est fournit 5 images correspondant à 5 parties morphologiques : le pronotum, la tête, l'élytre, mandibule gauche et droite. Pour chacune des ces images nous avons 2 types de landmarks, le premier sont les landmarks manuelles fournit par le biologiste et le second sont les landmarks obtenus à partir d'un algorithme de machine learning (Probabilistic Hough Transform). Ainsi, une comparaison entre les deux types de landmarks sera possible.

Cette analyse permet l'observation du phénotype/génotype des scarabées dans le domaine de la recherche agricole. Elle permet donc de découvrir si il existe un lien entre le phénotype et l'écosystème des individus.

Afin de faciliter la partie des analyses des images, il nous a été proposé de créer un outil de visualisation de landmark sur des images de scarabées. L'outil de visualisation devra permettre l'affichage des deux types de repères (manuelle et prédit), la distance entre les deux ainsi que l'affichage de graphiques de statistique descriptive.

## 1.2 État de l'art

Depuis bien longtemps, les scientifiques cherches à améliorer le partage de données ainsi que de faciliter l'analyse de data. C'est ainsi que en 1989, Tim Berners-Lee créa un projet nommé le World Wide Web, permettant aux universités et instituts du monde de partager des données et informations plus facilement. À cette époque, l'information pouvait être échangée seulement via des fichiers de type texte. Ainsi, c'est seulement à partir de 1992 que l'on voit l'apparition d'image sur le WWW. Avec l'ouverture du Web au grand public, le partage d'images et de photographies est devenu indispensable au 21ème siècle. ...(still working)

## 1.3 Présentation des données

Pour répondre aux besoins de ce projet, certains éléments effectués en amont sont utiles même indispensables. Premièrement, les images à traiter doivent être fournies. Ces images correspondent aux différentes parties du corps des scarabées : la tête, les mandibules droites et gauches, l'élytre et le thorax. Ces photographies sont prises par le biologiste du laboratoire. Plus le nombre de photographies fournies est grand, plus les résultats seront concluant pour le client. En plus de ces données, les marquages manuelles et prédits seront également nécessaires. ces données nous sont tout comme les images, préalablement confiées par le biologiste.

### 1.3.1 Les données d'entrée

Les photographies, les fichiers contenant les landmarks ainsi que ceux contenant les données statistiques sont nécessairement fournis par le client. Ainsi pour chaque partie morphologique d'un scarabée nous avons un répertoire contenant environ 293 images. Dans chaque répertoire se trouve 3 sous répertoires : images, landmark manuel et landmark prédit.

### 1.3.2 Les fichiers .TPS

Les fichiers avec l'extension .tps sont le plus souvent associés aux applications de base de données Clarion TopScan et TopSpeed. Chaque ligne de ces fichiers correspondent à une entrée de données souvent séparée par des espaces ou des virgules. Les fichiers sur lesquels nous travaillons [Figure 1] contiennent :

- la partie du scarabée à laquelle il est associé
- les coordonnées en pixels de chaque repère
- le nom de l'image associée

```
LM=11
108 166
114 175
101 175
116 171
142 173
126 36
109 25
103 25
86 37
71 173
98 171
IMAGE=Elytra001.JPG
```

Figure 1 : Contenu du fichier TPS pour le repère manuel sur l'Elytra du premier individu

### 1.3.3 Les fichiers CSV

Les fichiers CSV (Comma-separated values) sont des fichiers contenant des tableaux dont les valeurs sont séparées par des virgules [Figure 2]. Les fichiers fournis contiennent chacun 6 colonnes : images, landmark, Xpred, Ypred, xmanuel, ymanuel et enfin une colonne nommée distance. Le calcul utilisé pour la distance est la distance euclidienne. Chaque partie morphologique est associée à un fichier csv.

#### elytra

Image	Landmark	Xpred	Ypred	xmanuel	ymanuel	distance
1	1	109.157184	29.134944	108	26	3.34169847427802
1	2	115.678368	17.297472	114	17	1.70452595586222
1	3	100.1694912	17.98848	101	17	1.29106064043383
1	4	116.534496	23.6088	116	21	2.66299144084543
1	5	141.837216	20.366304	142	19	1.37596702397695
1	6	132.187104	153.512256	126	156	6.66851753513118
1	7	114.426624	167.773824	109	167	5.48151909787351
1	8	108.21744	167.420832	103	167	5.23438436932405
1	9	90.0869952	155.47632	86	155	4.11465800610732

Figure 2 : Contenu du fichier CSV pour l'Elytra de tous les Scarabées

### 1.3.4 Les images de scarabées

Les images sont des photographies fournies par des biologistes afin d'implémenter et d'afficher les deux types de points de repères sur celui-ci **[Figure 3]**. Toutes les images ont la même taille (192x192). Ils sont enregistrés au format Joint Photographic Experts Group (.JPG), c'est le format le plus couramment utilisé pour les images. Il y a 5 images pour chaque scarabée.

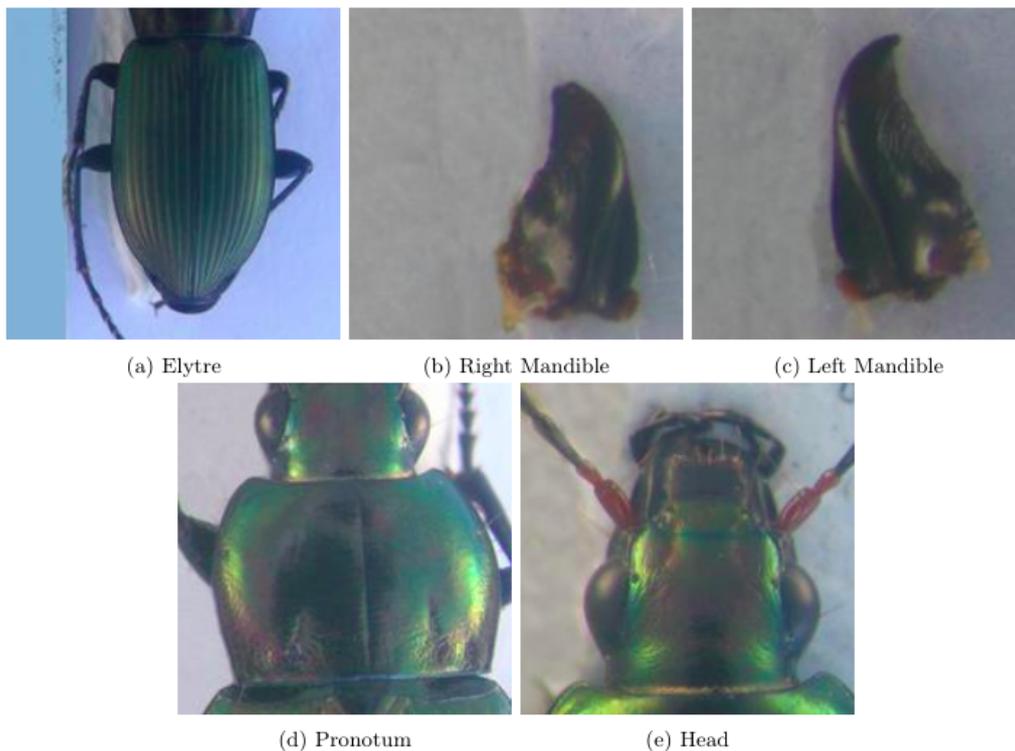


Figure 3 : Échantillon d'images des cinq parties étudiées du Scarabée

## 1.4 Besoin et Objectifs

L'objectif de ce projet est d'étudier les variations morphologiques chez les scarabées. A partir de fichiers tps fournis par l'utilisateur, l'algorithme devra permettre dans un premier temps, d'afficher les points sur l'image importée en fonction des coordonnées se trouvant dans les fichiers. Dans une deuxième temps, la distance entre ces points devra s'afficher, caractérisant la différence entre les landmarks prédit (par le machine learning) et les landmarks obtenus par les biologistes.

### 1.4.1 Besoins fonctionnels

L'objectif principal est de créer un site web capable d'importer des images et des fichiers (de type .tps) afin d'afficher les résultats nécessaires à partir de ces mêmes fichiers[Figure 4].

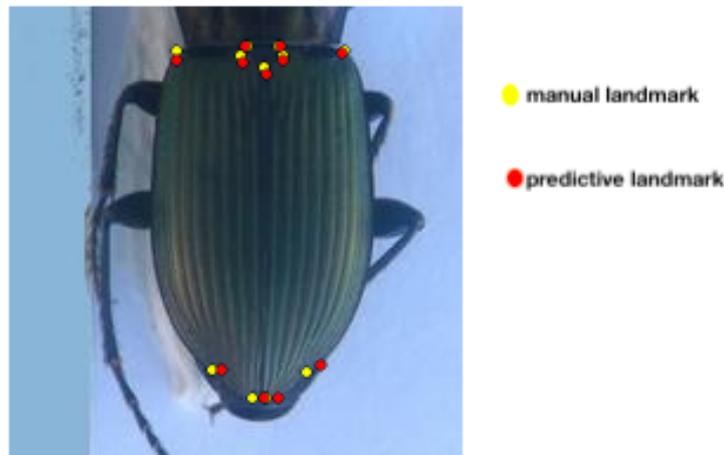


Figure 4 : Résultat attendu suite à l'exécution du programme

#### Traitement des données

Une fois importés, l'image ainsi que les fichiers tps seront affichés dans un menu latéral sur le site web. Le traitement des données permettra l'extraction des coordonnées enregistrés dans les fichiers TPS et l'affichage des points correspondant aux coordonnées (prédit et manuels) sur l'image importée.

#### Visualisation des données

Les résultats visibles sur le site web seront les photos de la partie sélectionnée du scarabée contenant les deux types de points de repères. L'utilisateur pourra choisir d'afficher soit les points prédits, soit manuels, et la distance entre les deux types de points. Un formulaire sera donc nécessaire afin que l'utilisateur indique le type de points qu'il veut placer. La sélection sera analysée par des fonctions JavaScript intégrées dans le HTML qui permettront l'affichage des points de repères sur l'image (prédit, manuels) ainsi que la visualisation de la différence entre les deux types de points de repères. Tous ces affichages se feront lorsque l'utilisateur cochera les cases adéquates. Une fois la case cochée, le script permettra l'affichage d'une ligne entre chaque point de repère manuel et prédit à chaque position sur l'image.

La statistique de la repartitions des points placés sur l'image ainsi que la différence entre les points seront misent en avant, permettant à l'utilisateur d'avoir une vue globale sur le placement de chaque point pour les mêmes parties du corps.

## 1.4.2 Besoins non fonctionnels

### JavaScript

L'objectif final de ce projet est de pouvoir visualiser et interpréter des données importées via un service web. Nous utiliserons alors plusieurs langages qui sont le JavaScript en lien avec le HTML et le CSS afin de répondre à la demande de l'utilisateur et de lui présenter les images comportant les landmarks. Nous utiliseront Vanilla JS qui est le framework de base le plus utilisé.

### Chart js

En bio-informatique, les statistiques sont indispensables. Les données contenues dans les fichiers CSV mis à notre disposition, permettrons l'implémentation de graphiques, facilitant leur visualisation. Pour se faire l'utilisation de Chart JS semble nécessaire. Chart JS est une librairie JavaScript gratuite qui facilite la construction de graphiques dynamiques. Ces derniers sont indispensable à certains sites web car comme dit précédemment, ils permettent d'avoir une meilleure interprétation statistique. Dans ce projet, les graphiques qui seront modélisés à partir des fichiers CSV, montreront la variation de la position de chaque type de points de repères sur chaque image des parties sélectionnées. En effet, on observera par exemple, la variation du point de repère manuel L1 dans chacune des images pour une partie morphologique du scarabée.

## 1.5 Les données de sortie

Il faudra également que le site web permette d'enregistrer les images et les graphiques sur l'ordinateur de l'utilisateur, afin qu'il puisse en garder une trace et les réutiliser.

# Chapitre 2

## Conception du Projet

### 2.1 Organisation du logiciel

L'organisation du site ([Figure 4]), se ferait en trois parties. La première partie que l'on retrouve est la page home. Elle a pour but de résumer le projet et d'apporter quelques informations supplémentaires. La seconde page, correspond à l'onglet 'Landmarks'. Elle contient le script principal. Ce dernier permet d'importer des images et des fichiers TPS, d'y afficher les données précédemment chargées pour pouvoir les télécharger (les images résultantes). Pour terminer, un troisième onglet dénommé 'Statistics', permet l'importation de fichiers CSV qui aboutirait à la modélisation de graphiques pouvant être visualisés et téléchargés.

L'utilisateur pourra ainsi facilement naviguer sur le site.

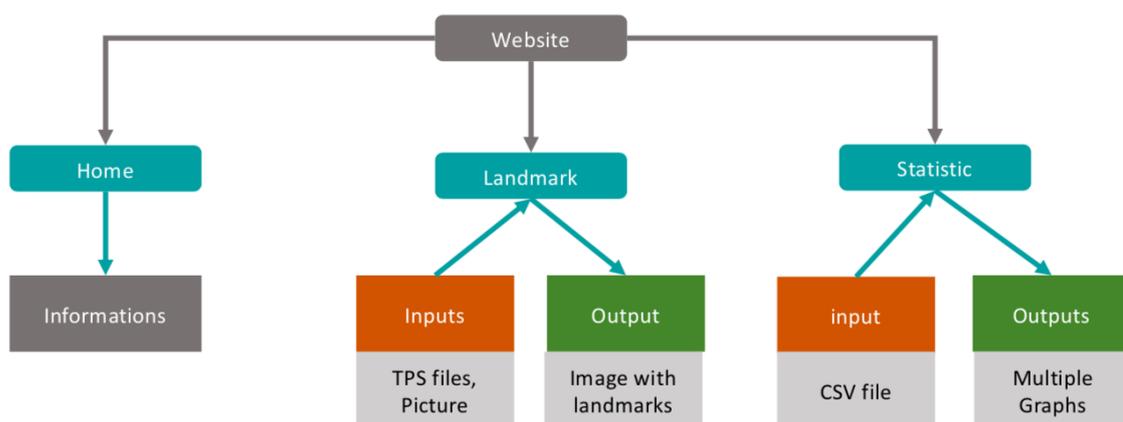


Figure 5 : Architecture logicielle du site internet

## 2.2 Outils de développements

### 2.2.1 Onglet Landmark

Le fonctionnement du script de cette page suivra le pipeline suivant [Figure 16]. Dans un premier temps l'utilisateur devra importer l'image et la glisser dans la zone où s'effectuent les opérations. Il devra ensuite importer les deux fichiers de type .tps correspondant à l'image. Il faudra par la suite que l'utilisateur clique sur un des trois boutons présent sur la page. Le bouton 'Manual', effectuera le placement des coordonnées des landmarks manuels sur l'image. Le bouton 'predicted' affichera les points des landmarks prédit. Enfin, le bouton 'distance' permettra d'afficher la distance (en pixel) entre les deux types de landmarks pour chacun des points. Le client pourra, à la fin, télécharger la nouvelle image contenant ces différentes données.

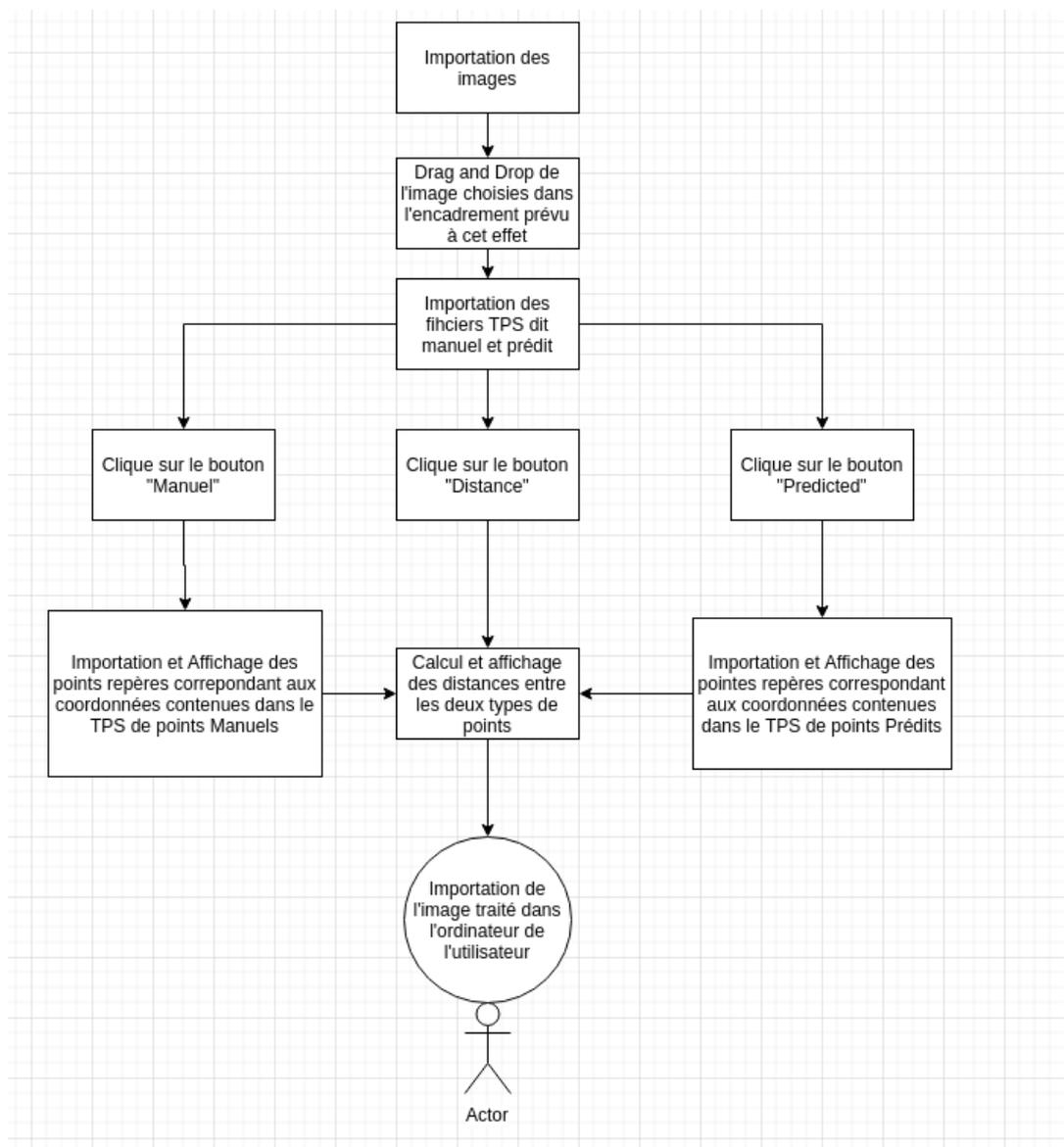


Figure 6 : Pipeline de la page Landmark

## 2.2.2 Onglet Statistic

Une partie importante de notre projet consiste en l'analyse statistique des données. Cette page permettra donc de visualiser les données fournies par les fichiers de type .csv sous forme de graphiques. Pour ce faire, nous utiliserons une librairie JavaScript nommée Chart.js. Chart.js permet la création de graphiques dynamiques et colorés. Huit types de graphiques différents y sont disponibles. On y retrouve par exemple les graphiques de type 'line' qui ont été choisis pour la conception de notre site. Sur la page, l'utilisateur devra importer un fichier de type csv. Ce dernier contient toutes les données concernant une partie morphologique spécifique du scarabée. Ce tableau contient sept en-têtes dans l'ordre suivant :

- Image
- Point de repère : la valeur varie d'une partie à l'autre, par exemple, l'élytre a onze points de repères sur chaque image mais les pronotums en ont huit.
- X-prediction : la coordonnée X en pixel pour le point de repère **prédit** correspondant.
- Y-prediction : la coordonnée Y en pixel pour le point de repère **prédit** correspondant.
- X-manuel : la coordonnée X en pixel pour le point de repère **manuel** correspondant.
- Y-manuel : la coordonnée Y en pixel pour le point de repère **manuel** correspondant.
- la distance : elle montre la différence (en pixel) entre les coordonnées des repères manuels marqués par les biologistes (grâce aux expériences in vitro ou in vivo) et ceux des repères prédits par le machine learning. Cette distance se nomme la distance euclidienne :

$$d = \sqrt{(x1 - x2)^2 + (y1 - y2)^2}$$

$x1 - x2$  est la valeur absolue de la différence entre les coordonnées x-manuel et x-prédiction.  $y1 - y2$  est la valeur absolue de la différence entre les coordonnées y-manuelles et y-prédiction.

Une fois importé, le script permettra de sélectionner les données nécessaires à la création des graphiques. Nous aurons 2 types de graphiques :

- les graphiques permettant d'observer, la distance en pixel entre les landmarks manuels et les landmarks prédits pour un landmark, pour toutes les images concernant une partie morphologique spécifique. Ainsi, si on prend l'exemple du fichier 'elytra.csv', le script affichera 11 graphiques correspondant aux 11 landmarks. Un exemple de graphe serait le suivant [Figure 7].

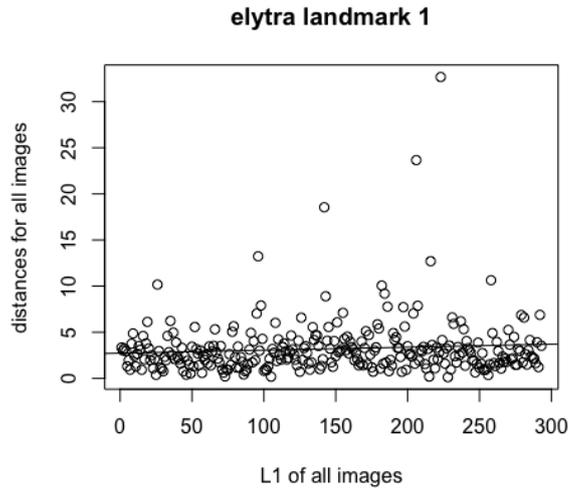


Figure 7 : Nuage de points de la distance entre les landmarks manuels et les landmarks prédit pour toutes les images des élytres pour le premier landmark ('L1').

- Un graphique qui représentera la moyenne des distances en pixel des deux types de point pour tous les landmarks. Dans le cas d'élytra le graphique contiendra 11 points représentant la moyenne de chaque landmark [Figure 8].

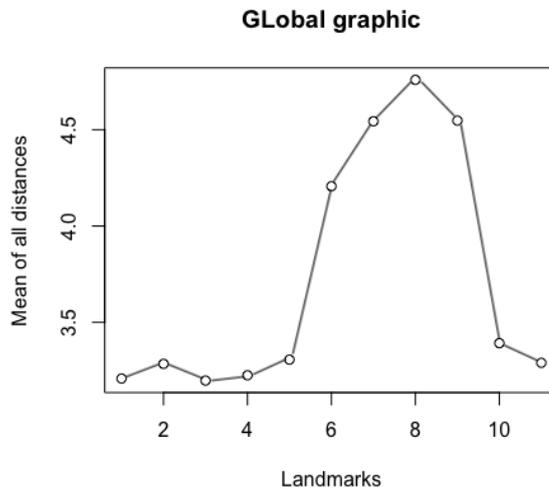


Figure 8 : Nuage de points de la moyenne des distances entre les landmarks manuels et les landmark prédit pour toutes les images des élytres pour le premier landmark ('L1').

L'utilisateur pourra également exporter les graphiques au format «.png» par le biais d'un bouton que nous aurions mis en place. Le téléchargement du fichier permettra ainsi à l'utilisateur de réutiliser les données qui ont été générés.

## 2.3 Interface Graphique

L'interface graphique nous utiliserons en majorité le langage CSS3 et le langage JavaScript (par la manipulation du DOM(Document Object Model)). En ce qui concerne l'esthétique, deux teintes de couleurs prédomineront : gris et bleu. Comme présenté précédemment [Figure 5] le site contiendra trois pages. Sur chacune de ces pages, nous retrouveront, le titre du site, le logo(un coléoptère) ainsi qu'un slogan.

La première est la page d'accueil [Figure 9] , la seconde est la page d'application [Figure 10] et la dernière est la page relative aux statistiques [Figure 11]. Dans la page d'accueil, nous afficherons des informations générales et images concernant les espèces de coléoptères afin d'enrichir notre site Web. Dans la deuxième fenêtre, l'utilisateur devra importer les images et les fichiers correspondants afin d'en faire l'analyse et d'y afficher le résultat escompté. Nous prévoyons d'avoir un menu coulissant dans le coin supérieur gauche de la deuxième fenêtre de notre site Web, qui sera composé de deux boutons. Ces deux boutons serviront à l'importation des images et des fichiers au format '.tps'. Au centre de cette deuxième onglet, il y aura un espace vide se situant près des trois boutons d'affichages (Landmarks manuels, Landmarks prédit, distances). Cet emplacement est la zone où l'image importée devra être déposée.

Enfin, le troisième onglet sera entièrement consacré aux statistiques. Deux boutons seront disponibles pour l'utilisateur lui permettant d'importer le fichier de type '.csv' et d'afficher les graphiques.

Afin d'être accessible à tous, le site sera hébergé sur le Github. Le Github est un service de gestion d'hébergement Web et de développement de logiciels. Vous pourrez accéder au site avec le lien suivant :

**[https ://cleobeldia.github.io/Projet \\_PDP/homePage.html](https://cleobeldia.github.io/Projet_PDP/homePage.html)**

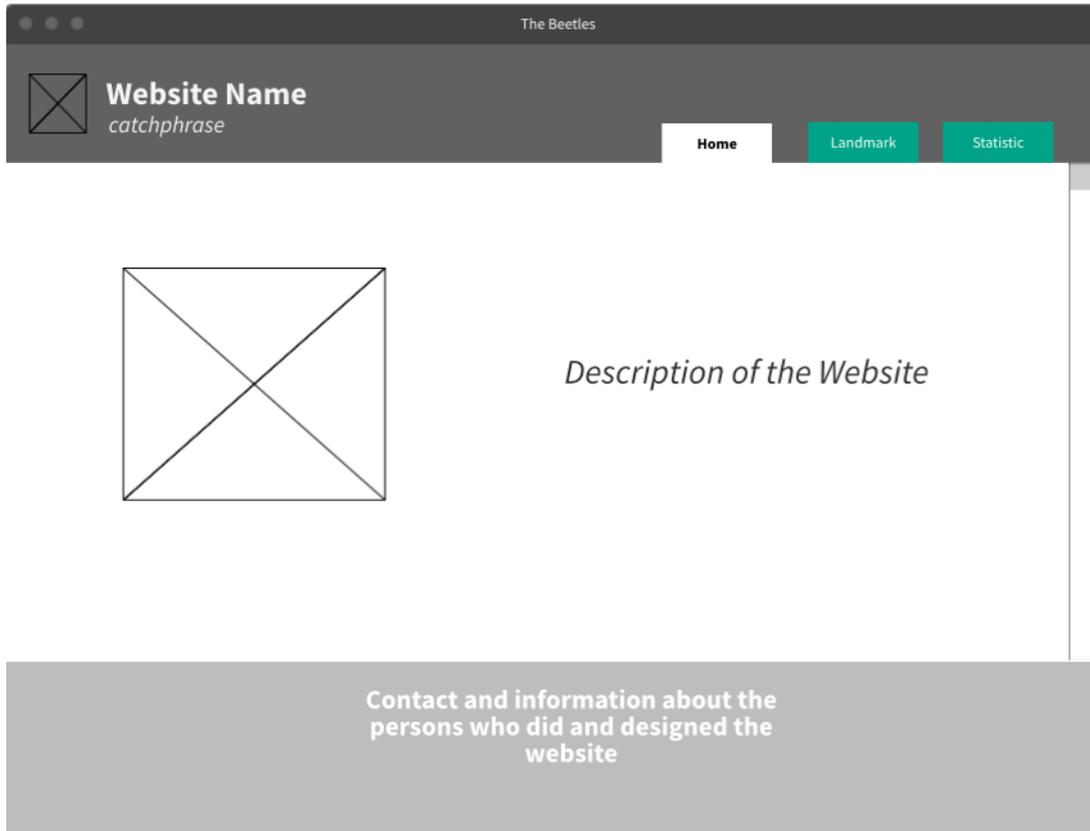


Figure 9 : Croquis de la l'onglet 'Home' de la page Web

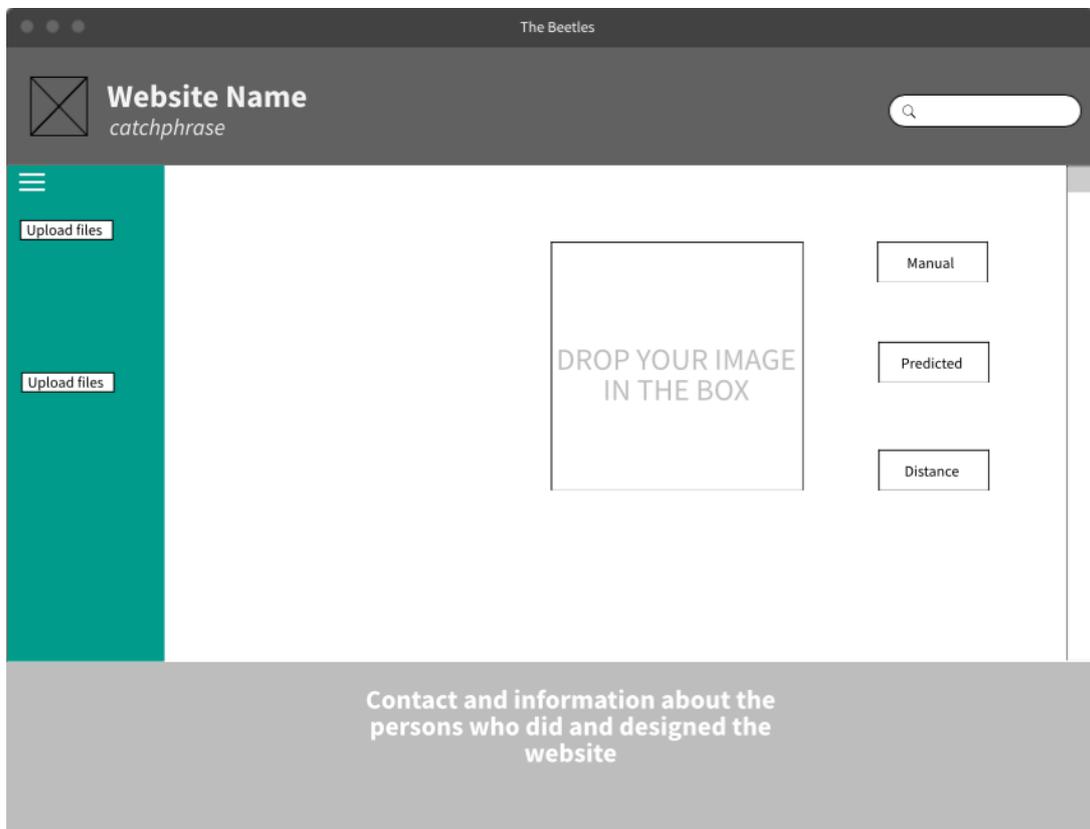


Figure 10 : Croquis de la l'onglet 'Landmarks' de la page Web

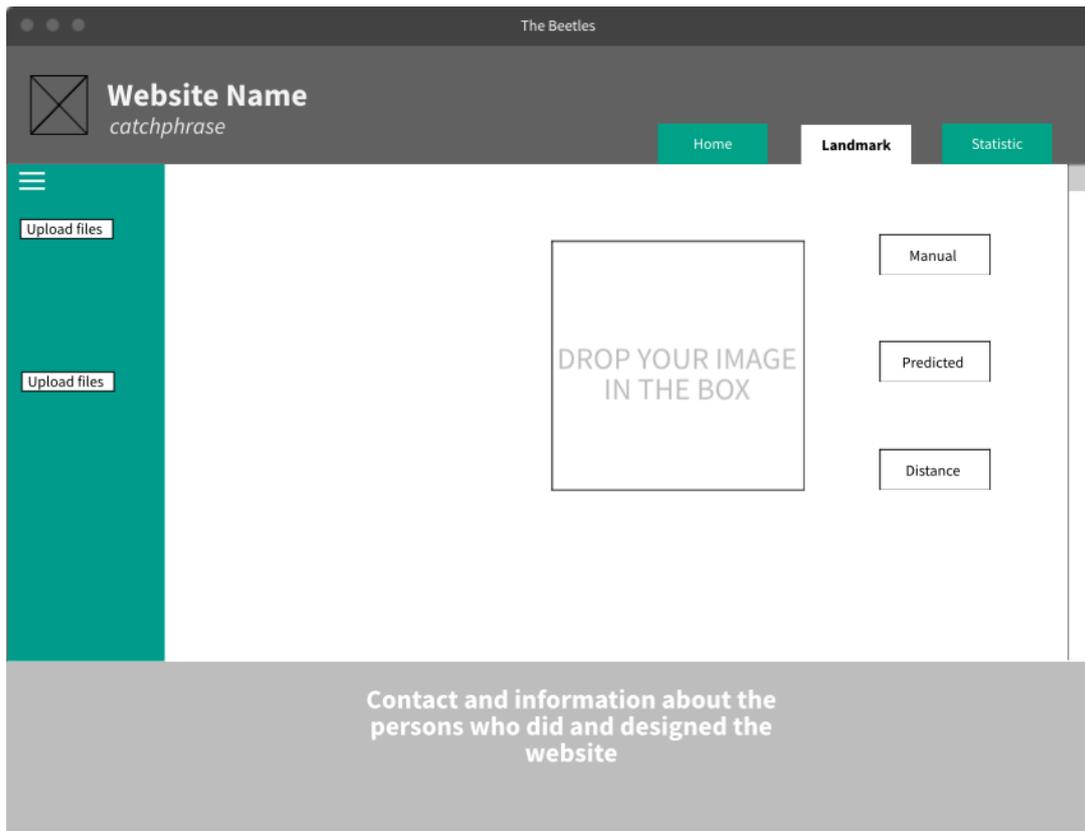


Figure 11 : Croquis de la l'onglet 'Statistics' de la page Web

# Chapitre 3

## Réalisation du Projet

Faire des schéma d'explication !

### 3.1 Importation des données

Afin d'assurer le bon fonctionnement du site internet, des images ainsi que des fichiers contenant les coordonnées des points repères à placer (fichier de type .TPS) sont nécessaires. En plus de ces derniers, des fichiers de type .csv, contenant toutes les informations pour chaque point de chaque image, sont également nécessaires afin d'avoir une approche statistique des données. Pour cela, il est donc utile de les importer dans la page Web afin d'extraire les informations et de les manipuler.

#### Landmarks

Afin de répondre aux besoins du traitement d'image, un code a été implémenté en JavaScript (cf script *landmarksPage.js*, permettant l'importation d'une image et des deux fichiers de type '.TPS'. Pour se faire, plusieurs fonctions ont été implémentées :

- La fonction *LoadFile* permet l'importation de l'image d'intérêt.
- La fonction *LoadTPS* permet l'importation des fichiers au format '.tps' correspondant à l'image d'intérêt. Cette fonction permet l'importation du contenu du fichier. La fonction *listpoint* permet alors d'insérer les coordonnées de chaque point dans deux listes distinctes correspondantes. Ainsi, une des listes contiendra la position en x des points à afficher et la seconde la position en y.
- Les fonctions *allowDrop*, *drap* et *drop* permettent à l'utilisateur de pouvoir prendre l'image en cliquant dessus et de la déposer jusqu'à la zone prévue.

## Statistiques

Afin de répondre aux besoins d'analyses de la distance enregistrée entre les landmarks manuels et prédit, un script JavaScript (cf **statistic.js**) a été implémenté dans le script html correspondant. Le code va permettre l'importation des données qui se trouvent dans les fichiers csv chacun relatif à une des cinq parties morphologiques des scarabées. Le fichier .csv téléchargé par l'utilisateur contient les valeurs qui vont être utilisées comme paramètres des fonctions responsable de la création des graphiques d'analyses :

- La fonction *Process()* : responsable du tri et stockage des données importées dans des listes.
- La fonction *renderChart(data,labels,name)* : responsable de la création des graphiques. Ces graphiques auront pour ordonné Le paramètre 'data' et 'labels' pour abscisse.
- la fonction *mean(list)* : Calcule la moyenne des distances pour chaque landmarks.

Ces fonctions seront minutieusement expliquées dans la partie visualisation statistique.

## 3.2 Visualisation landmark - gestion des données

Un des axes important de ce projet est l'observation du positionnement des points prédit et manuels de la distance les séparant. Un code a donc été implémenté en Javascript (cf script *landmarksPage.js*) permettant ainsi la visualisation des points repères de chaque catégorie. Une fois l'importation des fichiers (au format '.TPS') faite, et l'image placée dans le 'canvas' prévu pour le traitement, la visualisation des points peut se faire à l'aide de différentes fonctions :

- La fonction *draw* : prend en arguments des listes de coordonnées x et y, une taille ainsi qu'une couleur. Cette fonction permet de répartir les points en fonction de leurs coordonnées sur l'image placée dans la zone prévue.
- La fonction *distance* prend en arguments les deux listes correspondant contenant les coordonnées de tout les points manuels, ainsi que deux autres listes comportant les coordonnées des points prédit. Cette fonction permet la mise en évidence de la distance qu'il y a entre chaque type de points et leur affichage sur la page Web.
- La fonction *creerTab2(x,y,type)* : permet le remplissage des quatre tableaux présents sur la page. En effet, l'onglet 'Landmaks' contient quatre tableaux qui sont initialement vide et non visible. Lorsque l'utilisateur importe tous les fichiers nécessaires (les deux fichiers de types .tps), ces tableaux seront implémentés à l'aide de cette fonctions. Nous aurons alors, quatre tableaux contenant toutes les données toujours non visible. Le premier contient simplement le nom/numéro des points. Le second contient les coordonnées x et y des points prédit et le troisième les coordonnées x et y des points obtenus manuel-

lement. Enfin, le quatrième et dernier tableau, contient les distances entre les deux types de points.

- La fonction *afficherTab(type)* : prend en argument l'identifiant d'un des tableaux (type). Elle permet par la suite de rendre visible le tableau ayant pour identifiant l'argument de la fonction.

- La fonction *clearTab(type)* : prend en argument l'identifiant d'un des tableaux (type). Elle permet par la suite de changer le type d'affichage du tableau ayant pour identifiant l'argument de la fonction et de rendre ce dernier visible.

- La fonction *clearpoint* permet d'effacer les points dessinés au préalable sur l'image.

- La fonction *isChecked* permet de gérer les input de type checkBox sur la page Web afin d'afficher les points prédit, manuels, les deux en même temps, la distance ou même les trois paramètres ensembles. Tout cela est rendu possible à l'aide de boucles de conditions, qui vérifient quels boutons sont cochés et lesquels ne le sont pas. Ainsi en fonction de ces conditions, les données affichées seront différentes. Par exemple si seul le bouton 'predict' est coché seul les points prédit s'afficheront sur l'image.

### 3.3 Visualisation Statistique

Le deuxième axe important dans la mise en place de ce projet, est la modélisation et l'observation des données statistiques sous forme de graphiques dynamiques.

Pour ce faire, un script javascript dédié uniquement à l'analyse statistique est implémenté dans la section statistiques du fichier html. Il contient des fonctions essentielles et permet le bon fonctionnement de la page statistique **[Figure 12]**.

Parallèlement, il est impératif d'inclure chart.js dans la page. Cela se fait en y incluant le script suivant :

```
<script src="_COPY9@2.8.0"></script>
```

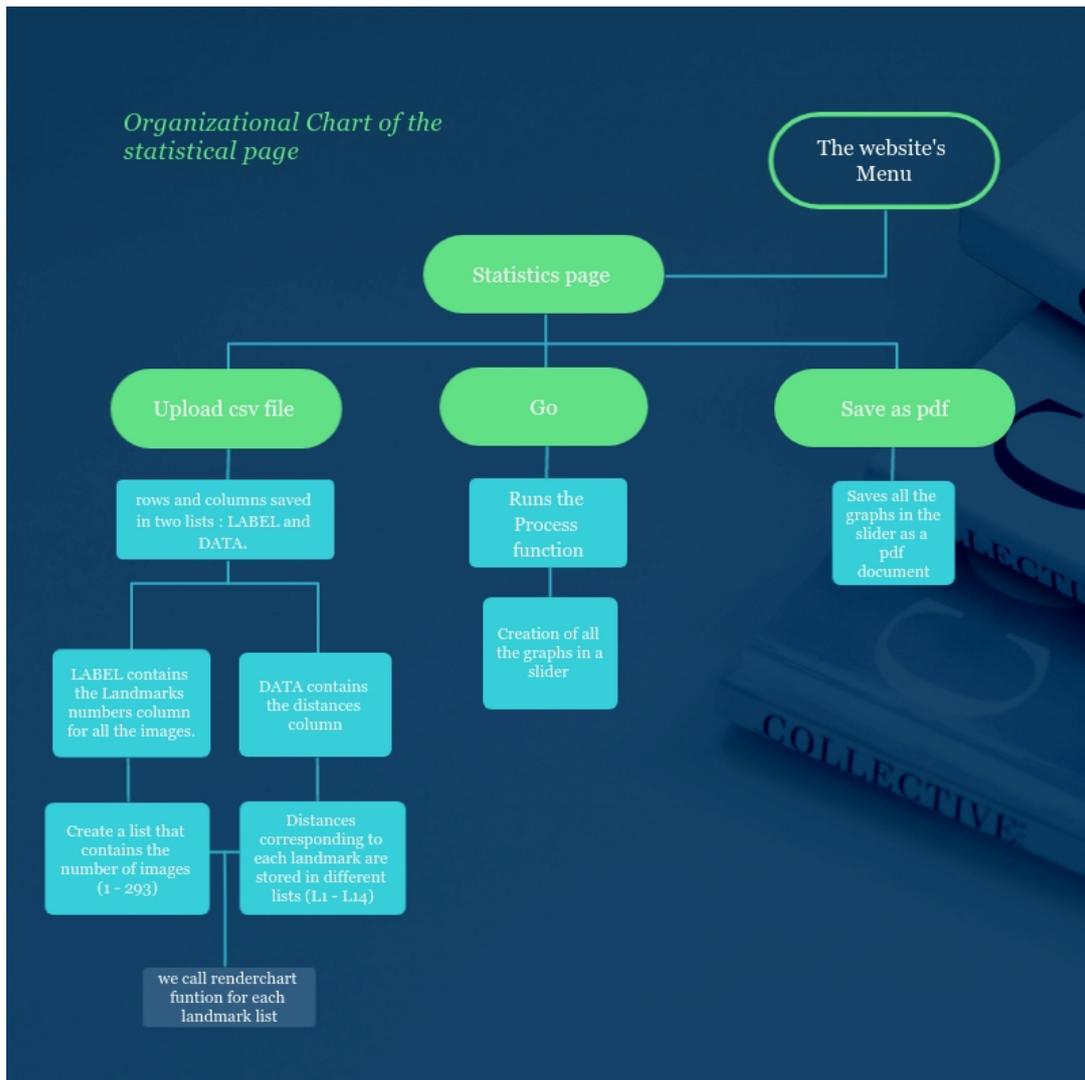


Figure 12 : Pipeline de la page statistique.

Le rôle premier de cet onglet est comme dit précédemment, l'affichage des graphiques. Il faut pour ce faire, que ces derniers soient, dans un premier, créés. Cela ce fait à partir des données importées par l'utilisateur. C'est le rôle que la fonction  $renderChart(data, labels, name)$  a. Elle prend en arguments trois variables, 'data', 'labels' et 'name'. 'data' fait référence à une liste contenant toutes les distances d'un landmark précis ('LM1' exemple) de toutes les images (pour une seul partie morphologique du scarabée). 'labels' correspond tout simplement au numéro d'identification de chaque landmark. Ainsi afin que tous les graphiques, correspondant aux distances de tous les landmarks ('LM1', 'LM2', 'LM3', etc...) de toutes les images, se fassent, la fonction  $Process()$  est indispensable.

Cette fonction va à avoir pour rôle de créer tous ces graphiques clés en faisant appel à la fonction  $renderChart$ . Elle va au préalable, implémenter 18 listes correspondant à tous les landmarks possibles (le nombre de repères varie en fonction de la partie morphologique étudiée, le nombre maximal de landmarks est 18). Ainsi seront stockés dans ces listes, tous les landmarks identiques de chaque image. Par exemple, tous les landmarks '1' de toutes les images (d'une seule partie morphologique) seront stockés dans la liste 'LM1'.

Tous les graphiques seront par la suite stockés dans une liste est affichés sur la page sous la forme d'un carrousel.

Ces graphiques permettent à l'utilisateur de visualiser l'évolution de la distance (en pixel) entre les landmarks prédit et les landmarks manuels selon les images[Figure 13].

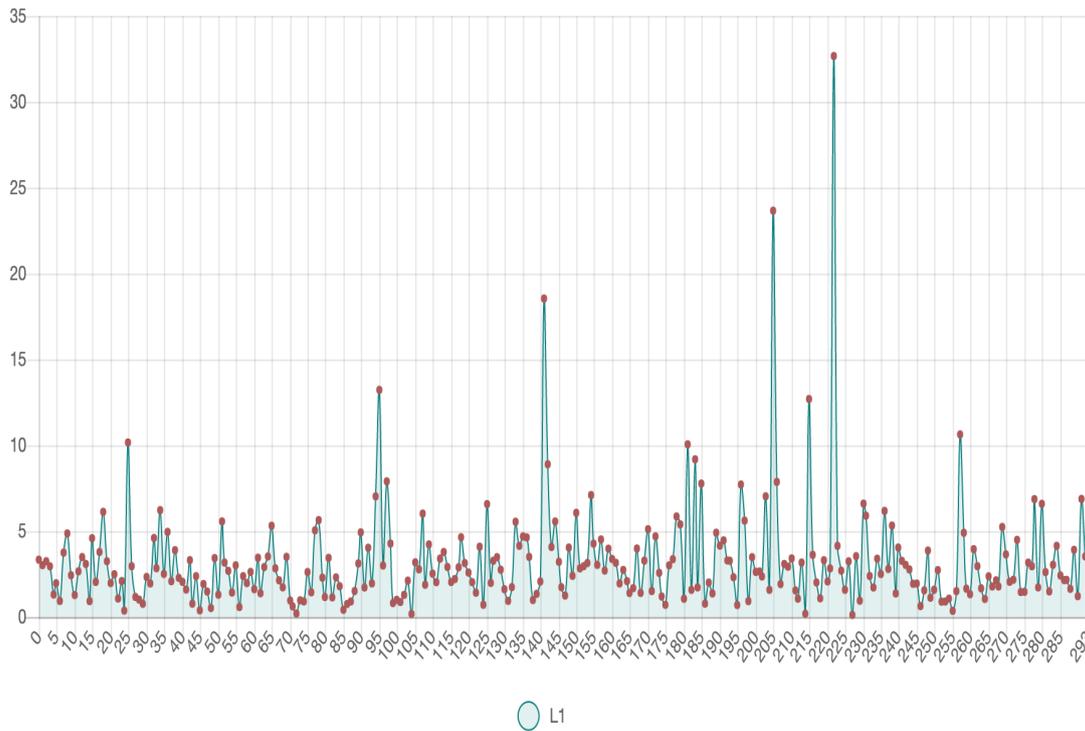


Figure 13 : Graphique 'L1' pour la partie morphologique du scarabée dites 'Elytre'

L'utilisateur pourra aussi visualiser Le graphique global qui vient s'ajouter à la fin du carrousel. Ce graphique permet la visualisation de l'évolution de la moyenne des distances de chaque landmark. Il est crée dans la fonction *process* et comme pour les autres graphiques il est crée à l'aide de la fonction *renderChart*. Avant sa création, la fonction *mean(list)* va calculer la moyenne des distances pour un landmark en faisant la moyenne des valeurs de la liste en paramètre.

Si nous prenons comme exemple le fichier 'Elytra.csv', le nombre de landmark pour cette partie est égal à 11. les 11 premières listes (allant de 'LM1' à 'LM11') seront alors implémentées des distances correspondantes. Les moyennes des valeurs de chacune de ces listes se feront alors, à l'aide de la fonction *mean*, permettant la création d'un graphique global de l'évolution de la moyenne des distances de tous les landmarks (de 1 à 11) pour la partie morphologique du scarabée appelée 'Elytre'.

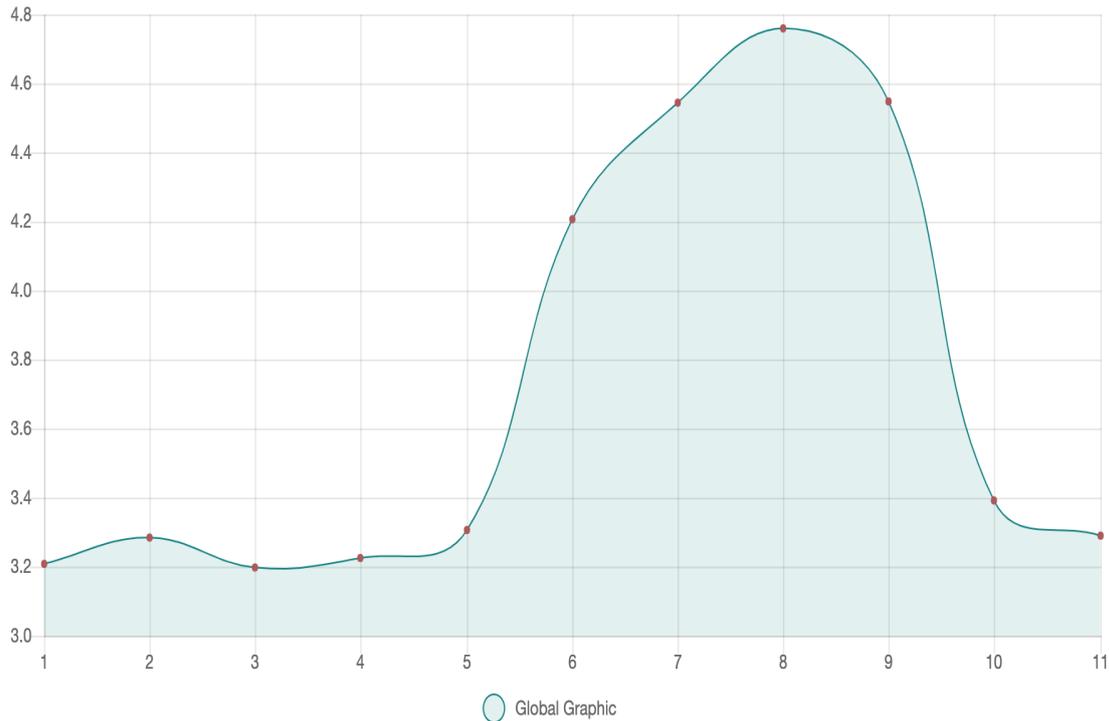


Figure 14 : Graphique 'Global' pour la partie morphologique du scarabée dites 'Elytre'

## 3.4 Les données de sortie

### 3.4.1 Sortie de l'image traitée

Afin de garder une trace de l'affichage des points ainsi que la distance qui les sépare, une fonction nommée *download\_beetle* qui permet l'enregistrement de l'image traitée dans l'ordinateur de l'utilisateur, est indispensable. Elle va créer une variable enregistrer qui utilise la méthode `toDataURL()` attribué au canvas dans lequel se trouve l'image qu'on veut enregistrer. Cette méthode retourne un data contenant la représentation de l'image dans le format indiqué (jpeg). Dans un second temps, on attribue l'hyperlien 'href' de l'image prise comme paramètre de la fonction.

### 3.4.2 Sortie statistiques

Afin de donner la possibilité à l'utilisateur de sauvegarder les analyses statistiques des données (les graphiques), une fonction nommée *download\_graph* a été implémentée. Cette fonction permet à l'utilisateur de sauvegarder chacun des graphiques, affichés à l'écran, au format jpeg.

Comme pour la sauvegarde de l'image, la fonction utilise la méthode `toDataURL` qui retourne un data contenant la représentation de l'image dans le format précisés. Cette méthode est attribuée à la variable `id_graph`. Cette dernière est définie dans la fonc-

tion *ChangeIndex* responsable de l’affichage successive des graphiques dans le carrousel. Rappelons que pour afficher les graphiques dans le carrousel, on a crée une liste nommée `canvas_array` qui stock tous les canvas utilisés pour dessiner chacun des graphes. Cette variable (`id_graph`) va, de ce fait, stocker l’identifiant du graphique affiché à l’écran.

## 3.5 Construction page Web

La construction de la page Web, s’est faites à l’aide des langages en CSS, HTML et JavaScript. Le site web est divisé en trois parties : Home, Landmarks, Statistics. Ces trois éléments sont donc les onglets visibles de la page internet.

Ils ont tous les trois la même organisation générale. Autrement dit, on retrouve dans chaque onglets, une entête, le corps de page ainsi qu’un bas de page. L’entête et le bas de page ne sont pas relatifs à chaque onglet, ils seront donc identiques pour les trois pages, contrairement au corps de page.

Quatre couleurs prédominantes ont été choisies pour le site, le rouge (`rgb(175, 88, 88)`), le blanc(`rgb(255,255,255)`) le vert/bleu(`rgb(0,128,128)`) et le gris(`rgb(128,128,128)`).

### 3.5.1 Entête

L’entête, correspond à la partie se situant en haut de la page. Elle est composée de deux parties distinct, la partie dites ‘Titre’ et le ‘menu’ **[Figure 15]**.

La partie Titre est comme son nom l’indique, la section où l’on retrouve le titre, une petite phrase d’accroche ainsi qu’un logo représentatif du site. Le nom du site internet est : ‘The Beetles’ et le logo représentatif de la page est un scarabée blanc au format ‘.png’.

Le menu, se situant en dessous et opposé à l’entête, comprends trois ‘boutons’. Ces trois boutons, ‘Home’, ‘Landmarks’ et ‘Statistics’ sont les trois liens direct vers chacune de nos trois pages internet. Ce menu, est constamment affiché à l’écran (Lorsque l’utilisateur descend la page, le menu suit le défilement).

La couleur de fond de cette partie est grise. L’écriture est de couleur blanche et la couleur de fond des trois boutons est de couleur `rgb(0,128,128)` (sauf pour le bouton correspondant à la page sur laquelle l’utilisateur se trouve, lui permettant plus facilement de se repérer sur le site internet). Ces derniers changent de couleur lorsque le curseur de la souris se situe dessus.



Figure 15 : Entête

### 3.5.2 Bas de page

Le bas de page, correspond à la partie se situant en bas de la page. Cette partie est partagée en 3 sections informatives. On y trouve premièrement, une section 'Who are we?' rappelant le nom des personnes ayant participées à la mise en place du site internet. La seconde section 'Contact', liste les adresses mails étudiantes des quatre participantes. La troisième section 'Find Us' indique simplement l'adresse postale de l'université de Bordeaux (campus de Talence). [Figure 16].

Cette section a une couleur de fond rgb(175, 88, 88) et l'écriture est de couleur blanche.



Figure 16 : Bas de page

### 3.5.3 Corps de page de la page des landmarks

La section correspondant au corps de page de la page web est, comme dit précédemment, relative à chaque onglet. En ce qui concerne l'onglet 'landmarks', Elle est partagée en deux parties dites fonctionnelles [Figure 17].

#### section d'importation

La première section, la section d'importation, est située sur la gauche [Figure 17]. Cette section contient les trois boutons permettant le choix et l'importation des trois fichiers nécessaires (les deux fichiers au format '.TPS' ainsi que l'image correspondante) au bon fonctionnement de l'application. Sous ces trois boutons, on retrouve un espace qui contiendra l'image choisie et importée par l'utilisateur. Enfin, le dernier élément de cette section est le bouton 'reset', permettant le rafraîchissement de la page et donc l'utilisation d'une nouvelle image.

Cette section a une couleur de fond bleu/verte et les trois boutons d'importation sont de couleur grise. l'espace qui contient l'image importée est délimité par un encadré gris. Le bouton reset a quant à lui un fond de couleur blanche et une écriture rouge. Les quatre 'boutons' de la section voient leur couleur changer lorsque le curseur de la souris se trouve au dessus, offrant à l'utilisateur une utilisation simplifiée et claire de l'application. De plus, lorsque l'utilisateur a importé un fichier au format '.TPS', le nom de ce dernier remplace les champs initialement présents sur le bouton.

## section d'affichage

La seconde section, est le 'coeur' de l'onglet Landmark. C'est dans cette section que vont s'afficher les données sortantes. On y retrouve tout d'abords quatre 'boutons' (trois input de type checkbox et un bouton). Trois de ces boutons (input de type checkbox) vont permettre l'affichage des points ainsi que la distance séparant les deux types de points. CeT affichage se fait sur l'image préalablement importée et déposée dans l'emplacement dédié. Cet emplacement, se situe juste en dessous de ces trois boutons. Enfin, encore plus bas, se trouve un bouton permettant le téléchargement de cette image sortante.

Pour terminer, afin que l'utilisateur utilise l'application de la bonne manière, un 'protocole' est présent en haut de la section. Cet espace correspond tout simplement à l'affichage de toutes les étapes permettant l'obtention d'une image comprenant les données voulues (landmarks prédit, manuel et les distances). Ainsi, à chaque fois que l'utilisateur aura validé une étape, cet espace sera actualisé afin d'afficher les instructions pour l'étape qui suit. Ce 'protocole' comme pour le menu, est toujours présent sur la page. Ainsi, même si l'utilisateur fait défiler la page plus bas, le protocole sera présent. D'autres éléments importants se trouvent dans cette section. Il s'agit du tableau des distances et des tableaux des coordonnées des deux types de landmarks (prédit et manuels). Ces tableaux ne sont pas directement visible sur la page. Il s'affiche si et seulement si, toutes les étapes ont été respectées. En effet, afin que ces tableaux s'affichent il faut dans un premier temps, obligatoirement, que l'utilisateur ai importé les deux fichiers au format '.TPS' en plus de l'image correspondante. Dans un second temps, lorsque l'utilisateur cochera un des boutons, un des tableaux correspondant s'affichera, juste en dessous du bouton de téléchargement. Ainsi, lorsque l'utilisateur cochera le bouton 'manual', le tableau contenant les coordonnées des points manuels s'affichera, en plus de l'affichage des landmarks sur l'image. Si les deux boutons 'manual' et 'predict' sont cochés alors les deux tableaux correspondant s'afficheront. L'affichage du tableau des distances nécessite une condition supplémentaire. Il ne s'affichera que si tous les boutons ('prédit', 'manual' et 'distance') sont cochés. Ainsi tous les tableaux s'afficheront **[Figure 18]**.

Cette section a une couleur de fond blanche. Les trois boutons d'affichages sont rouges, ils deviennent rouge foncé lorsque le curseur les survole. Lorsqu'ils sont cochés, il deviennent bleu. Le bouton de téléchargement est quant à lui bleu/vert et devient rouge au passage du curseur. l'emplacement situé juste au dessus de ce bouton, dédié au dépôt de l'image, est délimité par un encadré gris en pointillés. Les tableaux s'affichent en bleu/vert et blanc. Enfin, le protocole s'affiche dans un encadré bleu/vert et se situe dans un encadrement rouge en pointillés.

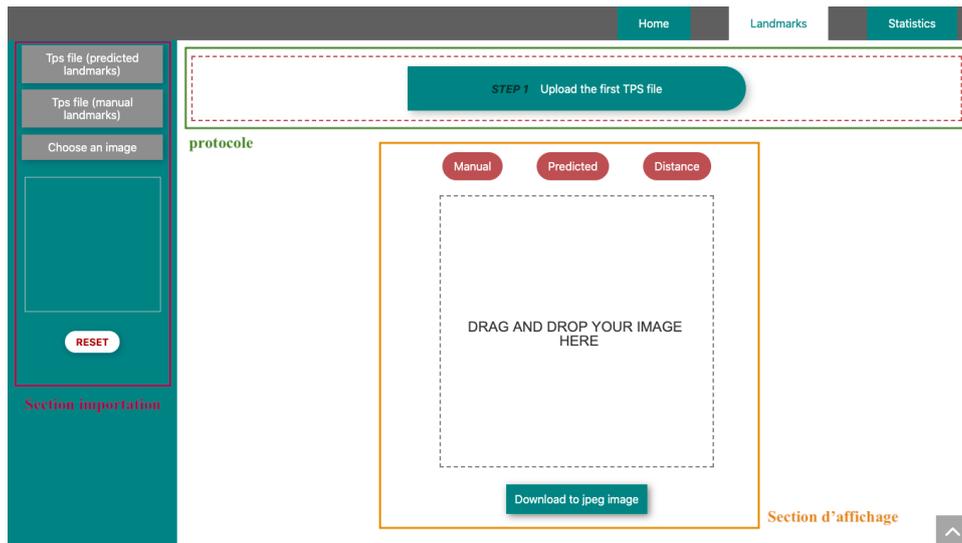


Figure 17 : Aperçu de l'onglet 'landmarks' légendés

Landmarks	Predicted		Manual		Distance
	x	y	x	y	
1	109	162	108	166	4.12
2	115	174	114	175	1.41
3	100	174	101	175	1.41
4	116	168	116	171	3
5	141	171	142	173	2.24
6	132	38	126	36	6.32
7	114	24	109	25	5.1
8	108	24	103	25	5.1
9	90	36	86	37	4.12
10	71	168	71	173	5
11	99	167	98	171	4.12

Figure 18 : Aperçu des tableaux de l'onglet 'Landmarks'

### 3.5.4 Corps de page de l'onglet des statistiques

Le corps de page de cet onglet est constitué de deux boutons et d'un espace prévu à l'affichage des graphiques [Figure 18]. Tous les éléments de cet section sont centrés. Le bouton 'Upload Csv' est un input de type file, lorsque l'utilisateur importe le fichier au format '.csv', le nom de ce fichier vient remplacer le champs initial du bouton. Ce dernier ainsi que le bouton 'Go' se situe en haut du corps de page, alignés. En dessous, se trouve l'emplacement prévu pour l'affichage des graphiques. Cet emplacement est 'vide', seul une instruction permettant d'aider l'utilisateur à afficher les graphiques, y est inscrite. Une fois que l'utilisateur a importé les données (fichier csv) et a cliqué sur le bouton 'Go', le premier graphique s'affiche [Figure 19]. Les graphiques s'affichent dans un carrousel. Ainsi l'utilisateur devra seulement appuyer sur la bouton de droite '>' pour afficher le graphique suivant et sur le bouton de gauche '<' pour afficher le graphique précédent.

Cela est rendu possible grâce à deux fonctions, présente dans le script JavaScript (cf scrip 'landmarksPage.js'),  $changeIndex(id)$  et  $changeImage(i)$ . La fonction  $changeIndex(id)$  va simplement définir, en fonction de l'action de l'utilisateur (clique sur le bouton précédent ou sur le bouton suivant), la position  $i$ . Cette position va être réutilisée dans la fonction  $changeImage(i)$  qui va permettre l'ajout du graphique, située à la position  $i$  dans la liste des graphiques, dans l'emplacement prévus à l'affichage des graphiques.

Comme pour les autres onglets, les couleurs qui prédominent sont le rouge, le bleu/vert ainsi que le gris. Les deux boutons présent dans l'onglet sont de couleur rouge et se foncent au passage du curseur de la souris. Lorsque l'utilisateur importe le fichier csv choisi, le nom de ce fichier remplace le champs initial du bouton. En dessous de ces boutons, l'emplacement prévus à l'affichage des graphiques est délimité par un encadré en pointillés de couleur bleu/vert. On y trouve les boutons permettant le défilement des graphiques, qui sont de couleurs bleu/vert aussi.

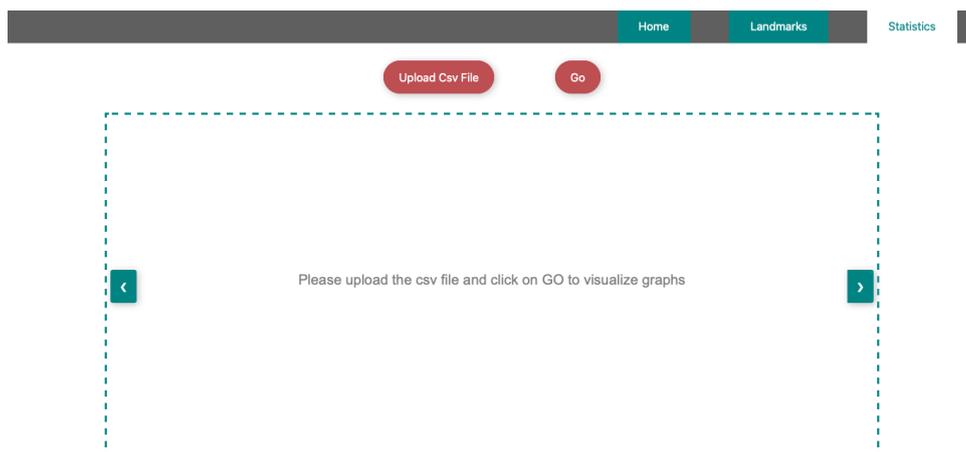


Figure 18 : Aperçu de l'onglet 'Statistics'

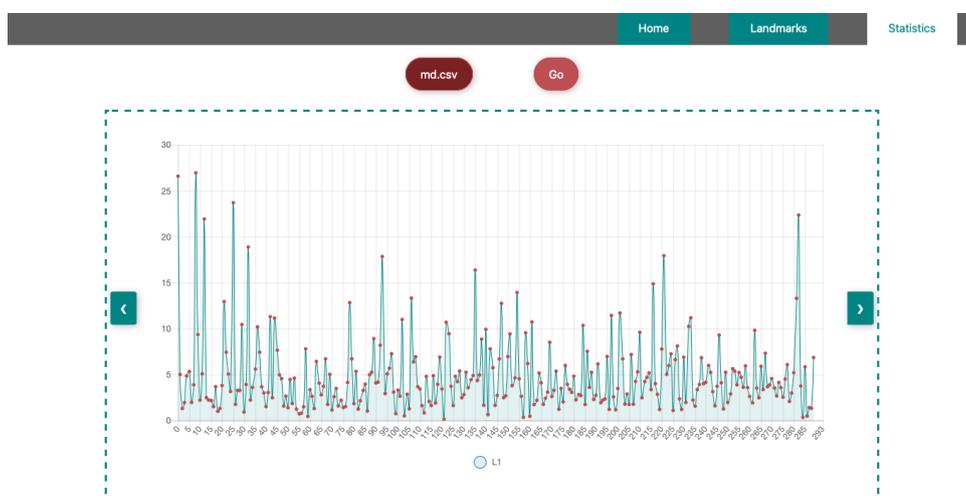


Figure 19 : Aperçu de l'onglet 'Statistics', après l'affichage des graphiques

### 3.5.5 Corps de page de l'onglet 'Home'

Cet onglet correspond à l'onglet de présentation et d'informations. On y retrouve alors, une brève description du projet, des informations concernant le scarabée, et quelques liens utiles [Figure 20]. Il est constituée de trois grandes sections. La première correspond à un brève rappel sur les fonctionnalités du site web. La seconde section, redéfinie et rappelle l'importance du machine learning et son rôle dans ce projet. La troisième section se situant en dessous de la seconde, contient les informations les plus importantes concernant les scarabées et une descriptions de leur organisation morphologique. La quatrième et dernière partie du corps de page de cet onglet se situe sur la partie droite de la page. Elle est constamment affichée à l'écran et suit le défilement de la fenêtre. On y retrouve des liens utiles vers des articles ou des livres par exemple.

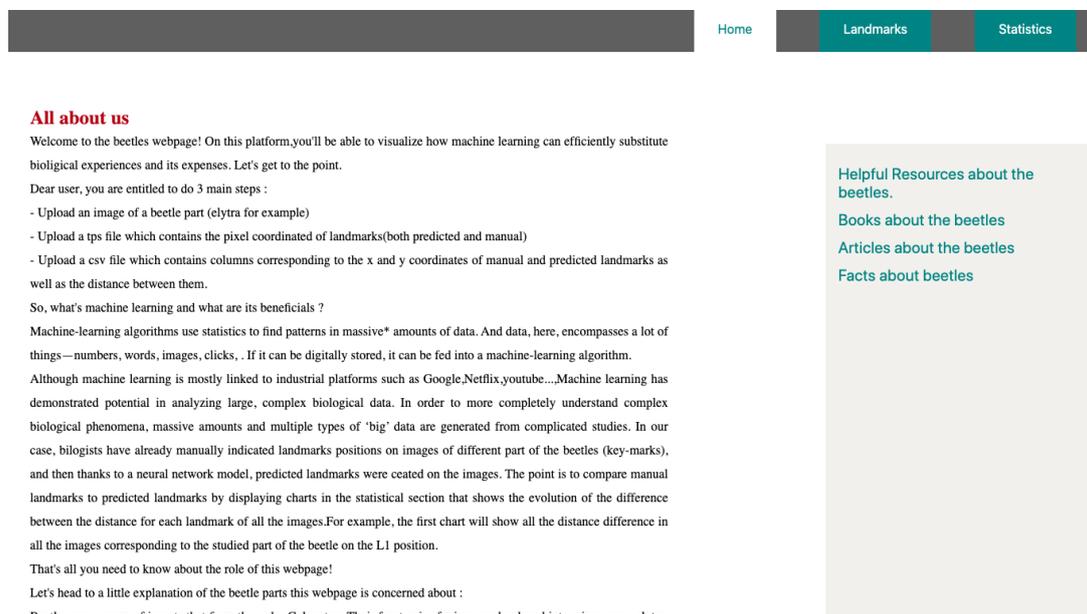


Figure 20 : Aperçu de l'onglet 'Home'

### 3.5.6 Dimensions et proportions de la page web

Afin que l'affichage de la page web se fasse correctement, peu importe la résolution de l'écran de l'utilisateur, la majorité des éléments de la page ont une largeur dites 'auto' ou alors une largeur définie en pourcentage qui s'adapteront à la résolution de l'écran de l'utilisateur.

# Chapitre 4

## Perspectives d'améliorations

### 4.1 Importation de répertoire

L'étude fournie un très grand nombre d'images, environs un milliers. En plus de cela, deux fichiers de type tps sont donnés pour chaque image, se traduisant par un très grand nombre de data nécessaires à l'importation. Par conséquent l'importation du répertoire entier en un seul clique serait un gain de temps considérable, l'utilisateur aura juste à cliquer sur la photo qui l'intéresse et la transférer dans la zone prévue à cet effet. Ainsi, pour une nouvelle image, l'utilisateur n'aurait plus la nécessité d'allé chercher une seconde fois une image dans son répertoire personnel puisque cette dernière aurait déjà été importée sur la page Web. Il n'aurait plus aussi à importer les fichiers au format '.tps' chacun leur tour, ils seraient détectés automatiquement via leur nom. Cela permettrait ainsi d'éviter plusieurs erreurs lors du choix des fichiers. L'image de sortie sera alors 'cohérente'. Car il est possible que l'utilisateur importe un fichier TPS, comprenant les coordonnées des points prédit, dédié à l'image 'Elytre001.JPG' par exemple et que le fichier TPS, contenant les coordonnées dis manuels, soit lui dédié à l'image 'Elytre002.JPG'. Ainsi, si l'image choisi par l'utilisateur est 'Elytre001.JPG', les landmarks manuels affichés ne seront pas cohérents étant donné qu'ils ne correspondent pas à cette image.

### 4.2 Création d'un compte personnel

Notre site web permet à l'utilisateur de télécharger l'image de sortie une fois les données affichées dessus. Il aurait été intéressant de rajouter la possibilité à l'utilisateur de créer un compte personnel sur le site web. Cela lui aurait permit par la suite, d'enregistrer tous les graphiques ainsi que les images 'modifiées' directement sur le site Web. Par conséquent, l'utilisateur pourra stocker toutes ses données dans une seule et même localisation. Pour ce faire, une base de données est indispensable. Cela nécessiterai en plus, l'utilisation d'autres langages tels que 'PHP' par exemple qui permettrait l'interaction

avec cette base de données.

## 4.3 Rendre l'application totalement fonctionnelle sur mobile

### 'Drag and Drop' de l'image

Afin de pouvoir afficher les landmarks sur l'image, l'utilisateur doit, normalement, prendre l'image importée et la déposer dans l'emplacement prévu.

Cependant, Lorsque le site web et plus précisément l'onglet 'Landmarks' est ouvert sur mobile, L'utilisateur ne peut pas effectuer l'action 'drag and drop', il faut donc que l'image importée s'affiche directement dans l'emplacement prévu.

# Conclusion

Ainsi, les landmarks dis manuels et ceux obtenues l'aide du machine learning semble être proche. En effet, les distances calculées entre ces points sont généralement inférieur à cinq pixels (sauf exception, Les landmarks numéro 16 du pronotum dont la distance entre les landmarks prédit et manuels est supérieur à six). Le modèle de machine learning utilisé semble alors efficace et précis. Les résultats des analyses faites par le site montre ainsi comment le machine learning est une bonne alternative aux quelques expériences biologiques parfois coûteuses.

Grâce à ce projet, nous avons pu comprendre et expérimenter les différentes étapes de la conception et la réalisation d'un site internet, allons de la réflexion, au fonctionnement jusqu'au design du site. Cela nous a donc permit d'être plus à l'aise avec les langages de programmation HTML5, CSS3 ainsi que JavaScript.

Nous avons appris à manipuler les données des fichiers csv et tps, à l'aide des langages html et javaScript, afin de s'en servir pour la réalisation des fonctions premières de notre site internet (visualiser les deux types de landmarks sur l'image des scarabées et visualiser les graphiques). Parallèlement, l'aspect esthétique à été primordiale. Effectivement, c'est le premier aspect que l'utilisateur voit lorsqu'il arrive sur le site web. De plus, il a permit de simplifier la compréhension et l'utilisation de l'application. Le langage CSS3 à été un outil indispensable pour cette réalisation.

De plus, l'utilisation de bibliothèques tels que chart.js, nous a permit d'appréhender d'autres outils alternatifs à un certain langage. En effet, 'chart.js' s'est avéré être une très bonne alternative au langage R puisqu'il nous a permit d'implémenter des graphiques interactifs et esthétiquement qualitatifs .

En général, les grandes lignes du projet ont été respectées mais des améliorations peuvent être encore réalisées.

Finalement, nous pouvons dire que le travail d'équipe a été très valorisant. En effet, chaque participant a réussi à apporter des notions supplémentaires lors de la réalisation du projet. Ce projet a été un bon exercice pour nos futurs carrières. Effectivement, l'apprentissage en profondeur de certains langages, la découverte de nouvelles bibliothèques ainsi que la collaboration entre tous les participants, sont des paramètres important, que l'on peut ajouter à notre bagage culturel et professionnel.

# Bibliographie

- **Estimating landmarks on 2D images of beetle mandibles**  
*Marie Beurton-Aimar, van Linh Le, Nicolas Parisey, Jean-Pierre Salmon*
  
- **Le guide complet CSS**  
*Fabien Basmaison, Antoine Cailliau, Jean-Rémy Duboc, février 2011*
  
- **JavaScript pour le Web 2.0**  
*Thierry Templier, ARNAUD GOUGEON, janvier 2007*
  
- **Apprenez les langages HTML5, CSS3 et JavaScript pour créer votre premier site web**  
*Denis MATARAZZO, septembre 2014*