

Titre: Développement d'un modèle paramétrable pour l'étude de
Title: l'influence de la position d'un cycliste sur son aérodynamisme

Auteur: Hadrien Koenig
Author:

Date: 2014

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Koenig, H. (2014). Développement d'un modèle paramétrable pour l'étude de
l'influence de la position d'un cycliste sur son aérodynamisme [Mémoire de
maîtrise, École Polytechnique de Montréal]. PolyPublie.
Citation: <https://publications.polymtl.ca/1555/>

 **Document en libre accès dans PolyPublie**
Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/1555/>
PolyPublie URL:

Directeurs de recherche: Delphine Périé-Curnier, & Jean-Yves Trépanier
Advisors:

Programme: Génie mécanique
Program:

UNIVERSITÉ DE MONTRÉAL

DÉVELOPPEMENT D'UN MODÈLE PARAMÉTRABLE POUR L'ÉTUDE DE
L'INFLUENCE DE LA POSITION D'UN CYCLISTE SUR SON
AÉRODYNAMISME

HADRIEN KOENIG

DÉPARTEMENT DE MÉCANIQUE

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE MÉCANIQUE)

OCTOBRE 2014

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

DÉVELOPPEMENT D'UN MODÈLE PARAMÉTRABLE POUR L'ÉTUDE DE
L'INFLUENCE DE LA POSITION D'UN CYCLISTE SUR SON AÉRODYNAMISME

présenté par : KOENIG Hadrien

en vue de l'obtention du diplôme de : Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de :

M. RAISON Maxime, Doctorat, président

Mme PÉRIÉ-CURNIER Delphine, Doctorat, membre et directrice de recherche

M. TRÉPANIER Jean-Yves, Ph.D., membre et codirecteur de recherche

M. BEGON Mickaël, Ph.D., membre

DÉDICACE

À tous mes amis qui m'ont supporté et aidé lors de cette aventure

REMERCIEMENTS

Tout d'abord, j'aimerais remercier Mme. Delphine Perié-Curnier, directrice de ce mémoire, ainsi que Mr. Jean-Yves Trépanier codirecteur, pour m'avoir fait confiance dans cet ambitieux projet de maîtrise, et pour en avoir assuré le bon déroulement.

Mes remerciements vont également à M. Eddy Petro pour ses précieux conseils, le temps qu'il m'a accordé à m'expliquer la CFD.

Je remercie aussi M. Philippe Débanné pour son aide précieuse lors des acquisitions ainsi que son aide lors du traitement des données.

Merci à Alexandre Pigeon et Hong Zhao pour leur aide sur mon projet.

Enfin, merci à ma famille et tous mes amis qui m'ont soutenu et aidé tout au long de ma maîtrise.

RÉSUMÉ

Le monde du cyclisme est un domaine en expansion constante du point de vue des performances, mais aussi au niveau de l'accessibilité des technologies de haut niveau au grand public. Par exemple, les cadres aérodynamiques professionnels se trouvent de plus en plus en vente pour le grand public. L'entreprise Guru Cycles a développé un système permettant de tester plusieurs positions du cycliste pour tenter d'optimiser le confort, la puissance développée et la position aérodynamique. Toutefois, l'aérodynamisme n'est pas quantifié, mais son évaluation repose sur l'expérience du positionneur et donc sur une évaluation visuel. Ainsi un outil permettant de calculer ou d'estimer quantitativement l'aérodynamisme des différentes positions était nécessaire pour compléter ce système.

Les objectifs du projet sont donc de développer et valider un ensemble d'outils permettant de connaître les performances aérodynamiques de n'importe quel cycliste. Pour cela un mannequin surfacique a été développé pour pouvoir réaliser une simulation des flux d'air autour de la surface obtenue. L'utilisation de méthode informatique pour simuler des problèmes portant sur des fluides est nommée la CFD (Computational Fluid Dynamics). Le mannequin paramétrique repose sur un ensemble de mesures obtenues grâce au système Kinect de Microsoft.

La modélisation de la surface extérieure du cycliste repose sur un squelette utilisant la modélisation multicorps pour permettre de modifier la position du mannequin. Ainsi plusieurs configurations du cycliste sur son vélo peuvent être simulées permettant ensuite de les comparer. La deuxième partie de la modélisation du modèle est la création d'ellipses s'appuyant sur le squelette. Ces ellipses servent comme support pour la surface, elles sont directement fixées sur le squelette et sont directement paramétrées par leurs petits et grands axes.

La personnalisation du modèle nécessite certaines mesures sur le sujet tel que la longueur des différents membres, mais aussi les dimensions des ellipses. Pour cela, l'acquisition consiste en une photo de face et une de profil du sujet grâce au système Kinect en plus d'une acquisition de calibration. La Kinect permet d'obtenir l'emplacement des articulations du sujet de façon automatique et de déduire le squelette du sujet. Les dimensions des ellipses sur les clichés peuvent être déduites en mesurant la largeur du membre voulu correspondant à l'emplacement de l'ellipse sur le modèle en s'appuyant sur les membres du squelette. La combinaison des deux clichés permet d'obtenir les deux dimensions de l'ellipse. Pour les longueurs des segments, la récupération des

longueurs des membres se fait de manière rapide par l'utilisation des emplacements des articulations. Les mesures sont injectées directement dans le modèle permettant d'obtenir le modèle personnalisé.

Le modèle est ensuite utilisé pour créer le maillage nécessaire à la simulation CFD. Dans un premier temps un domaine d'étude a été créé autour de la surface. Sa forme est une pyramide tronquée et penchée sur le côté. En utilisant ce domaine et la surface du sujet, un maillage de type non structuré est créé entre les deux pour modéliser l'air autour du cycliste. L'étape suivante est la préparation de la simulation CFD. La modélisation des turbulences du fluide choisie est une approche RANS avec un modèle k- ϵ .

Au cours du projet, deux validations sont nécessaires : une validation de l'ensemble mesure sur le sujet et mannequin, ainsi qu'une validation de la simulation CFD. La première validation repose sur une comparaison d'un scan 3D du sujet par un système d'acquisition 3D avec le modèle obtenu après personnalisation. Les distances normales à la surface du modèle entre les deux surfaces sont en moyenne de 15 à 20 mm alors que le maximum est de 35 mm. La deuxième validation repose sur une comparaison de résultat obtenu par la simulation CFD avec certains résultats obtenus par Defraeye (Defraeye, Blocken, Koninckx, Hespel, & Carmeliet, 2010a). La simulation donne des valeurs de 0,177 m² pour le produit du coefficient de traînée avec l'aire frontale qui est comprise entre les valeurs de la simulation de Defraeye (0,197 m²) et la valeur obtenue en soufflerie (0,175 m²).

Enfin, le modèle apporte beaucoup de perspective d'avenir grâce à sa capacité d'adaptation. Une utilisation du modèle dans le saut à ski par exemple, ou aussi l'utilisation de la Kinect pour la mesure des angles. Mais la modélisation du tronc est un des points à améliorer, notamment la détection de la lordose et de la cyphose pour recaler la courbure de la sinusoïde qui modélise le dos. D'autres améliorations sont possibles notamment l'ajout de degrés de liberté du modèle, et de type de morphologie possible à modéliser tel que la poitrine pour les femmes.

ABSTRACT

The world of cycling is a constantly expanding domain from the performance point of view but also regarding the accessibility of the high-level technologies. For example, the professional aerodynamic frames are more and more available to the general public. The Guru Cycles Company developed a system allowing to test several positions of the cyclist to try to optimize the comfort, the developed power and the aerodynamic position. With this system, during the positioning, the aerodynamics is not quantified but its empirical qualitative evaluation is rather based on the experience of the fitter. Therefore, a tool allowing to calculate or to estimate quantitatively the aerodynamics of the various positions was necessary to complete this system.

Thus, the objectives of the project are to develop and to validate a set of tools allowing to know the aerodynamic performances of any cyclist. To do so, a surface model was developed in order to realize a simulation of the airflow around the obtained surface. The use of computational method to solve fluid dynamics problems is named the CFD (Computational Fluid Dynamics). The parametric model relies on a set of measures obtained thanks to the Kinect system from Microsoft.

The modelling of the outside surface of the cyclist is based on a skeleton using the multi-body modelling to allow the modification of the position of the model. Thus, several configurations of the cyclist on his bike can be simulated, allowing then to compare them. The second part of the construction of the model is the creation of ellipses leaning on the skeleton. These ellipses serve as support for the surface. They are directly fixed to the skeleton and are directly parameterized by their major and minor axis.

The personalization of the model requires certain measures on the subject such as the length of the various members but also the dimensions of ellipses. For that purpose, the acquisition consists of two photos, one of front and one of profile of the subject with the Kinect. The Kinect allows to obtain the location of the articulations of the subject in an automatic way and to deduct the skeleton of the subject. The dimensions of the ellipses on acquisition can be realized by measuring the width of the desired member corresponding with the location of the ellipse on the model resting on the skeleton. The combination of both pictures allows to obtain both dimensions of the ellipse.

For the lengths of segments the recovery of the lengths of the members is made in a fast way by the use of the locations of the articulations. The measures are directly injected in the model

allowing to obtain the personalized model. The model is then used to create the mesh necessary for the CFD simulation. At first, a domain of study was defined around the surface, its shape being a cut pyramid tilted on the side. By using this domain and the surface of the subject, an unstructured mesh is created between both to represent the fluid region around the cyclist. The following step is the preparation of the CFD simulation. The chosen turbulence modelling is a RANS approach with a $k-\epsilon$ model.

During the project, two validations are necessary: a validation of the set of measures on the subject and the model, and a validation of the CFD simulation. The first validation rests on a comparison of a 3D scan of the subject by a system of 3D acquisition with the model obtained after personalization. The distances between both surfaces are on average 15 to 20mm while the maximum is of 35mm. The second validation rests on a comparison of results obtained by the CFD simulation with some results obtained by Defraeye (Defraeye et al., 2010a) The simulation gives values of $0,167\text{m}^2$ for the product of the coefficient of drag with the frontal air.

Finally, the model brings a lot of future perspectives thanks to its capacity of adaptation. A use of the model in the ski jumping for example, or also the use of Kinect for the measure of angles. But the modelling of the trunk is one point to be improved. In particular, the detection of lordosis and kyphosis to fit the curvature of the sinusoid that models the back. Other improvements are possible in particular the addition of degrees of freedom to the model, and the type of morphology possible to model such as the breast for the women.

TABLE DES MATIÈRES

DÉDICACE.....	III
REMERCIEMENTS	IV
RÉSUMÉ.....	V
ABSTRACT	VII
TABLE DES MATIÈRES	IX
LISTE DES TABLEAUX.....	XII
LISTE DES FIGURES	XIII
LISTE DES SIGLES ET ABRÉVIATIONS	XVI
LISTE DES ANNEXES.....	XVII
INTRODUCTION.....	1
CHAPITRE 1 ÉTAT DE L'ART.....	3
1.1 La CFD	3
1.2 Le mannequin	4
1.2.1 Création du mannequin par modélisation.....	5
1.2.2 Création du mannequin par scanner	6
1.3 Le corps humain	8
CHAPITRE 2 OBJECTIFS	11
2.1 Résumé de la problématique de recherche	11
2.2 Objectifs de recherche	11
CHAPITRE 3 MÉTHODES	12
3.1 Modèle CAD du cycliste	12
3.1.1 Principe global de modélisation	12
3.1.2 Modélisation du Squelette	15

3.1.3 Modèle surfacique des jambes	19
3.1.4 Modélisation du Torse.....	24
3.1.5 Modélisation des membres supérieurs	33
3.1.6 Assemblage des surfaces	35
3.2 Le système de mesure	36
3.2.1 La Kinect	36
3.2.2 Traitement des acquisitions	37
3.3 Le maillage.....	43
3.3.1 Création de la géométrie	43
3.3.2 La création du maillage	44
3.4 La Simulation	50
3.4.1 Le modèle du fluide.....	50
3.4.2 Les conditions aux limites.....	50
3.4.3 L'extraction des résultats	51
3.5 Les validations.....	54
3.5.1 La validation géométrique.....	54
3.5.2 La validation CFD	56
CHAPITRE 4 RÉSULTATS	58
4.1 Validation géométrique	58
4.2 Validation CFD	63
CHAPITRE 5 DISCUSSION	66
5.1 Mannequin.....	66
5.2 Kinect	67
5.3 Choix de la modélisation CFD	69

CONCLUSION	70
BIBLIOGRAPHIE	72
ANNEXES	74

LISTE DES TABLEAUX

Tableau 1-1 : Les articulations et leurs mouvements.....	9
Tableau 3-1: Liste des degrés de liberté modélisés.....	14
Tableau 3-2 : Valeurs des angles entre les vertèbres.	17
Tableau 3-3 : Emplacement des plans de définitions des esquisses du tronc.	25
Tableau 3-4 : Valeurs du paramètre $y(x)$	27
Tableau 3-5 : Emplacements des mesures des ellipses sur les membres.	41
Tableau 3-6 : Avantages et inconvénients des types d'élément.	45
Tableau 3-7 : Différentes densités de maillage testées.	46
Tableau 4-1 : Distance maximale entre le du scan et le modèle.	61
Tableau 4-2 : Valeurs des coefficients ACD simulés et de l'article de Defraeye.	63

LISTE DES FIGURES

Figure 1-1 : Modèle de la surface du corps humain (a) (Hanavan Jr, 1964), b) (Yeadon, 1990), c) (Hatze, 1980),d) (Jensen, 1978)).....	5
Figure 1-2 : le scanner Handyscan	7
Figure 1-3 : Le scanner Inspeck.....	7
Figure 1-4 : Les différents plans de référence du corps humain.....	8
Figure 3-1 : Squelette final du modèle et ellipses permettant la création de la surface.....	13
Figure 3-2 : Modélisation de la colonne vertébrale.....	16
Figure 3-3 : Squelette d'une jambe.....	18
Figure 3-4 : Éléments filaires de la cuisse et surface obtenue.	20
Figure 3-5 : Éléments filaires du tibia et surface obtenue.....	21
Figure 3-6 : Éléments filaires de l'avant du pied et surface obtenue.....	21
Figure 3-7 : Éléments filaires de la cheville et surface obtenue.....	22
Figure 3-8 : Interface entre les deux membres inférieurs.....	23
Figure 3-9 : Éléments filaires du tronc et surface obtenue.....	24
Figure 3-10 : Création d'une ellipse du tronc	25
Figure 3-11 : Exemple de forme du tronc dans le plan sagittal et dans le plan frontal (dimension en mm).	26
Figure 3-12 : Prise de mesure sur les éléments de construction.....	28
Figure 3-13 : Éléments filaires de l'épaule et surface obtenue.....	31
Figure 3-14 : Éléments filaires de la tête et surface obtenue.	32
Figure 3-15 : Éléments filaires du bras et surface obtenue.	33
Figure 3-16 : Éléments filaires de l'avant-bras et surface obtenue.....	34
Figure 3-17 : Surface finale des deux membres supérieurs.	34
Figure 3-18 : Modèle surfacique obtenu dans une position sur un vélo et une position debout....	35

Figure 3-19 : Détection automatique du squelette avec la Kinect.....	36
Figure 3-20 : Les différentes étapes de la segmentation : a) Image d'origine, b) Segmentation, c) Épaississement des contours de segmentation, d) Remplissage de la segmentation, e) Lissage du remplissage pour supprimer le bruit.....	38
Figure 3-21 : Image du calibre avant et après segmentation.....	39
Figure 3-22 : Numérotation des articulations sur le squelette.....	40
Figure 3-23 : Squelette (a) après repositionnement et b) avant).	40
Figure 3-24 : Orthogonalité des mesures des ellipses sur les membres.	41
Figure 3-25 : Le domaine d'étude (Inlet : surface d'entrée du fluide ; Outlet, surface de sortie du fluide ; Bottom : Surface inférieure modélisant le sol ; Top : Surface supérieure ; Right et Left : Surface à droite et à gauche du modèle; Part_1 : Surface importée du cycliste).	43
Figure 3-26 : C_D en fonction de l'indice de densité du maillage.....	46
Figure 3-27 : Maillage surfacique du modèle (a) et du domaine (b).....	48
Figure 3-28 : Maillage de la couche limite.	48
Figure 3-29 : Coupe du maillage du domaine.....	49
Figure 3-30 : Vecteur vitesse (m/s) dans le plan ZY (a) et dans le plan XZ (b).	51
Figure 3-31 : Courbe de convergence du Cd (sans unité) en fonction des itérations.....	52
Figure 3-32 : Cartographie de $y+$ sur la surface du cycliste.....	53
Figure 3-33 : Le système Inspect : Une caméra Inspeck (gauche), un ensemble de quatre caméra Inspeck (droite) : une de face, deux à 30° de chaque côté et une à l'arrière du sujet.	54
Figure 3-34 : Différentes positions du cycliste (Article de Defraeye (Defraeye et al., 2010a)) : a) Position haute, b) Position aérodynamique, c) Position contre la montre.....	56
Figure 3-35 : Position du modèle (gauche) et position dans l'article de référence (droite : image tirée de l'article de Thijs Defraeye (Defraeye et al., 2010a)).	57
Figure 4-1 : Superposition du scan et du modèle.....	58

Figure 4-2 : Mesure normale à la surface du modèle entre les deux surfaces obtenues sur le haut du corps	59
Figure 4-3 : Mesure normale à la surface du modèle entre les deux surfaces obtenues sur une jambe.	60
Figure 4-4 : Mesure de haut du corps du sujet 5	62
Figure 4-5 : Coefficient de pression de la position haute (sans unité)	64
Figure 4-6 : Coefficient de pression de la position aérodynamique.....	65

LISTE DES SIGLES ET ABRÉVIATIONS

CFD	Computational Fluid Dynamics
STEP	Standard for the Exchange of Product model data
RANS	Reynolds Average Navier-stokes
LES	Large-Eddy Simulation

LISTE DES ANNEXES

ANNEXE A - Liste des paramètres.....	74
ANNEXE B - Code du Logiciel.....	76
ANNEXE C - Code Matlab.....	91

INTRODUCTION

Dans notre société moderne, l'amélioration des performances est un objet de recherche perpétuelle. Un domaine particulièrement concerné est l'aérodynamisme de nos moyens de transport. Mais, contrairement aux idées reçues, cette recherche de performance ne concerne pas exclusivement l'aviation et l'automobile. Elle entre aussi en jeu dans le cadre d'autres transports plus communément associés aux loisirs, tels que le cyclisme.

Les performances d'un cycliste sont très dépendantes de son aérodynamisme et plus particulièrement de la force créée par la résistance de l'air s'opposant à son déplacement, appelée force de traînée. Comme le montrent Grappe et coll. (Grappe, Candau, Belli, & Rouillon, 1997), ainsi que Kyle et Burk (Kyle & Burke, 1984), au-delà de 50 km/h, au moins 90 % de la force resistive au mouvement d'un cycliste est due à cette force de traînée.

Les améliorations peuvent porter sur plusieurs points, certains sont matériels comme le cadre et les accessoires, et d'autres sont biomécanique comme la position du cycliste. Les améliorations matérielles ont été le sujet de nombreuses recherches par les compagnies et les universités et sont déjà bien optimisées. Le deuxième type d'amélioration est beaucoup plus complexe, des critères comme le confort et la puissance développée devant être pris en compte. Mais ce deuxième type d'amélioration est important, car l'humain représente 70 % de la force de traînée selon Gross et coll. (Gross, Kyle, & Malewicki, 1983), ainsi que Kyle et Burk (Kyle & Burke, 1984). Une petite modification sur la position du cycliste peut avoir une grande influence sur sa traînée aérodynamique ((Broker, 2003), (García-López et al., 2008), (Grappe et al., 1997), (Jeukendrup & Martin, 2001)).

Avant l'avènement de l'informatique, la méthode expérimentale était la principale solution pour pouvoir analyser des écoulements gazeux. La soufflerie était alors utilisée fréquemment en recherche, et en particulier dans le cas du cyclisme. Cette méthode comporte plusieurs inconvénients. Le premier est qu'elle ne donne que peu de détails sur l'écoulement. Ce manque d'information sur les champs des paramètres est dû à la complexité de la mise en œuvre de mesures dans la soufflerie. L'autre problème est le prix des expériences, car une soufflerie est une installation lourde qui demande une main d'œuvre expérimentée et une préparation des installations.

Une autre méthode de plus en plus utilisée est l'implémentation d'un modèle informatique pour la simulation afin de connaître l'écoulement dans ses moindres détails. Cette méthode est appelée communément : CFD (Computational Fluid Dynamics : Mécanique des fluides numériques). Couramment utilisée dans l'aéronautique et l'automobile, elle se démocratise de plus en plus dans d'autres domaines, notamment dans le sport. Ainsi on trouve des études en natation ((Barry Bixler, Pease, & Fairhurst, 2007; Barry Bixler & Riewald, 2002; B Bixler & Schloder, 1996)), en cyclisme (Defraeye et al., 2010a; Defraeye, Blocken, Koninckx, Hespel, & Carmeliet, 2010b, 2011), et dans d'autres sports de haut niveau.

Dans le cas du cyclisme, la CFD est souvent utilisée, mais elle comporte des limites. Les études faites se concentrent généralement sur un nombre limité de cyclistes et positions. Ainsi ils utilisent des modèles statiques non paramétrables souvent créés à partir de nuages de points.

L'acquisition du nuage de point par les méthodes de rétro-ingénierie, telles qu'un scanner laser, demande du temps et des installations coûteuses. Ces techniques sont difficiles à appliquer pour des cyclistes amateurs, et une alternative moins coûteuse et plus rapide est nécessaire. Ainsi le but du projet est de créer un ensemble d'outils permettant de réaliser une simulation CFD personnalisée d'un cycliste.

Le mémoire se sépare en six parties. La première fait état de l'art de différentes modélisations du corps humain existantes. La deuxième met en place les différents objectifs du projet. La méthode de construction des différents outils développés pour répondre aux objectifs du projet est exposée dans le troisième chapitre. Enfin, les résultats des différentes validations des outils créés sont présentés dans la quatrième partie. Le cinquième constitue une analyse critique des méthodes employées et des résultats obtenus. La dernière partie donne des recommandations pour la suite du projet.

CHAPITRE 1 ÉTAT DE L'ART

1.1 La CFD

La CFD est un domaine très complexe et en amélioration continue. Elle a pour but de modéliser et simuler des écoulements fluides grâce à l'outil informatique. Dans l'ensemble des paramètres comparés, le coefficient de traînée et le coefficient de la force tangentiel au déplacement sont les plus utilisés pour décrire les performances aérodynamiques en cyclisme. La force de traînée correspond à la force qu'oppose l'air contre le déplacement du cycliste, alors que la force tangentielle correspond à la force de côté qui peut déséquilibrer le cycliste dans son déplacement.

De nombreuses techniques ont été proposées ayant des degrés de simplification des phénomènes de turbulence plus ou moins importants. La première approche est la modélisation RANS (Reynolds Average Navier-Stokes). Elle repose sur les équations de Navier-Stokes et consiste à résoudre un écoulement moyen. Cette modélisation se trouve être la plus populaire en CFD. De nombreux modèles de turbulence sont utilisés, mais les deux plus importants sont les modèles $k-\varepsilon$ et $k-\omega$. Ces deux modèles reposent sur l'ajout de deux équations de transport pour modéliser l'effet des turbulences sur l'écoulement moyen. La première équation représentée par le k est une équation sur l'énergie cinétique alors que la deuxième correspond à la dissipation turbulente représentée par ε ou la dissipation spécifique représentée par ω . La deuxième approche de modélisation nommée LES (Large Eddy Simulation), repose sur la filtration des échelles de turbulences plus petites que le maillage, alors que celles plus grandes sont entièrement simulées. La modélisation des turbulences LES donne en générale de meilleurs résultats mais demande un temps de calcul plus grand par rapport à la modélisation RANS.

Les paramètres couramment extraits puis comparés sont les différentes forces créées par la présence du fluide sur le cycliste, mais aussi la pression. La formule qui régit le calcul de la force F est :

$$F = \rho \frac{U^2}{2} AC \quad (1.1)$$

ρ étant la masse volumique de l'air, U la vitesse du fluide, A l'aire de l'objet, C le coefficient de force et F la force. Les comparaisons entre différentes simulations se font en général sur le produit AC pour la force de traînée pour éviter d'avoir à déterminer l'aire frontale de l'objet étudié.

Au niveau de la simulation du cas d'un cycliste, Thijs Defraeye et coll.(Defraeye et al., 2010b) abordent ce sujet en comparant deux approches de modélisations de turbulence : RANS et LES. Les auteurs ont comparés trois positions différentes du cycliste (Defraeye et al., 2010a). Un des points importants de ces modélisations est le domaine d'étude qui correspond au volume autour du cycliste à simuler. Les dimensions du domaine qu'ils utilisent sont 12,7 m de profondeur, 3 m de large et 2,25 m de hauteur, et la forme du domaine est un pavé. La taille du domaine d'étude utilisé s'appuie sur ces dimensions comme dimensions de base.

L'étude de Defraye et coll. sur les différentes modélisations de turbulence a montré que la modélisation $k-\omega$ apporte les meilleurs résultats, mais que la modélisation $k-\varepsilon$ est une bonne alternative. Cette deuxième modélisation est celle aussi la plus répandue dans les études CFD dans le sport tel que Bixier et coll. en natation (Barry Bixler et al., 2007; Barry Bixler & Riewald, 2002; B Bixler & Schloder, 1996) ou encore dans le football (Barber, Chin, & Carré, 2009). Les auteurs montrent aussi que le choix de la loi de paroi est important. De plus, ils montrent qu'il est mieux de privilégier une modélisation de la couche limite par rapport à une approximation. L'approximation utilise une loi pour déterminer les valeurs internes à la couche limite, alors que la modélisation calcule ces valeurs.

1.2 Le mannequin

La création d'une surface 3D représentant le corps humain d'un sujet peut être effectuée selon deux méthodes, l'une reposant sur une modélisation de la surface par des surfaces mathématiques et des mesures sur le sujet, l'autre consistant à réaliser un scan 3D du sujet par des méthodes optiques.

1.2.1 Cration du mannequin par modelisation

Depuis longtemps, des recherches scientifiques ont tente de reproduire la forme du corps humain par des modles parametriques. Certains se veulent trs complexes et le plus proche de la ralit, alors que dautres tentent de simplifier au maximum les paramtres. Le premier type de mannequin essaye de modliser le comportement inertiel du corps humain. Pour cela, une approximation de la surface extrieure par des volumes plus ou moins complexes est faite. Les dimensions des volumes proviennent de mesures anthropomtriques pour une majorit de modles, dont les suivants sont les plus connus :

- Hanavan 15 volumes poly articuls (Hanavan Jr, 1964)
- Yeadon 40 volumes, 95 mesures anthropomtriques (Yeadon, 1990)
- Hatze 246 mesures, 1 volume complexe (Hatze, 1980)

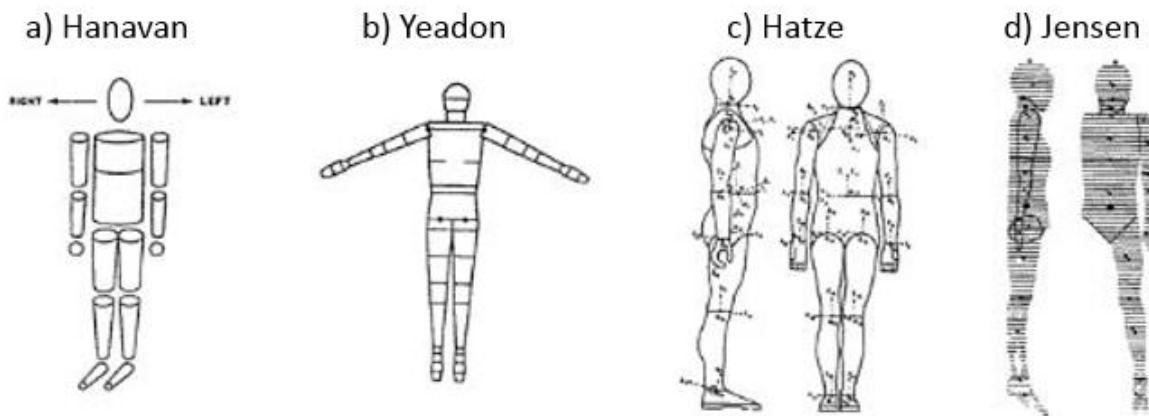


Figure 1-1 : Modle de la surface du corps humain (a) (Hanavan Jr, 1964), b) (Yeadon, 1990), c) (Hatze, 1980), d) (Jensen, 1978)).

La figure 1-1 illustre la forme finale de ces modles. Le premier modle, le plus simpliste (modle a), repose sur quinze volumes polyarticuls, mais ne comporte pas de continuit entre les membres. Vingt-cinq mesures anthropomtriques sont ncessaires  la cration de ce modle qui repose sur des formes gometriques simples telles que des cylindres ou des cones tronqus.

Le deuxième modèle proposé par Yeadon (Yeadon, 1990) (modèle b), est composé de 40 volumes et nécessite 95 mesures anthropométriques. Il repose aussi sur des formes simples comme le modèle précédent. Il est donc plus complexe, mais reste assez simpliste.

Le troisième modèle, celui de Hatze (Hatze, 1980) nécessite 246 mesures anthropométriques, ce qui le fait devenir un des plus complexes, mais un des plus fidèles au corps humain.

Le quatrième modèle, développé par Jensen (Jensen, 1978), ne nécessite pas de mesures directes. Ce modèle est basé sur deux photos prises du sujet en position debout, de face et de profil. Le corps est modélisé par une succession d'ellipses qui servent de support pour la surface modélisant l'enveloppe externe du corps du sujet. Les deux photographies sont segmentées tel qu'illustré sur la figure 1-1-d. Les dimensions des segments correspondants dans les deux images sont utilisées pour créer les ellipses et ainsi obtenir la surface du corps.

D'autres modèles sont utilisés dans l'industrie du textile pour la visualisation de l'apparence de vêtements sur le sujet. Le modèle développé par Kasap (Kasap, 2011) repose sur des dimensions qui influencent de façon visuelle le modèle, telles que des longueurs ou la circonférence des membres. Ce modèle permet surtout de représenter de façon visuelle le sujet dans des logiciels spécialisés tel que 3DS max.

1.2.2 Crédit du mannequin par scanner

La deuxième méthode de création de la surface extérieure d'un sujet consiste à utiliser un scanner optique. Les plus courants utilisent un laser qui balaye la surface et une caméra qui détecte la déformation du faisceau et ainsi permet de déterminer la forme de la surface. Ce type de scanner a été utilisé par Thijs Defraeye (Defraeye et al., 2010a) pour obtenir la surface utilisée lors de ses analyses CFD. Un exemple de scanner laser est illustré à la Figure 1-2, développé par Creaform.



Figure 1-2 : le scanner Handyscan¹.

Dans le cadre de ce projet, un autre type de scanner a été utilisé. Nommé Inspeck, ce scanner utilise la lumière structurée comme méthode optique : un projecteur émet une succession de franges noires et blanches sur la surface à analyser et une caméra enregistre une photo des déformations des franges sur la surface. Après traitement de l'image dans un logiciel, la déformation des franges permet d'obtenir un maillage de la surface scannée.



Figure 1-3 : Le scanner Inspeck.

Ce type de modélisation permet d'obtenir un nuage de points de la surface scannée. Le sujet doit par conséquent avoir la position à étudier lors de l'acquisition, car le résultat de l'acquisition est un modèle fixe.

¹ Image consultée le 01-09-14 à l'adresse <http://www.creaform3d.com/fr/solutions-de-metrologie/scanners-3d-portables-handyscan-3d>

1.3 Le corps humain

Les trois plans couramment utilisés dans le domaine biomédical sont définis sur la figure 1-4. Ces plans permettent d'avoir un référentiel du corps humain.

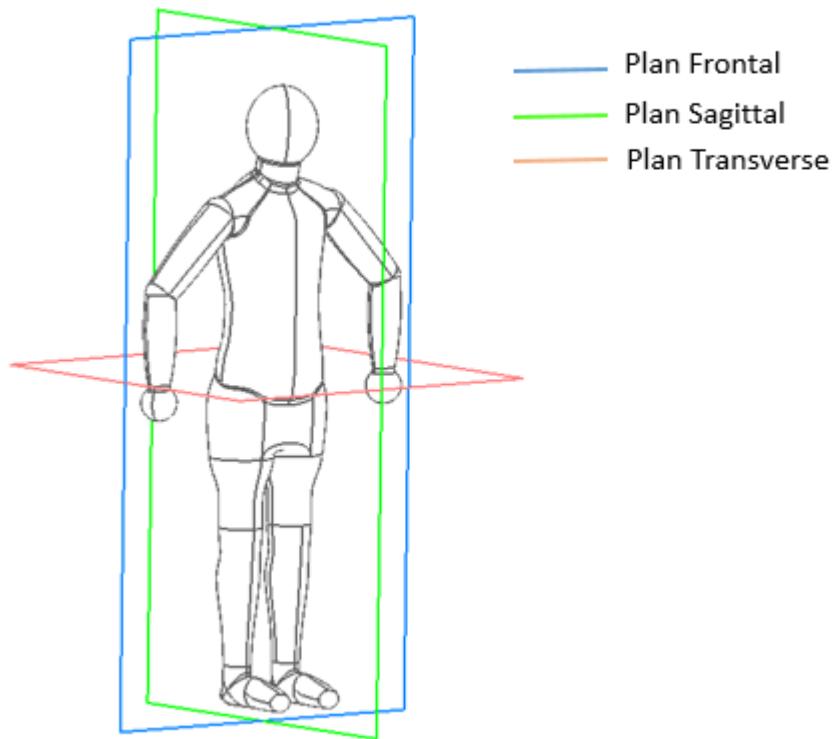


Figure 1-4 : Les différents plans de référence du corps humain.

Le corps humain est composé de plusieurs centaines d'articulations et leurs degrés de mobilité sont variables. Seuls les articulations utiles au projet et les degrés de liberté modélisés vont être listées dans le tableau 1-1.

Tableau 1-1 : Les articulations et leurs mouvements.

Articulation	Mouvement	Description du mouvement
Coude	Flexion/extension,	Pliement du coude
Épaule	Abduction/adduction	Rotation du bras dans le plan frontal
	Rotation médiale/latérale	Rotation du bras se basant sur le corps rigide du bras pour axe
	Flexion/extension	Rotation du bras dans le plan sagittal
Hanche	Flexion/extension	Mouvement dans le plan sagittal
Genou	Flexion/extension	Mouvement dans le plan sagittal
Cheville	Dorsiflexion/ flexion plantaire	Mouvement dans le plan sagittal
Cervicale	Flexion/extension	Mouvement dans le plan sagittal
Lombaire	Flexion/extension	Mouvement dans le plan sagittal
Thoracique	Flexion/extension	Mouvement dans le plan sagittal

La colonne vertébrale est une succession d'os de tailles différentes nommés vertèbres, reliées les unes aux autres par des disques intervertébraux. Les disques donnent la dimension déformable de la colonne. Selon le Gray's anatomy (Gray, 1918), la colonne mesure en moyenne 71 cm pour les hommes. La colonne est constituée de quatre types de vertèbres, nommées de bas en haut, sacrées, lombaires, thoraciques et cervicales, dont les tailles moyennes sont respectivement 12,5 cm, 18 cm, 28cm et 12,5 cm.

Dans la région lombaire, Nissan et Gilad (Nissan & Gilad, 1986) constatent sur 157 colonnes lombaires normales d'hommes âgés entre 20 et 38 ans que la hauteur moyenne du corps vertébral se trouve autour de 34 mm.

La dimension des disques intervertébraux doit être prise en compte dans la création de la colonne vertébrale dans le logiciel de modélisation. Selon le livre « Anatomie et physiologie humaines » (MARIEB, 2004), 25 % de la hauteur de la colonne est constituée des disques intervertébraux.

Ces différentes dimensions seront utilisées afin de pouvoir modéliser le squelette de la colonne vertébrale dans la partie 3.1.2.1.

CHAPITRE 2 OBJECTIFS

2.1 Résumé de la problématique de recherche

La commercialisation à plus grande échelle de cadres de vélo haut de gamme pousse les industriels à rechercher les performances maximales pour se démarquer de leurs concurrents. Présentement, la source d'amélioration des performances vient généralement du cadre, mais une nouvelle approche émergente consiste à personnaliser les dimensions du cadre pour que le cycliste adopte une position optimale pour minimiser la force de trainée.

Le but de ce projet est de développer un ensemble d'outils permettant d'obtenir des données quantitatives pour la comparaison de différentes positions. L'idée originale est d'utiliser un modèle paramétrique articulé reposant sur des mesures anthropométriques.

2.2 Objectifs de recherche

Objectif principal : L'objectif principal de ce projet de recherche est de développer un modèle numérique permettant de comparer de façon quantitative plusieurs positionnements d'un cycliste dépendant des dimensions du cadre.

Objectifs spécifiques : Les objectifs spécifiques définis pour atteindre cet objectif principal sont :

- Réalisation d'un système de mesures anthropométriques d'un cycliste
- Création d'un modèle CAO paramétrique s'appuyant sur les mesures anthropométriques et incluant des articulations nécessaires à la création de différents positions
- Utilisation de la CFD pour obtenir des valeurs de comparaisons des positions
- Validation du système de mesure et du modèle CAO par comparaison du sujet scanné et du modèle créé
- Validation de la simulation CFD par comparaison des résultats avec la littérature.

CHAPITRE 3 MÉTHODES

Ce chapitre traite des différentes méthodes utilisées dans le cadre de ce mémoire. La première méthode comprend la création du mannequin 3D. La deuxième traite de l'acquisition des données anthropométriques. La troisième détaille la génération du maillage, et la quatrième donne les paramètres choisis pour la simulation. La dernière méthode traite des méthodes de validation du modèle et de la simulation CFD.

3.1 Modèle CAD du cycliste

Cette partie traite de la méthodologie utilisée pour créer le mannequin 3D. Les principes de modélisation utilisés seront dans un premier temps exposés pour ensuite entrer dans le détail de la modélisation de chaque membre du corps. Tout au long de cette partie, certaines données ont été mesurées sur différents modèles du corps humain existants étant donné qu'elles étaient difficiles à obtenir. Un des modèles utilisés tout au long de la création du modèle est le mannequin du module d'ergonomie de Catia (Dassault Systèmes, Vélizy-Villacoublay, France).

3.1.1 Principe global de modélisation

La modélisation se décompose en deux parties, la création d'un squelette filaire permettant de modéliser les mouvements pris en compte dans le modèle et la création de la surface externe. Le squelette est basé sur un modèle multicorps illustré à la figure 3-1. Lors de sa création, les membres sont paramétrés en longueur pour obtenir un squelette adaptable à la taille du cycliste à modéliser, et un paramétrage en angles est aussi nécessaire pour pouvoir positionner le modèle comme désiré. La création de la surface externe utilise comme base le squelette. Elle est aussi paramétrée dans le but de représenter la corpulence du sujet. Les ellipses servent de lignes de passage à la surface modélisée du sujet.

La création du squelette nécessite le choix des degrés de liberté qui seront modélisés. Tous les degrés de liberté physiologiques ne seront pas modélisés afin d'éviter une trop grande complexité du modèle. Le choix de ces degrés de liberté est basé sur les mouvements classiques d'un cycliste en position aérodynamique. C'est-à-dire que le cycliste est en position assise sur la selle avec les mains et les avant-bras sur le guidon et le dos penché vers l'avant presque à

l'horizontale. Ainsi tous les mouvements latéraux du tronc et des jambes sont bloqués pour garder la symétrie du corps. Les mouvements des bras sont identiques pour la même raison, alors que seuls ceux des jambes sont dissociés. Un dernier angle est considéré pour modéliser l'inclinaison de l'ensemble du modèle dans le plan sagittal, ce paramètre a été nommé : Inclinaison_Globale.

Une fois les degrés de liberté définis, le deuxième type de paramètre à implémenter dans le modèle est la longueur des membres. En plus de sept longueurs de membres, deux largeurs sont nécessaires pour définir entièrement le squelette, nommées Largeur_Epaule et Largeur_Hanche.

L'ensemble des éléments nécessaires à la création du modèle multicorps est listé dans le tableau 3-1. L'emplacement des ellipses est un pourcentage du segment correspondant, le point à 0 % correspond au point du début du membre et 100 % vers l'extrémité du membre.

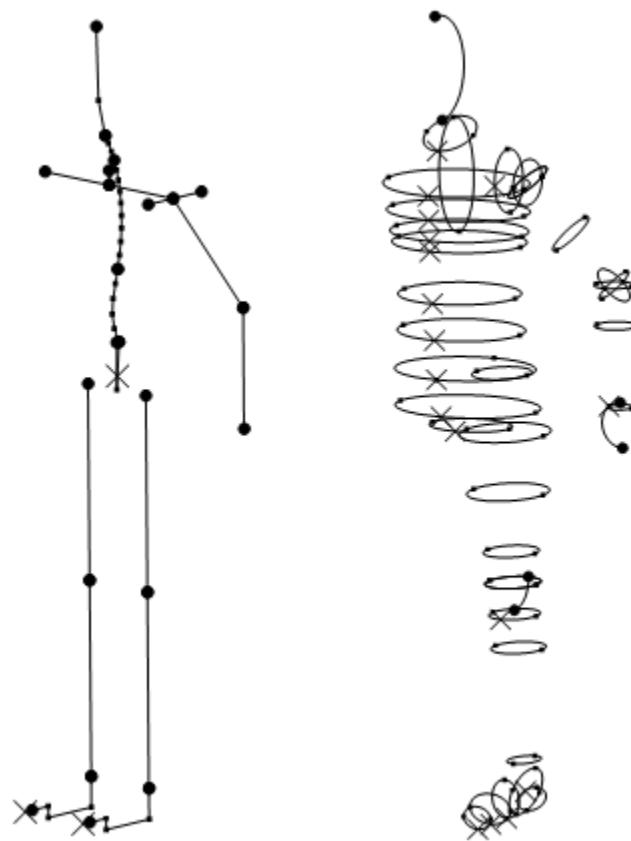


Figure 3-1 : Squelette final du modèle et ellipses permettant la création de la surface.

Tableau 3-1: Liste des degrés de liberté modélisés.

Corps rigide	Longueur	Emplacement des ellipses de construction (%)	Articulation précédant le corps	Degrés de liberté de l'articulation
Avant-bras	Longueur_AvantBras	0 33,33 100	Coude	Angle_Flexion_AvantBras
Bras	Longueur_Bras	0 50 100	Épaule	Angle_Flexion_Bras Angle_Abduction_Bras Angle_RotationLateral_Bras
Cervicale	Longueur_Tete			Angle_Tete
Colonne vertébrale	Longueur_Torce	Voir partie 3.1.4		
Cuisse	Longueur_Cuisse	0 30 60 90 100	Hanche	Angle_Jambe_D Angle_Jambe_G
Tibia	Longueur_Tibia	0 15 33,33 90 100	Genou	Angle_Genou_G Angle_Genou_D
Pied	Longueur_Pied	0 35 65 100	Cheville	Angle_Pied_G Angle_Pied_D

Le squelette étant créé, la surface autour de ce dernier est alors construite selon une méthode inspirée du travail de Jensen (Jensen, 1978). Chaque membre est segmenté de façon orthogonale au corps rigide du squelette le définissant, et à des pourcentages définis des corps rigides (Tableau 3-1). La méthode de mesure utilisée pour segmenter et mesurer les corps rigides est celle développée dans la partie 3.2.2. Une fois que les dimensions sont récupérées, les ellipses sont construites aux emplacements correspondants. Enfin, entre les différentes ellipses de chaque membre, une surface multisection est utilisée pour modéliser la surface du corps.

3.1.2 Modélisation du Squelette

Tout au long de cette partie, les dimensions des vertèbres ainsi que les angles en position verticale ont été mesurés sur un modèle de la colonne vertébrale.

3.1.2.1 La colonne

Chaque vertèbre constitue un corps rigide de même pour le sacrum, sauf les cervicales qui ont été modélisées comme un seul corps. Au niveau de la construction, la colonne est décomposée en quatre parties (Figure 4-1) :

- les vertèbres lombaires représentées par cinq corps rigides
- les vertèbres thoraciques représentées par douze corps rigides
- le sacrum représenté par un seul corps rigide
- les vertèbres cervicales et la tête représentées par deux corps rigides

La référence du modèle se situe au niveau de l'extrémité inférieure du sacrum. La composante Z correspond à la verticale et la composante X correspond à la direction postéro-antérieure.

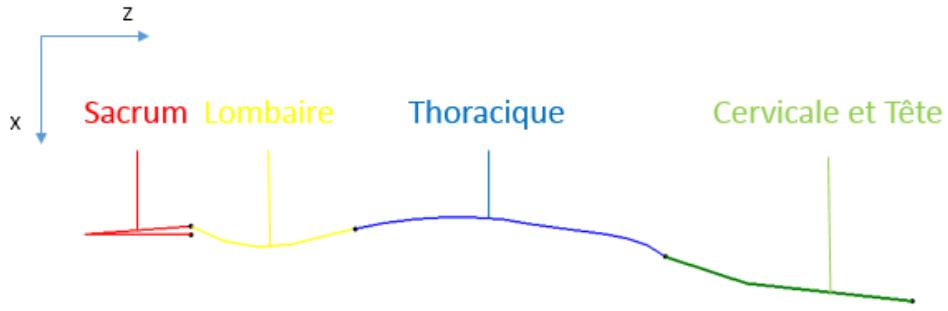


Figure 3-2 : Modélisation de la colonne vertébrale.

Pour modéliser complètement la colonne, deux paramètres sont nécessaires sur chaque corps rigide, sa longueur et l'angle qu'il forme avec le corps rigide qui le précède. Pour le sacrum, comme aucun élément ne le précède, l'angle qui le détermine est défini par rapport à une droite représentant la verticalité du modèle dans le repère local du modèle. Le paramètre Inclinaison_Global modifie ce repère en faisant une rotation autour du point inférieur du sacrum autour de l'axe \vec{y} .

La longueur des corps rigides est définie par une fraction de la longueur totale du tronc (paramètre Longueur_Tronc), sauf pour le corps rigide des cervicales et celui de la tête qui sont liés à un autre paramètre (Longueur_Tete). La fraction de chaque vertèbre, notée $y(x)$, est définie par l'équation 3.1, avec x le numéro de la vertèbre (1 correspond à L5, 2 à L4, ..., 16 à T2 et 17 à T1).

$$y(x) = -0,0016x + 0,0634 \quad (4.1)$$

$$L_x = y(x) * \text{Longueur_Tronc} \quad (4.2)$$

L'équation 4.2 donne la longueur de la vertèbre x en fonction du paramètre Longueur_Tronc. La fraction du sacrum est $y(0) = 0,1832$, sa longueur est calculable par l'équation 4.2. Cette valeur est un rapport de la longueur moyenne du sacrum avec la longueur totale de la colonne, respectivement 12,5 cm et 71 cm.

Les corps rigides de la tête et des cervicales sont une proportion du paramètre Longueur_Tete. Le corps rigide tête est à 67 % et celui des cervicales est à 33 %.

La valeur des angles entre les vertèbres suit l'équation 4.3.

$$\theta = b + a * \theta_{position} \quad (4.3)$$

Avec θ l'angle entre les deux corps rigides, $\theta_{position}$ l'angle qui définit l'inclinaison de la portion de colonne dont fait partie le corps rigide, et a et b deux constantes. Les valeurs prises par les deux constantes sont listées au tableau 3-2.

Tableau 3-2 : Valeurs des angles entre les vertèbres.

Jonction	b (°)	a (sans unité)	$\theta_{position}$ (°)
Repère-S1	4,3	-1	Inclinaison_Global
S1-L5	28,4	0,47	Angle_Lombaire
L5-L4	165,7	0,27	Angle_Lombaire
L4-L3	165,8	0,25	Angle_Lombaire
L3-L2	170,2	0,19	Angle_Lombaire
L2-L1	181,2	0,185	Angle_Lombaire
L1-T12	181	0,28	Angle_Thoracique
Ti-Ti-1	176,3	-0,136	Angle_Thoracique
T1-Cervical	165	0,5	Angle_Tete
Cervical-Tête	168	0,5	Angle_Tete

3.1.2.2 Les membres

Le squelette de la jambe (figure 3-3) est contenu dans un plan parallèle au plan sagittal. Ce plan est décalé de la moitié de la dimension de la largeur des hanches (Largeur_Hanche).

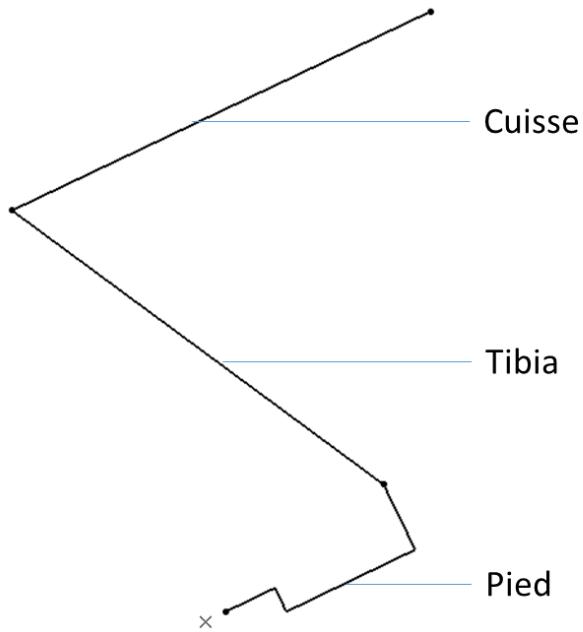


Figure 3-3 : Squelette d'une jambe.

Le deuxième membre est le membre supérieur, il s'appuie sur le corps rigide définissant l'emplacement des épaules. Ce corps rigide est défini par le paramètre Largeur_Epaule et est centré par rapport à la colonne. Il est lié à l'extrémité distale de la vertèbre T3 par un corps rigide incliné.

Le squelette du membre supérieur est composé de deux parties, le bras et l'avant-bras, et est contenu dans le plan de rotation du bras. Ce plan est défini par deux plans intermédiaires, chaque plan permet de définir un des degrés de liberté de l'épaule. Ainsi trois angles définissent la mobilité de l'épaule, Angle_Flexion_Bras, Angle_RotationLateral_Bras et Angle_Abduction_Bras. La longueur définissant le bras est Longueur_Bras. L'avant-bras est défini par un angle (Angle_Flexion_AvantBras) et une longueur (Longueur_AvantBras).

Maintenant que le squelette est entièrement défini, l'étape suivante est la définition de la surface modélisant le volume du mannequin.

3.1.3 Modèle surfacique des jambes

Dans toute cette partie, sauf indication contraire, chaque profil circulaire est lié à un paramètre de type rayon et chaque profil elliptique est lié à deux paramètres de type longueur correspondant à la longueur petit axe et grand axe. Ces paramètres sont notés `Membre_Ei_c`, avec « `Membre` » le nom du membre, `i` le numéro du profil et `c` le type de paramètre (`r` : rayon, `a` : grand axe, `b` : petit axe). Les plans des profils sont positionnés comme expliqué dans la partie 3.1.1 dans le tableau 3-1.

Le premier membre modélisé est la cuisse. Elle comporte cinq cercles et une esquisse. Les cercles sont positionnés sur des plans parallèles entre eux ayant pour normale le corps rigide de la cuisse. Les esquisses sont centrées sur le corps rigide. On notera que le cercle de l'esquisse 5 a le même rayon que celui de l'esquisse 4, et que le cercle 1 à un rayon valant 50 % celui du 2. Une surface multisection est utilisée entre les différentes esquisses (figure 3-4). Une surface multisection est une surface B-Spine passant par plusieurs profils filaires (sections) et utilise une courbe B-spline 3D qui passe par un point de chaque profil comme guide. Sur la figure 3-4, les différentes ellipses sont les sections.

Une surface de révolution complète le modèle, avec comme axe le corps rigide de la cuisse et comme profil un quart de cercle s'appuyant avec le même rayon et centre que l'esquisse 5. Le plan de l'esquisse est celui du squelette de la cuisse.

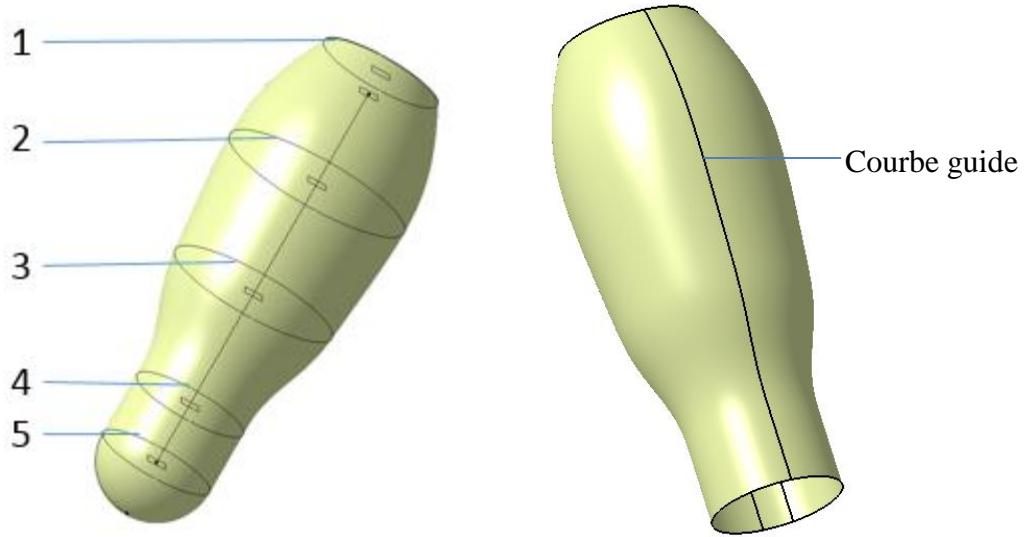


Figure 3-4 : Éléments filaires de la cuisse et surface obtenue.

Le tibia est modélisé uniquement par une surface multisection composée de quatre profils. Le premier profil est identique aux profils 4 et 5 de la cuisse. Seul le quatrième profil est une ellipse, les trois autres sont des cercles. Les deux premières esquisses sont centrées sur le corps rigide du tibia alors que les deux suivantes ont comme point d'accroche la droite de construction visible sur la figure 3-5. Cette droite est définie dans le plan de définition du squelette de la jambe et s'appuie sur les deux premières esquisses. Une surface multisection est aussi utilisée pour créer la surface du tibia.

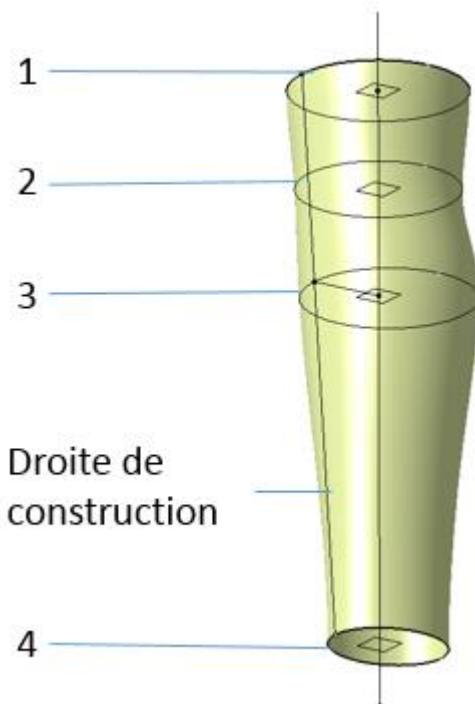


Figure 3-5 : Éléments filaires du tibia et surface obtenue.

Le pied est divisé en deux parties, l'avant du pied et la cheville (Figure 3-6 et Figure 3-7). L'avant du pied est défini par trois ellipses, les deux premières sont fixées sur la partie inférieure du squelette du pied, alors que la troisième est centrée sur la droite finissant le squelette du pied. La direction du grand grand-axe des ellipses est horizontale pour les deux dernières et verticale pour la première.

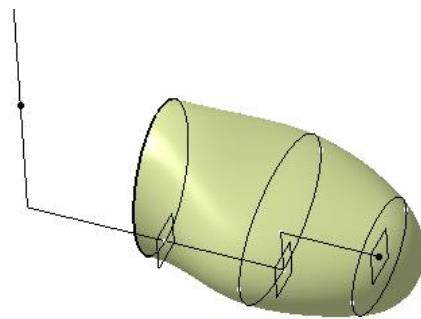


Figure 3-6 : Éléments filaires de l'avant du pied et surface obtenue.

La cheville visible sur la figure 3-7 repose sur la surface du pied et celle du tibia. Une extrusion est réalisée à partir des deux surfaces et sont ensuite fusionnées. Le résultat obtenu est coupé au niveau de l'intersection des deux extrusions par un plan dont la normale est la médiane des deux corps rigides. Une multisection est réalisée avec le contour obtenu par la découpe avec une ellipse décalé vers le talon. Enfin, une surface de fermeture est utilisée pour réaliser le talon.

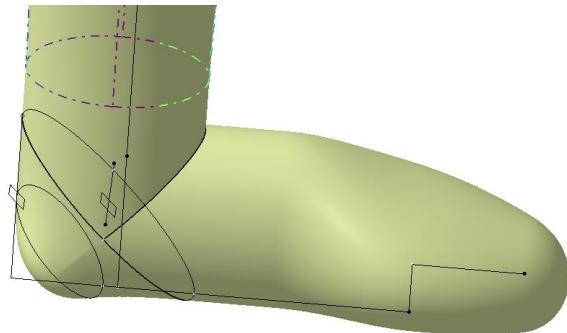


Figure 3-7 : Éléments filaires de la cheville et surface obtenue.

Un assemblage des différentes surfaces créées précédemment permet d'obtenir la surface du membre inférieur, le même travail doit être fait pour le deuxième membre inférieur. Des congés d'arêtes permettent de supprimer les discontinuités en tangence de la surface, notamment la jonction du genou.

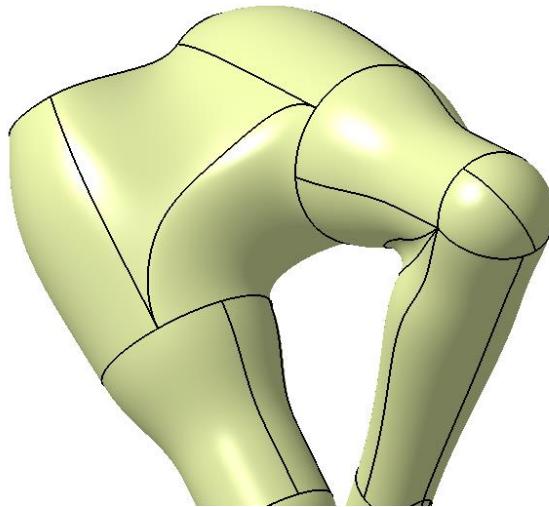


Figure 3-8 : Interface entre les deux membres inférieurs.

La création de l’interface entre les membres inférieurs (figure 3-8) est faite entre la première esquisse de la cuisse et la troisième. Les deux membres inférieurs sont coupés au niveau de leurs troisièmes profils puis la partie supérieure obtenue est recoupée par le plan de définition de la cuisse correspondant. Deux parties semblables à des demi-cylindres sont obtenues, une surface multisection liant les bords supérieurs et une autre pour les bords inférieurs sont créées. Une découpe par une spline 3D du bord au niveau de l’esquisse 3 des cuisses permet de réaliser une surface de fermeture pour obtenir le résultat escompté.

3.1.4 Modélisation du Torse

Le torse est la partie la plus complexe du mannequin. Il est composé de trois parties : le tronc, la tête avec le cou et les épaules.

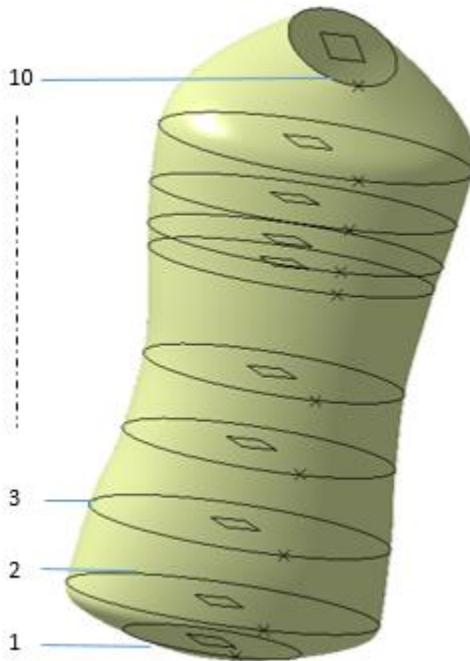


Figure 3-9 : Éléments filaires du tronc et surface obtenue.

Le tronc visible sur la figure 3-9 est une surface multisection constituée de dix ellipses réparties le long du squelette de la colonne vertébrale. En position droite de la colonne vertébrale (angles thoraciques et lombaires et inclinaison globale à 0° ainsi que), les neuf premières ellipses sont parallèles au plan transverse et la dixième est perpendiculaire au corps rigide de la colonne vertébrale qu'il coupe. Chaque plan coupe la colonne vertébrale à une fraction donnée d'une des vertèbres. Les emplacements de ces plans sont répertoriés dans le tableau 3-3. Dans deux cas, une fraction négative est nécessaire pour modéliser le bas du tronc se trouvant en dessous du sacrum.

Tableau 3-3 : Emplacement des plans de définitions des esquisses du tronc.

Numéro du plan	Emplacement	Fraction (%)
1	S1	-0,4
2	S1	-0,8
3	S1	0,3
4	L5	0,4
5	L3	0,8
6	T11	0,4
7	T10	0,2
8	T9	0,8
9	T6	0,1
10	T1	0

Ensuite, les esquisses sont créées sur les plans, trois longueurs par ellipse les dimensionnent et positionnent. Elles sont visibles sur figure 3-10. Le paramètre a correspond au grand-axe de l'ellipse, b au petit-axe et e est la distance du centre de l'ellipse à la colonne vertébrale.

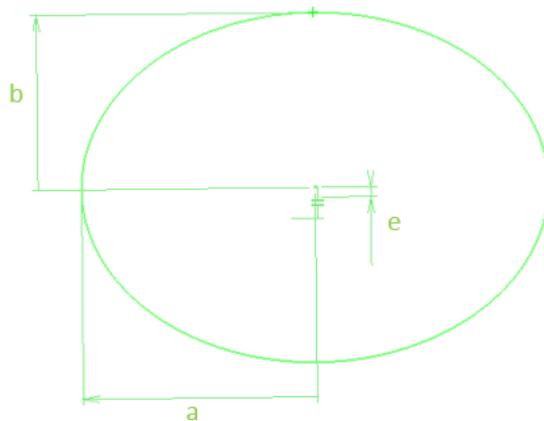


Figure 3-10 : Création d'une ellipse du tronc

La création du tronc a pour but de modéliser deux formes simplificatrices du tronc par des équations simples, ces deux formes sont celles du contour dans le plan sagittal et celle du contour dans le plan frontal (visible à la figure 3-11). Chaque forme est caractérisée par deux paramètres, une moyenne et une amplitude permettant de modéliser les formes sinusoïdales.

Pour les équations, la notation utilisée est la suivante :

- Tronc_Amplitude_a noté A1
- Tronc_Moyenne_a noté M1
- Tronc_Amplitude_b noté A
- Tronc_Moyenne_b noté M

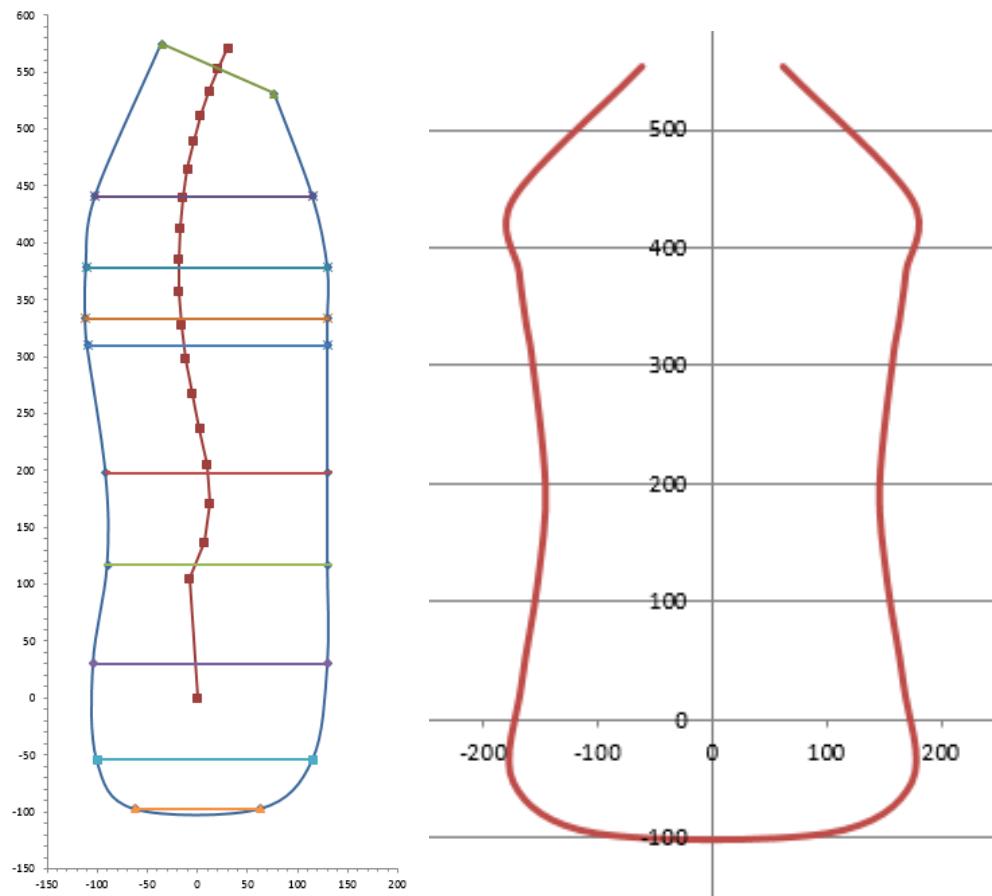


Figure 3-11 : Exemple de forme du tronc dans le plan sagittal et dans le plan frontal (dimension en mm).

Les deux premiers paramètres permettent de définir la longueur a selon l'équation (3.4). Les quatre autres permettent de définir les longueurs b et e selon les équations (3.5) et (3.6) respectivement. Dans les trois équations, x est le numéro de l'esquisse et y(x) est un ratio défini au tableau 3-4.

Dans les équations, deux paramètres « Mesures » et « Mesures2 » sont utilisés. La figure 3-12 montre comment sont mesurés ces deux paramètres. Ces deux mesures permettent de déterminer l'intersection du plan numéro x avec la colonne vertébrale. La mesure est faite sur une copie des esquisses de la colonne vertébrale restant en position verticale. Les deux paramètres φ et T sont une proportion de la longueur du torse, 8,7 % pour le déphasage et 69 % pour la période.

$$a(x) = A1 * \cos(y(x) * 2\pi) + M1 \quad (4.4)$$

$$b(x) = M + A * (1 - 1/2 * \sin((\text{Mesure}(x) - \varphi) * 2\pi/T)) \quad (4.5)$$

$$e(x) = A * (1 + 1/2 * \sin((\text{Mesure}(x) - \varphi) * 2\pi/T)) - \text{Mesure2}(x) \quad (4.6)$$

Tableau 3-4 : Valeurs du paramètre y(x).

Numéro de l'esquisse : x	y(x)
1	-
2	0,11008
3	0,17
4	0,34
5	0,5
6	0,73
7	0,78
8	0,86
9	1
10	-

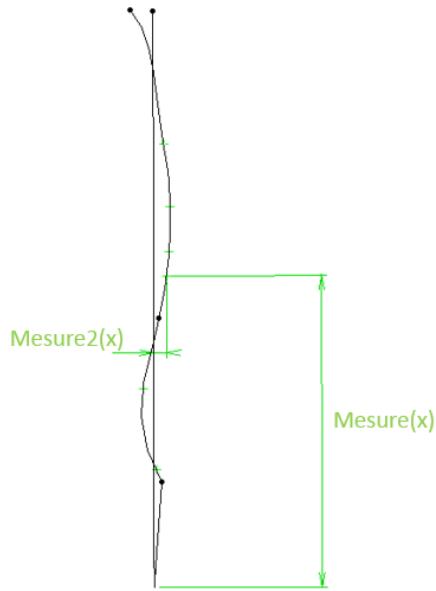


Figure 3-12 : Prise de mesure sur les éléments de construction.

La forme recherchée pour le tronc est montrée à la figure 3-11. Dans le plan sagittal, une droite est recherchée pour l'avant du tronc alors que pour l'arrière, une période de sinus est modélisée. Trois paramètres sont nécessaires pour définir la fonction sinus :

- Son amplitude A
- Sa période T
- Son déphasage φ

En plus des paramètres définissant la fonction sinus, un paramètre M représentant le décalage de la fonction sinus par rapport à l'axe \vec{z} est ajouté. Ainsi la distance de l'arrière du tronc par rapport à l'axe \vec{z} notée $x_1(z)$ est donnée par l'équation 4.9. La distance de l'avant par rapport à l'axe \vec{z} notée $x_2(z)$ est donnée par l'équation 4.10.

$$x_1(z) = -M + A * \sin((z - \varphi) * 2\pi/T) \quad (4.7)$$

$$x_2(z) = M + 2 * A \quad (4.8)$$

La distance x_1 est valable pour les ellipses 2 à 9, alors que x_2 n'est valable que pour les ellipses 3 à 8. Ainsi pour les ellipses 3 à 8, on a :

$$\frac{x_2(z) - x_1(z)}{2} = b(x) \quad (4.9)$$

car

$$Mesure(x) = z \quad (4.10)$$

Grâce aux deux distances x_1 et x_2 , l'équation de distance du centre avec l'axe \vec{z} est facilement obtenue :

$$d = \frac{x_2(z) + x_1(z)}{2} \quad (4.11)$$

Ainsi comme la distance de l'axe \vec{z} avec la colonne est mesurée et correspond à $Mesure2(x)$, l'excentricité de l'ellipse donne l'équation suivante :

$$\frac{x_2(z) + x_1(z)}{2} - Mesure2(x) = e(x) \quad (4.12)$$

En remplaçant x_1 et x_2 dans les équations 4.11 et 4.14, les équations 4.7 et 4.8 sont ainsi obtenues.

Quatre ellipses ne sont pas paramétrées. Les ellipses 2 et 9 ont le même x_1 alors que le x_2 est légèrement modifié :

$$x_2(z) = M + A \quad (4.13)$$

Pour l'esquisse 1, elle est centrée sur l'axe \vec{z} , donc l'excentricité e devient nulle et la valeur de la moitié du petit axe devient :

$$b(1) = 0,8 * M \quad (4.14)$$

L'esquisse 10 est un cercle et est liée à un paramètre `Tronc_E10_r` qui correspond à une mesure du rayon du cou.

Pour la forme du tronc dans le plan frontal, une forme de sinus est aussi voulue, d'où l'équation 4.6. Aucun décalage n'est nécessaire dans ce cas.

Une surface multisection est faite sur l'ensemble de ces profils pour modéliser la surface du tronc. Une surface de remplissage est aussi faite sur l'esquisse 1 avec comme tangence la surface précédente. Un assemblage donne la surface finale du tronc.

L'épaule est l'articulation la plus complexe à modéliser due à ses trois degrés de liberté. Trois ellipses la composent. Les deux premières (ellipses à gauche de la Figure 3-13) ont leur grand axe parallèle à la droite verticale définie lors de la modélisation du squelette des épaules. La troisième a son grand axe perpendiculaire aux deux autres ellipses.

La valeur du petit axe de l'esquisse 1 correspond à 60 % du petit axe de la neuvième ellipse du tronc, et son grand axe est à 22 % du paramètre Longueur_Tronc. Son centre est à mi-distance entre le corps rigide de l'épaule et la colonne vertébrale. Le plan de l'esquisse 1 est celui de la colonne vertébrale.

Les dimensions de l'esquisse 2 sont 55 % du grand axe de l'esquisse 1 et 83 % du petit axe. L'esquisse 3 a son grand-axe égal au petit-axe de l'esquisse 2 et son petit-axe 72 % du grand-axe de l'esquisse 3. Le plan de l'esquisse 2 et 3 est un plan décalé de l'esquisse 1, le plan 3 est coïncident avec l'épaule, le plan 2 est à 75 % de la distance entre le plan 1 et 3. Les différents ratios précédents ont été définis pour permettre d'obtenir des surfaces fonctionnelles et assemblables entre elles.

Une multisection est utilisée entre les trois ellipses et une surface de révolution utilisant l'ellipse 3 comme profil permet de définir l'épaule, une symétrie par le plan de la colonne permet de définir l'autre épaule.

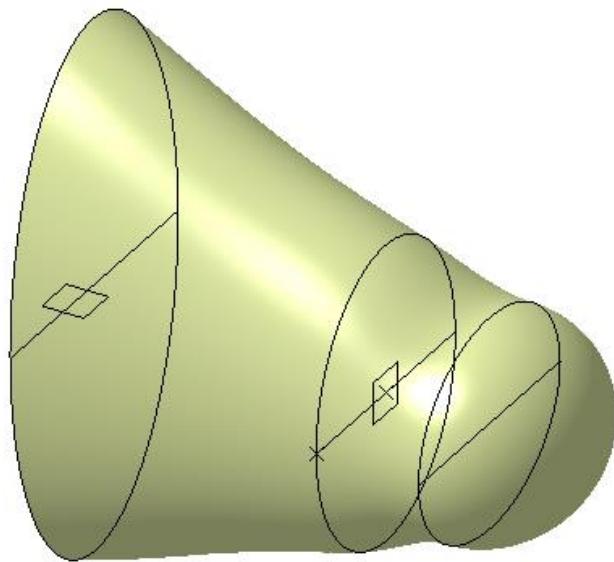


Figure 3-13 : Éléments filaires de l'épaule et surface obtenue.

La dernière partie de la modélisation du tronc est la tête, qui est composée d'une demi-ellipse. Le cou est une extraction de la dernière ellipse du tronc dans la direction de la dernière vertèbre avec comme élément limitant la fin du corps rigide de la vertèbre. La deuxième partie du cou est une extraction utilisant la limite supérieure de l'extraction précédente, comme direction le corps rigide de la tête et s'arrêtant à 60 % de la longueur de ce corps rigide.

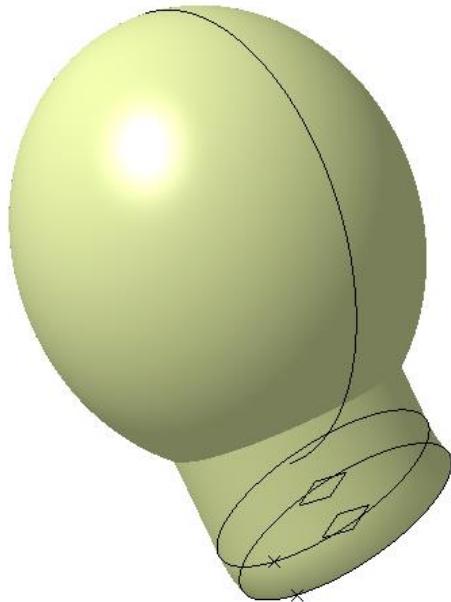


Figure 3-14 : Éléments filaires de la tête et surface obtenue.

Les dimensions de l'ellipse de la tête sont 140 % de la longueur du corps rigide de la tête pour le grand-axe, alors que le petit-axe est lié à un paramètre mesuré. Les différentes parties sont ensuite assemblées pour former la surface de la tête.

Les trois parties du tronc sont ensuite assemblées et un congé est réalisé sur les zones ayant une arête vive. La surface de modélisation du tronc est ainsi obtenue.

3.1.5 Modélisation des membres supérieurs

Le membre supérieur entier est modélisé en deux parties : le bras, qui est la partie entre l'épaule et le coude et l'avant-bras qui est tout ce qui se trouve après le coude incluant la main.

Le bras est composé de trois ellipses et d'une esquisse nécessaire à créer la jonction entre les deux parties du membre. La première ellipse est liée aux angles du bras pour permettre qu'elle coïncide avec la surface de jonction de l'épaule. Les deux autres ellipses ont leur grand axe compris dans le plan de définition du squelette du bras et sont centrées sur le corps rigide du bras. Une quatrième ellipse identique à l'ellipse 3 est aussi présente pour créer la jonction avec l'avant-bras par une révolution autour de son petit-axe. Une multisection est utilisée sur les trois premières ellipses et est assemblée avec la surface de révolution pour obtenir la surface de la figure 3-15.

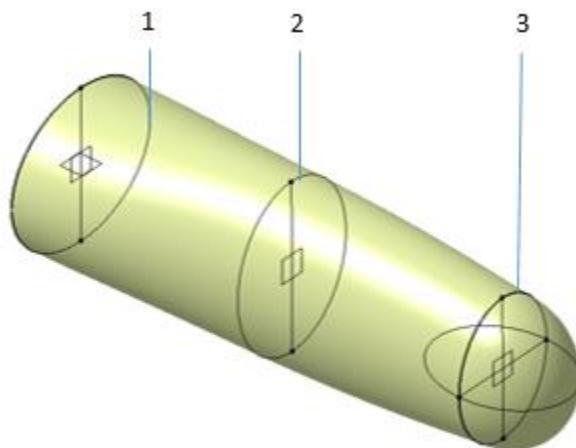


Figure 3-15 : Éléments filaires du bras et surface obtenue.

La surface de l'avant-bras est constituée de trois ellipses et d'un demi-cercle comme éléments filaires permettant de créer la surface. La première ellipse est identique à la dernière ellipse du membre précédent permettant de faire la jonction entre les deux surfaces facilement. La deuxième et la troisième ellipses ont leur grand axe compris dans le plan de définition du squelette du bras et sont centrées sur le corps rigide de l'avant-bras. L'esquisse du demi-cercle est comprise dans le plan de définition du squelette du bras. Les deux extrémités et le centre du demi-cercle coïncident avec le corps rigide du squelette de l'avant-bras.

Une révolution à 180° est faite du demi-cercle avec le corps rigide de l'avant-bras comme axe de rotation. Une multisection est faite avec les trois ellipses. Les deux surfaces sont assemblées et seule la surface extérieure est gardée. Un congé de 16 mm entre les deux surfaces est fait. Le résultat obtenu est visible sur la figure 3-16.

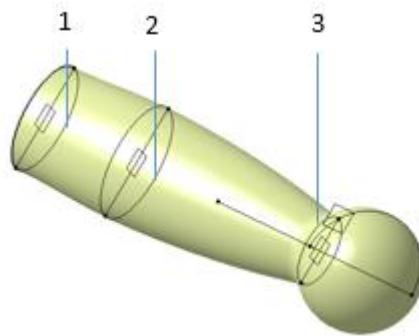


Figure 3-16 : Éléments filaires de l'avant-bras et surface obtenue.

Les deux surfaces sont assemblées et découpées pour ne garder que la partie extérieure. Une symétrie de cette surface avec le plan de définition de la colonne vertébrale (plan sagittal) permet d'obtenir le deuxième membre supérieur. Le résultat est visible sur la figure 3-17.

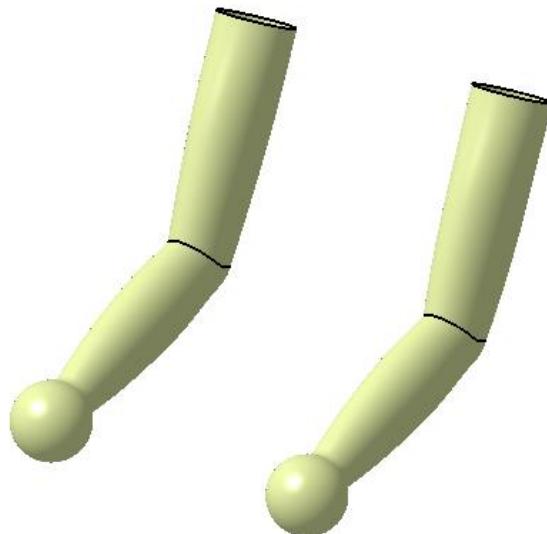


Figure 3-17 : Surface finale des deux membres supérieurs.

3.1.6 Assemblage des surfaces

La dernière étape du modèle est l'assemblage des surfaces. Pour cela, une découpe des surfaces est faite pour ne garder que la partie extérieure des surfaces, suivie d'un assemblage pour n'avoir qu'une seule et même surface de modélisation. Deux congés sont nécessaires pour assurer une continuité en tangence de toute la surface, un de 20 mm au niveau de la jonction entre les membres inférieurs et le tronc et un de 10 mm entre les bras et les épaules et entre le bras et l'avant-bras. Le résultat est visible à la figure 3-18, pour deux positions du même modèle.

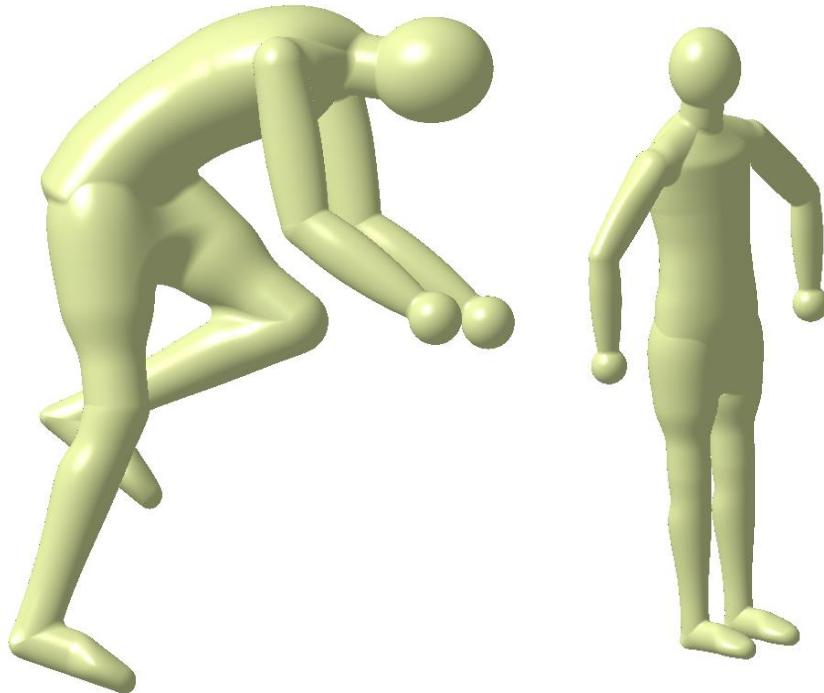


Figure 3-18 : Modèle surfacique obtenu dans une position sur un vélo et une position debout.

3.2 Le système de mesure

Dans cette partie, la méthode d'acquisition des données anthropométriques va être développée grâce au système Kinect (Microsoft, Redmond, États-Unis). Ensuite, la méthode de traitement des acquisitions sera présentée pour une utilisation par le modèle CAD des mesures.

3.2.1 La Kinect

Dans le but de paramétriser le modèle construit dans la partie 3.1, les mesures suivantes sur le sujet à modéliser doivent être réalisées :

- Longueurs du squelette
- Largeur des membres à des positions prédéterminées du membre dans deux directions perpendiculaires

Pour réaliser ces mesures : un des systèmes de Microsoft est choisi : la Kinect. Ce système intègre une détection en profondeur qui permet d'obtenir le contour du sujet en premier plan. Suite à cette détection, le logiciel intégré détecte l'emplacement des articulations du sujet (cf. Figure 3-19).

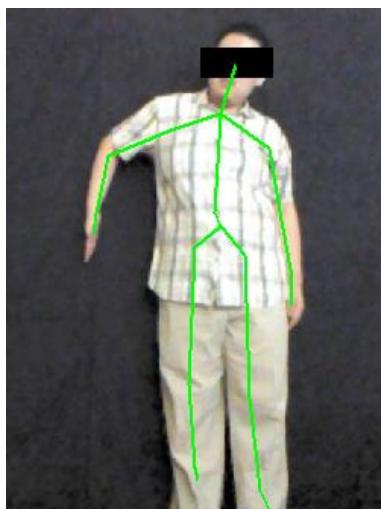


Figure 3-19 : Détection automatique du squelette avec la Kinect.

La largeur des membres est obtenue grâce à une adaptation de la méthode de Jensen (Jensen, 1978). Pour cela, deux acquisitions du sujet sont nécessaires ; une de face et une de profil. Lors des

deux acquisitions, le sujet doit se trouver à la même profondeur pour garder des dimensions apparentes égales. En effet, un sujet plus éloigné par rapport à une première acquisition sera plus petit sur l'image acquise.

Les images et les données des fichiers textes ne sont pas en mètres, mais en pixel. Un facteur de conversion est donc nécessaire. Pour cela, l'utilisation d'un calibre de forme circulaire a été choisie pour pouvoir déterminer ce facteur. Le calibre, de diamètre 12cm, doit être placé à la même profondeur que le sujet. La mesure du diamètre du calibre en pixel permet d'obtenir le facteur de conversion.

3.2.2 Traitement des acquisitions

Le traitement des données se déroule en trois étapes, le traitement de l'image du calibre, le calcul des longueurs des corps rigides, et le traitement des images pour l'obtention des largeurs des membres. Les trois étapes se font grâce au logiciel Matlab (MathWorks, Natick, États-Unis) dont les codes se trouvent à l'annexe C.

Les différentes étapes de traitements des images (Figure 3-20) se déroulent comme suivant. Les différentes segmentations sont réalisées à l'aide du filtre de Sobel, qui utilise le gradient de l'image transformée en niveau de gris. Ensuite, une dilatation de la segmentation est réalisée afin d'avoir une continuité des lignes de segmentation pour permettre un remplissage de la segmentation. Enfin, un lissage de la détection du corps permet de supprimer le bruit l'entourant.

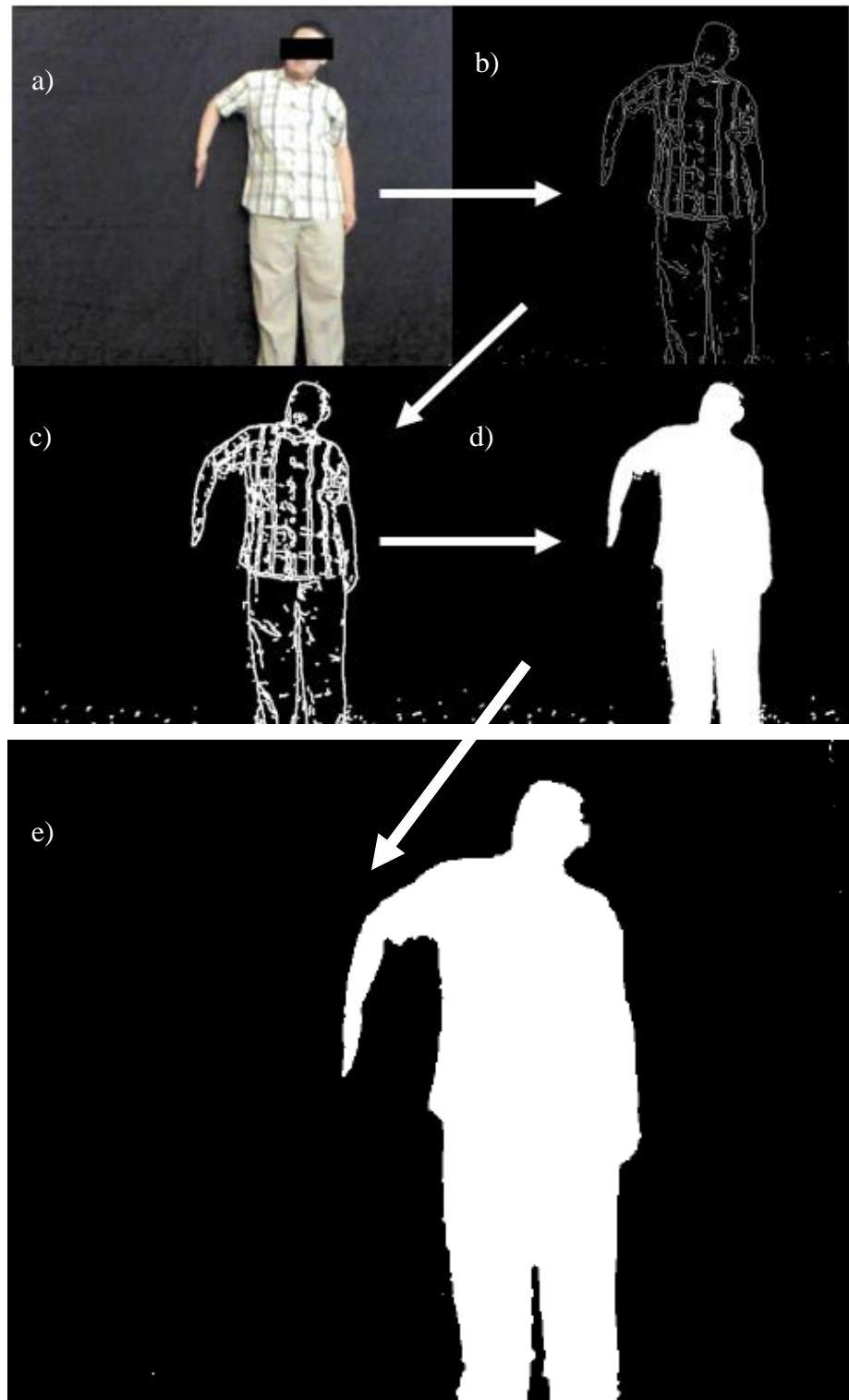


Figure 3-20 : Les différentes étapes de la segmentation : a) Image d'origine, b) Segmentation, c) Épaississement des contours de segmentation, d) Remplissage de la segmentation, e) Lissage du remplissage pour supprimer le bruit.

3.2.2.1 Le calibre

Une segmentation de l'acquisition du calibre est nécessaire pour obtenir le rayon du calibre en pixel (cf. Figure 3-21). Le résultat de la calibration est donné par le diamètre du calibre divisé par la moyenne d'une mesure selon la verticale et une autre selon l'horizontale de la taille maximum de la segmentation.

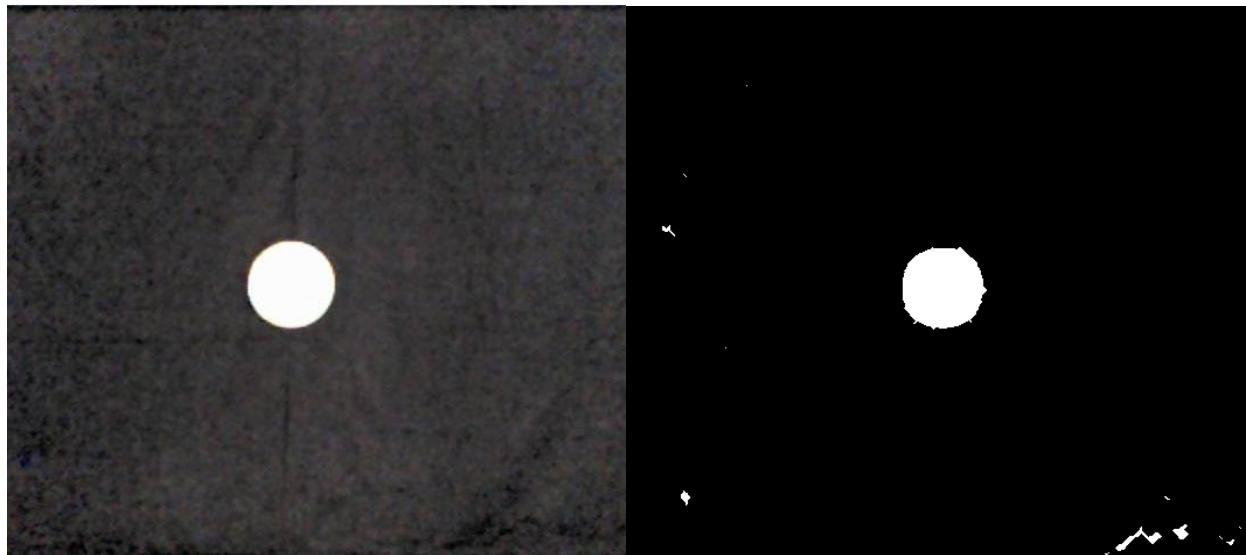


Figure 3-21 : Image du calibre avant et après segmentation.

3.2.2.2 Les corps rigides

Vingt coordonnées de points sont enregistrées dans le fichier texte de transfert entre le programme Kinect et le programme Matlab (Figure 3-22). Le positionnement des articulations étant imparfait, une action manuelle pour les repositionner est nécessaire (Figure 3-23) :

- L'ensemble des points 3, 4, 13 et 17 se trouvent trop haut sur le tronc
- Les extrémités du corps : points 1, 8, 12, 16 et 20 qui ne se trouvent pas sur l'extrémité maximum du sujet
- Recentrage sur le corps du reste des points si nécessaire

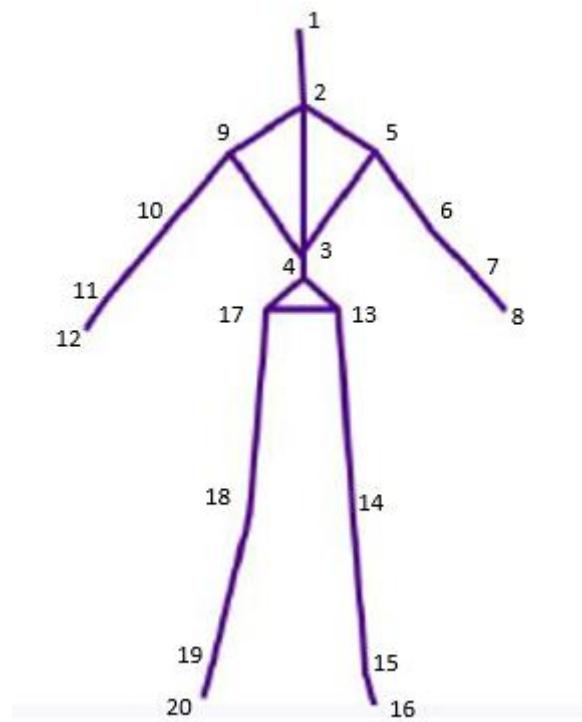


Figure 3-22 : Numérotation des articulations sur le squelette.

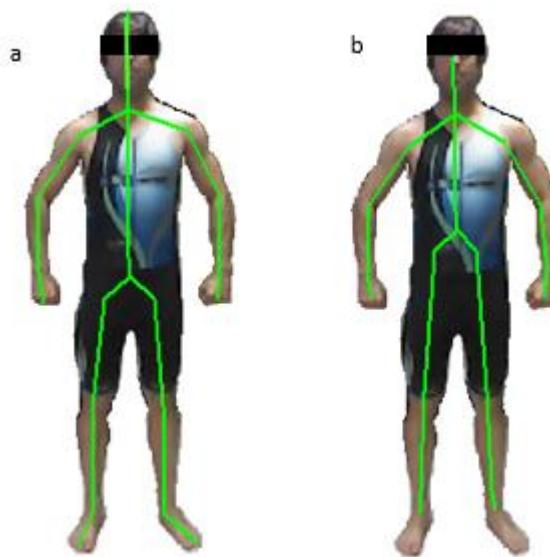


Figure 3-23 : Squelette (a) après repositionnement et b) avant).

3.2.2.3 Les mesures de largeur

Les mesures des grands et petits axes des ellipses doivent être faites perpendiculairement au segment du membre mesuré. Chacune des acquisitions permet d'obtenir une des deux dimensions sur chaque point de mesure. Ces différents emplacements sont donnés dans le tableau 3-5, ils sont différents de ceux des ellipses, car certaines ellipses ne sont pas mesurées.

Tableau 3-5 : Emplacements des mesures des ellipses sur les membres.

Membre	Pourcentages du membre		
Bras	50	100	
Avant-Bras	33,33	100	
Main	50		
Cuisse	30	60	90
Tibia	15	33,33	90
Tronc	50	100	

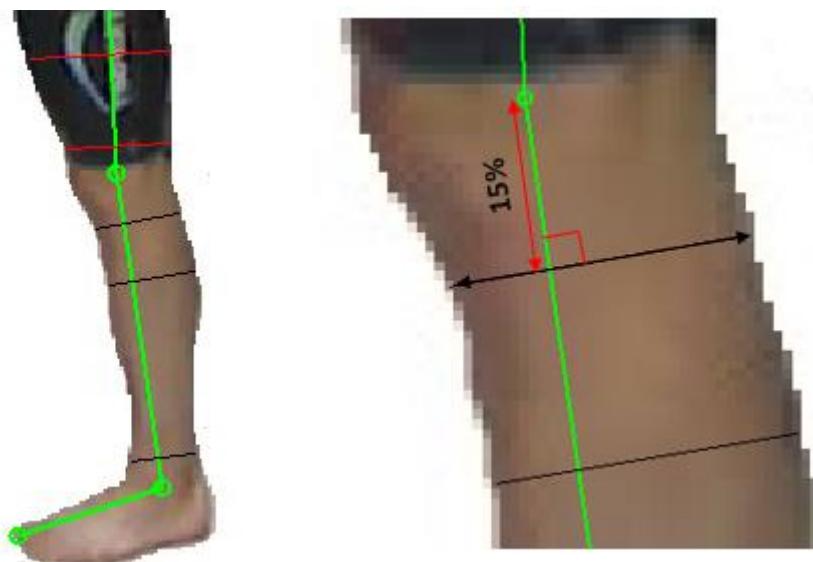


Figure 3-24 : Orthogonalité des mesures des ellipses sur les membres.

Ainsi en combinant une mesure de l'acquisition de face et une de profil, les dimensions du grand-axe et du petit-axe sont obtenues et définissent entièrement les ellipses des membres du modèle.

Sur l'acquisition des paramètres du tronc, les mesures ne sont pas directement transmises en tant que paramètres, mais un calcul préliminaire permet d'obtenir les paramètres nécessaires au modèle. Les équations permettant de les calculer sont les suivantes :

$$A1 = \frac{M1_0 - M1_{0,5}}{4} \quad (4.15)$$

$$M1 = \frac{M1_0 + M_{0,5}}{4} \quad (4.16)$$

$$A = \frac{M_0 - M_{0,5}}{2} \quad (4.17)$$

$$M = \frac{M_0 + M_{0,5} - 4 * A}{4} \quad (4.18)$$

Les paramètres $M1_x$ et M_x sont les mesures de largeurs du tronc, les $M1_x$ sont les mesures du tronc dans le plan frontal alors que le deuxième type est les mesures dans le plan sagittal. L'indice x correspond à la proportion de la longueur où est effectuée la mesure.

3.3 Le maillage

3.3.1 Creation de la geomtrie

Le maillage est créé à partir du logiciel Ansys (Ansys, inc., Canonsburg, États-Unis) et plus particulièrement son module ICEM. Lors de l'importation du fichier de transfert au format STEP. Les axes sont les mêmes que ceux du modèle Catia.

Le domaine d'étude choisi (figure 3-25) doit être assez grand pour éviter tout effet de bord. Les dimensions du domaine correspondent à des conditions extérieures. Pour cela, le domaine a été choisi de forme pyramidale tronqué. Le domaine a une profondeur de 14,5 m, la surface d'entrée (Inlet) est un rectangle de 4 m de large et 3,5 m de haut, et la sortie (Outlet) est de 8 m de large et 6,5 m de haut.

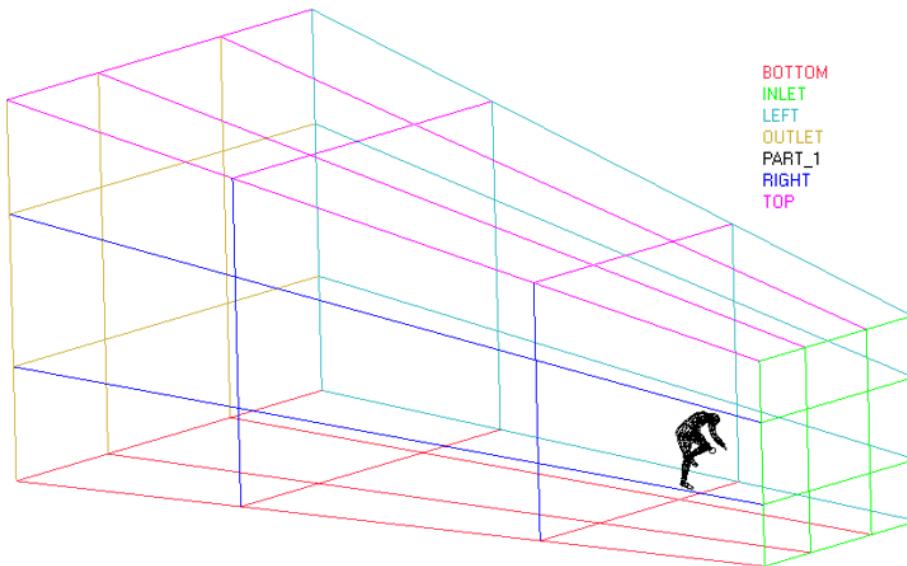


Figure 3-25 : Le domaine d'étude (Inlet : surface d'entrée du fluide ; Outlet, surface de sortie du fluide ; Bottom : Surface inférieure modélisant le sol ; Top : Surface supérieure ; Right et Left : Surface à droite et à gauche du modèle; Part_1 : Surface importée du cycliste).

Les dimensions du domaine ont été modifiées afin d'observer si la taille du domaine avait une influence sur les résultats. Les résultats obtenus sur les coefficients de traînée sont très proches

; 0,6047 pour le résultat minimum et 0,6163 pour le maximum. Ainsi, la taille du domaine ayant peu d'influence avec les tailles utilisées, la plus petite taille de domaine testée a donc été choisie.

Un dernier point doit être défini pour indiquer au logiciel l'emplacement du fluide et ainsi permettre de mailler la bonne zone. Ce point a pour coordonnées [-3.95 1 1.75] et se nomme « Fluid ».

3.3.2 La création du maillage

Deux types de méthode de construction de maillage sont possibles ; le maillage structuré et le maillage non structuré.

Le maillage structuré demande une génération manuelle pour chaque nouvelle géométrie, ce qui implique un temps de construction long. Le maillage non structuré est automatique et donc il est plus rapide à générer. D'un autre côté, la qualité du maillage est importante pour la convergence rapide de la simulation. En général un maillage structuré est beaucoup plus rapide et stable lors de la simulation par rapport au maillage non structuré étant donné qu'une bonne configuration du maillage doit en théorie donner un meilleur maillage, contrairement aux algorithmes de remplissage automatique. Le choix s'est porté sur le maillage non-structuré étant donnée la complexité du modèle géométrique et surtout le besoin d'automatiser le processus.

Le tableau 3-6 résume les avantages et inconvénients des trois types d'éléments proposés par le logiciel ICEM pour les maillages non-structurés. Le choix a été fait d'utiliser les éléments tétraédriques pour leurs adaptations plus grandes lors des géométries complexes.

Tableau 3-6 : Avantages et inconvénients des types d'élément.

Types d'élément	Avantages	Inconvénients
Hexaédrique	Minimise la diffusion numérique	-Grand nombres de cellules pour des géométries complexes
Tétraédrique	Nombres réduits de cellules pour des géométries complexes	-Temps de calcul long -sensibles à l'inclinaison asymétrique
Polyèdre	Grande précision	-Utilisation conséquente de l'espace mémoire -Grand nombre d'opérations par cellule

Le logiciel ICEM propose trois algorithmes de remplissage pour des éléments tétraédriques :

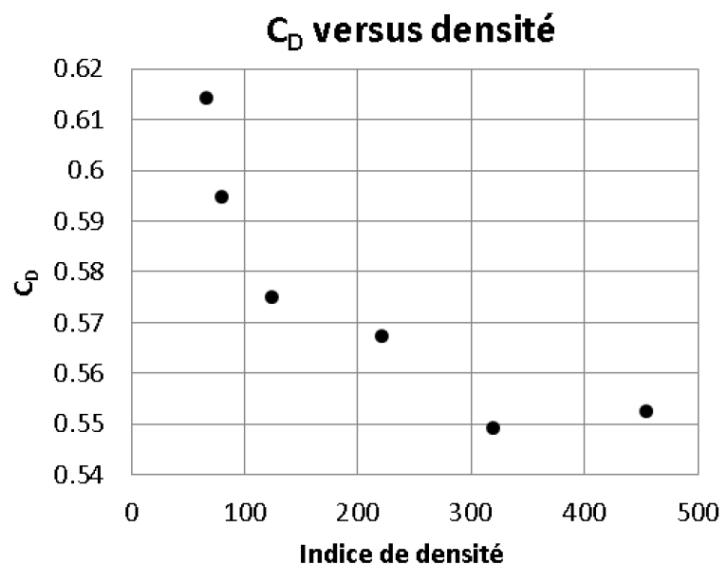
- *Robust (Octree)*
- *Quick (Delaunay)*
- *Smooth (Advancing Front)*

L'algorithme *Smooth* offre une bonne qualité de maillage, mais est sensible à la qualité de la géométrie importée, ce qui peut être source de problèmes lors du maillage de la géométrie obtenue par de nouvelles mesures. L'algorithme *Quick* a été source de nombreuses erreurs de maillage de façon récurrente. L'algorithme *Robust* offre un maillage exempt d'erreur et est celui qui a été utilisé pour la simulation.

La taille du maillage est un élément important du maillage étant donné que son choix détermine le temps de calcul ainsi que la précision du calcul. Plus le maillage est fin, plus le résultat sera précis, mais plus le calcul sera long. La densité de maillage est souvent testée par une étude de convergence du maillage. Ce test consiste à réaliser plusieurs simulations avec des densités différentes et d'observer les résultats obtenus. Le tableau 3-7 regroupe les différents couples de paramètres testés.

Tableau 3-7 : Différentes densités de maillage testées.

Maillage	Taille maximum	Taille minimum	Densité
1	0,6	0,025	67
2	0,5	0,025	80
3	0,4	0,02	125
4	0,3	0,015	222
5	0,25	0,0125	320
6	0,2	0,01	455

Figure 3-26 : C_D en fonction de l'indice de densité du maillage.

La figure 3-26 donne le graphique de convergence des maillages en traçant le coefficient de pression en fonction de l'indice de densité du maillage. Une convergence du résultat est visible pour des indices de densité supérieurs à 300. Ainsi le maillage 5 avec une taille maximale de maille de 0,25 et une taille minimale de maille de 0,0125 a donc été choisi.

La couche limite est une couche du fluide située en périphérie de l'objet étudié où les effets de viscosité sont les plus importants. Ainsi une attention particulière est nécessaire dans cette zone. Le solveur ANSYS Fluent propose deux lois de paroi :

- *Standard Wall Functions*
- *Enhanced Wall Treatment*

La première loi utilise une approximation pour résoudre les équations de la sous-couche visqueuse, alors que la deuxième les résout. Ainsi dans le premier cas aucun point de maillage n'est nécessaire dans la sous-couche alors que la deuxième loi de paroi nécessite des points de maillage. Le choix entre les deux modèles dépend d'un paramètre adimensionnel y^+ donné par l'équation 4.19 ; y étant l'ordonnée de la première maille, ν la viscosité cinématique et u_T la vitesse de frottement. Ce paramètre caractérise la taille de la première maille de la couche limite par rapport à la vitesse du fluide loin de la paroi. Ainsi le y^+ doit être plus faible pour la deuxième loi de paroi, généralement un y^+ compris entre 30 et 300 est nécessaire pour la loi de paroi standard alors que pour la deuxième un y^+ inférieur à 5 est préférable. Ce paramètre n'est vérifiable qu'une fois la simulation réalisée, car le y^+ dépend des vitesses aux abords de la couche limite et est difficilement calculable avant la simulation.

$$y^+ = \frac{u_T y}{\nu} \quad (4.19)$$

Le maillage de la couche limite est structuré contrairement au reste du domaine. Il est constitué de couches de mailles successives étant des prismes triangulaires. Trois paramètres sont nécessaires pour déterminer le maillage de la couche limite :

- Taille de la première maille ($5 \cdot 10^{-5}$)
- Nombre de couches de mailles structurées (20)
- Taux de croissance des mailles (1,2)

Une vérification de la qualité du maillage est faite ainsi qu'un lissage du maillage pour améliorer la qualité. Le maillage a une qualité supérieure à 0,25, seuls 5 éléments sur 1 553 613 éléments sont compris entre 0,2 et 0,25. La plus grosse partie des éléments a une qualité entre 0,9 et 0,95 (57,5 %).

La qualité du maillage est un coefficient compris entre 0 et 1, où 1 est la qualité maximum et 0 une qualité minimum. Un maillage de haute qualité permet une convergence plus rapide et une meilleure qualité des résultats. La qualité se base sur différentes dimensions des mailles, ainsi une maille trop déformée, trop allongée dans une dimension par rapport aux autres aura une qualité moindre.

Avec les différents paramètres de maillage choisis, la figure 3-27-a donne un aperçu du maillage surfacique du modèle au niveau de la tête, alors que la figure 3-27-b est le maillage du domaine d'étude. La figure 3-28 permet de visualiser la forme de la couche limite et ses différentes couches.

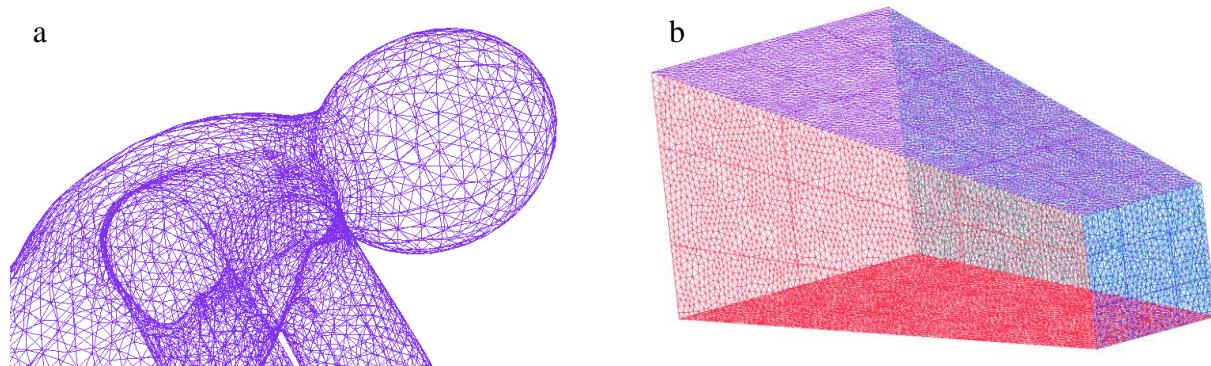


Figure 3-27 : Maillage surfacique du modèle (a) et du domaine (b).

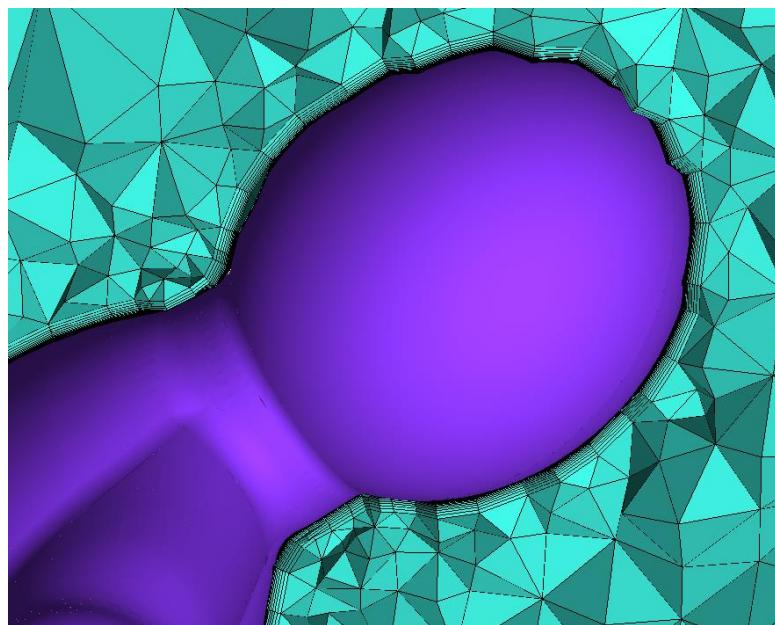


Figure 3-28 : Maillage de la couche limite.

La figure 3-29 donne un aperçu global du maillage du domaine d'étude grâce à une coupe dans le plan XZ de ce maillage.

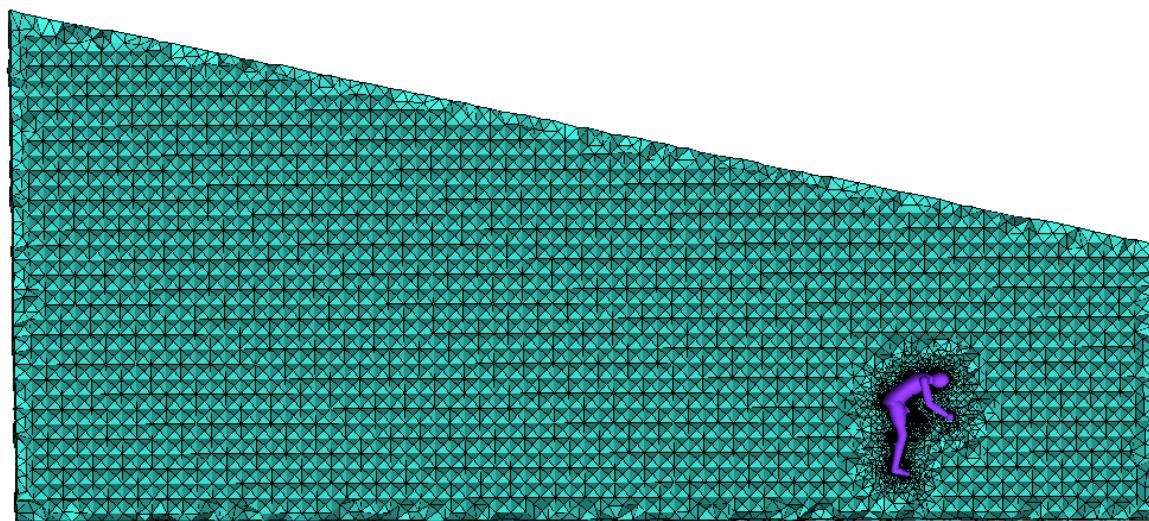


Figure 3-29 : Coupe du maillage du domaine.

3.4 La Simulation

Le maillage obtenu dans la partie précédente est exporté dans le logiciel ANSYS Fluent pour y effectuer la simulation de l'écoulement. Dans un premier temps, le modèle utilisé pour la simulation sera expliqué suivi par le choix des conditions aux limites et enfin, finir sur l'extraction des résultats.

3.4.1 Le modèle du fluide

La simulation passe par un choix de la modélisation de la turbulence. Plusieurs approches numériques sont possibles. Celle choisie se trouve être une des plus courantes nommée RANS (Reynolds Average Navier-stokes) et plus particulièrement le modèle k- ε .

Le fluide modélisé est de l'air. Les deux paramètres à définir ont donc les valeurs suivantes :

Masse volumique $1,225 \text{ kg/m}^3$

Viscosité : $1,7894 \times 10^{-5} \text{ Pa.s}$

3.4.2 Les conditions aux limites

Quatre types de conditions aux limites ont été utilisés :

- Wall : Le fluide ne peut pas traverser la paroi et il adhère à la paroi (vitesse nulle)
- Velocity : La vitesse du fluide et sa direction sont définies au niveau de la paroi
- Outflow : Condition libre de sortie de fluide
- Symmetry : le fluide se comporte de façon symétrique de chaque côté de la paroi

Le premier type de conditions aux limites (Wall) est utilisé pour la surface du cycliste nommé Part_1 et aussi la surface inférieure du domaine noté Bottom. L'utilisation de cette condition sur la surface inférieure du domaine permet de modéliser la présence du sol.

Le deuxième type de conditions aux limites (Velocity) est utilisé pour la surface frontale du domaine nommé Inlet. Ce type de condition est généralement utilisé pour définir une zone où les conditions de vents sont connues et cette zone se trouve être le plus souvent la surface d'entrée du fluide. Le vent dans la simulation est frontal avec une vitesse relative de 12,5 m/s.

Le troisième type de conditions aux limites (Outflow) est utilisé pour la surface arrière du domaine qui correspond à la sortie du fluide, cette surface est nommée Outlet. Une condition libre est aussi possible étant donné que la surface du cycliste se trouve assez loin et donc les effets de sa présence se retrouve annulé.

Enfin, le dernier type de conditions aux limites (Symmetry) est utilisé pour les deux surfaces latérales et la surface supérieure du domaine. Le choix n'est pas optimal mais permet d'obtenir un flux d'air recherché autour du cycliste, car les surfaces ne sont pas dans la direction du flux d'air.

La figure 3-30 montre les vecteurs vitesse autour du cycliste dans un plan ZY et XZ, le flux est constant autour de ce dernier et la vitesse est de 10 m.s^{-1} , c'est la vitesse utilisée par Defraeye pour les comparaisons.

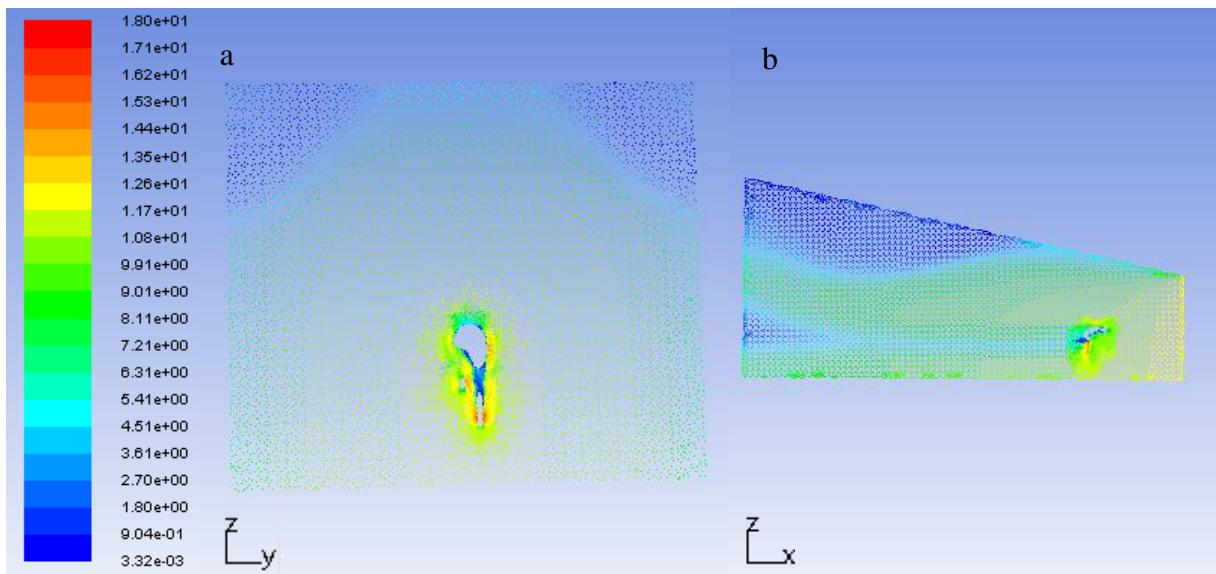


Figure 3-30 : Vecteur vitesse (m/s) dans le plan ZY (a) et dans le plan XZ (b).

3.4.3 L'extraction des résultats

Une vérification de la convergence de la simulation doit être faite, pour cela la courbe de la valeur du coefficient de trainé (C_d) en fonction de l'itération est récupérée (Figure 3-31). Cette courbe doit montrer qu'elle a une tendance à tendre vers une valeur au fur et à mesure des itérations.

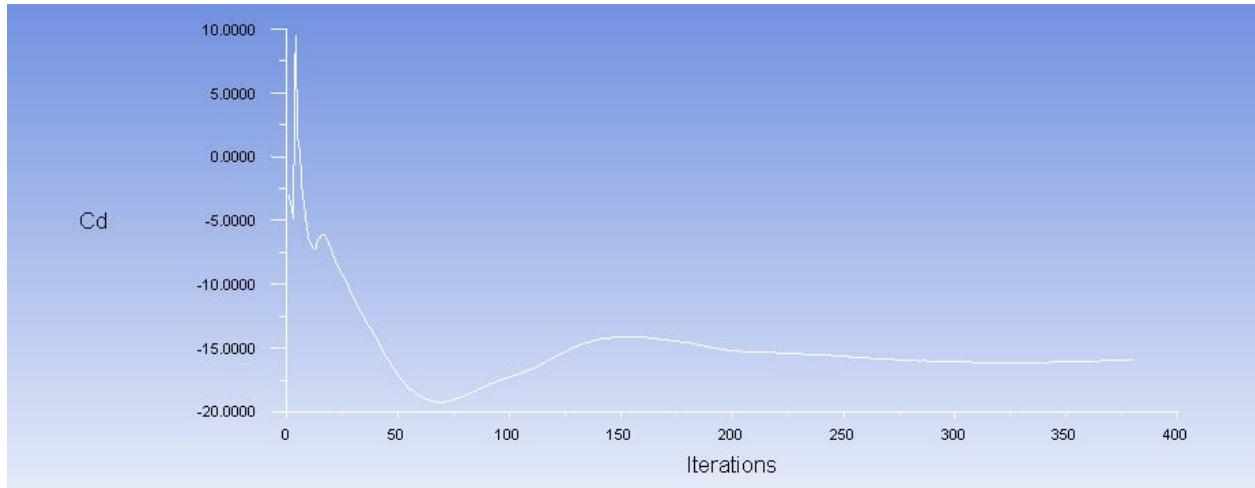


Figure 3-31 : Courbe de convergence du Cd (sans unité) en fonction des itérations.

Une fois que la simulation a convergé, l'extraction des données peut être effectuée. Deux données sont utiles : le coefficient de traînée notée Cd du cycliste, et les valeurs de $y+$ sur la surface du cycliste.

La valeur du coefficient de traînée est donnée directement par Fluent, alors que la cartographie de $y+$ sur la surface du cycliste doit être faite grâce à l'outil d'extraction de contour. Le résultat est visible sur la figure 3-32, la valeur maximum est de 1,29 ce qui valide le choix des dimensions de la couche limite et du modèle de modélisation de l'écoulement dans la couche limite donnée dans la partie 3.3.2 ($y+<5$).

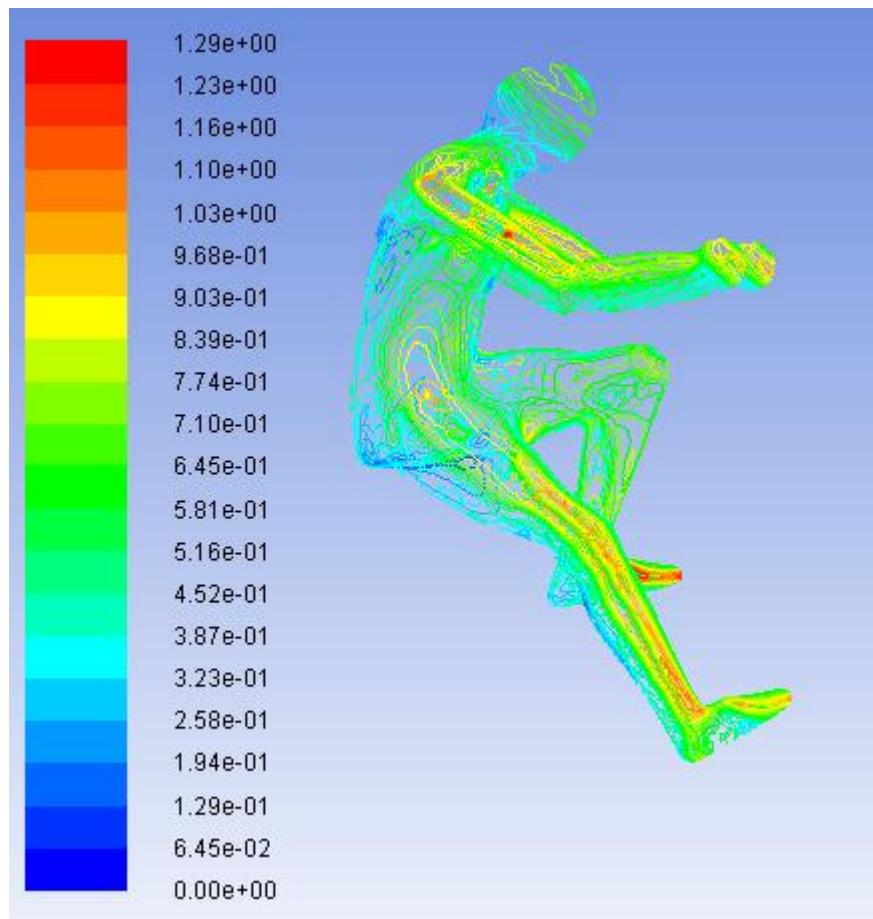


Figure 3-32 : Cartographie de $y+$ sur la surface du cycliste.

3.5 Les validations

Deux validations sont nécessaires dans le projet. La première concerne la mesure et la modélisation géométrique du cycliste, et la deuxième le maillage et la modélisation de l'écoulement. La première est nécessaire pour valider que le modèle obtenu est une modélisation acceptable du cycliste mesuré, et la deuxième pour valider les résultats obtenus et ainsi valider les paramètres de simulation utilisés.

3.5.1 La validation géométrique

La validation géométrique consiste à faire la comparaison entre le modèle Catia obtenu après mesure du cycliste et paramétrisation du modèle, et un scan de la surface extérieure du cycliste. Cinq sujets ayant donné leur consentement éclairé ont été recrutés selon le protocole approuvé par le comité d'éthique de l'école Polytechnique, avec les critères d'inclusions suivants :

- Pratique régulière du cyclisme (deux à trois fois par semaine)
- Étant âgé entre 18 et 60 ans

Deux femmes et trois hommes ont été ainsi recrutés. Ils ont été soumis à deux acquisitions, une utilisant le système Kinect développé dans la partie 3.2, et une autre utilisant le système Inspeck (Inspeck Inc., Montréal, Canada).



Figure 3-33 : Le système Inspect : Une caméra Inspeck (gauche), un ensemble de quatre caméra Inspeck (droite) : une de face, deux à 30° de chaque côté et une à l'arrière du sujet.

Le système Inspek (Figure 3-33) est composé de quatre caméras, trois frontales et une pour le dos. Cette disposition ne permet d'obtenir qu'une partie de la surface du tronc, car le système est utilisé pour la mesure de tronc de personnes scoliotiques afin d'aider à la création de corsets orthopédiques. Ainsi une acquisition du haut et une autre du bas du corps sont nécessaires.

Certaines étapes manuelles sont nécessaires :

- La segmentation de la zone d'intérêt
- La modification de la continuité : Des discontinuités de couleurs sur le sujet peut être à l'origine de discontinuité sur la surface.
- Le choix du point de référence permettant de déterminer la profondeur à laquelle se trouve le sujet (étape importante pour le regroupement des quatre maillages obtenus).

La continuité doit être corrigée dans certains cas lors de la présence d'un fort gradient de couleurs sur le sujet. Le traitement de ces acquisitions par le logiciel Faps permet d'obtenir un maillage de la zone d'intérêt.

Une fois que les acquisitions des quatre scanners ont été traitées, les quatre maillages sont importés dans le logiciel EM pour les fusionner. Un fichier de recalage permet d'approcher les maillages de leur position finale. Un nettoyage des maillages est ensuite effectué pour supprimer les imperfections souvent présentes sur les bords des maillages, zones étant plus difficiles à interpréter par le logiciel Faps.

Ensuite, un recalage automatique permet de rapprocher les maillages des deux scanners latéraux sur le maillage frontal par optimisation de la distance entre deux scans. Enfin, les quatre scans sont fusionnés.

La fusion de la partie supérieure et de la partie inférieure se déroule en deux étapes, le recalage des deux parties manuellement, et la découpe si nécessaire des morceaux de cuisse de la partie supérieure. La découpe peut être nécessaire si l'écartement des cuisses se trouve être changé entre les deux acquisitions ce qui crée une difficulté d'obtenir deux surfaces se superposant.

La surface de référence permettant de faire la comparaison est ainsi obtenue et peut être comparée au modèle obtenu. Pour une comparaison idéale, le modèle surfacique et le sujet scanné devraient être dans une position identique. Or pour des raisons de visibilité des scanners, les sujets ont été scannés les jambes un peu écartées, degrés de liberté non pris en compte par le modèle.

Ainsi la comparaison sera effectuée membre par membre, seule une inspection visuelle sera effectuée sur l'ensemble du modèle avec le scan complet. L'inspection visuelle servira principalement pour la comparaison des longueurs.

La comparaison des membres se déroule en trois étapes pour chaque membre sous le logiciel Catia :

- Découper le membre à comparer du reste du corps sur le modèle et le scan
- Mise en position manuelle du membre du modèle sur le scan du sujet
- Utilisation de l'outil *Distance Analysis* dans l'atelier *ICEM Shape Design Aeroexpert* de Catia permettant de calcul la distance normale entre deux surfaces en tout point.

Une cartographie des distances est obtenue ainsi que les distances maximum et minimum.

3.5.2 La validation CFD

Les résultats de la simulation ont été comparés à ceux de la littérature (Defraeye et al., 2010a).



Figure 3-34 : Différentes positions du cycliste (Article de Defraeye (Defraeye et al., 2010a)) : a) Position haute, b) Position aérodynamique, c) Position contre la montre.

La première position est celle qui crée le plus de forces de traînée, et donc celle qui peut permettre de mieux voir les différences dans les résultats au vu des forces plus grandes. Elle sera une des positions comparées. La position aérodynamique est aussi comparée.

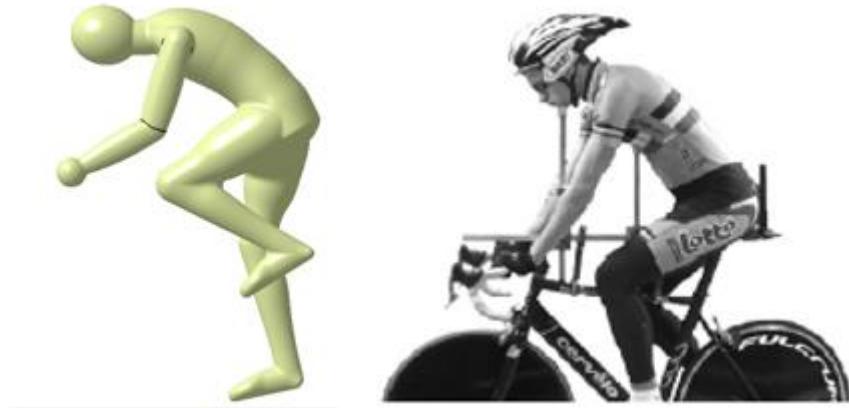


Figure 3-35 : Position du modèle (gauche) et position dans l'article de référence (droite : image tirée de l'article de Thijs Defraeye (Defraeye et al., 2010a)).

La première position (Figure 3-34-a) comparée est celle de la position haute. Dans un premier temps, il est important que les deux modèles aient les mêmes dimensions. Pour cela, seule la dimension entre les mains et l'arrière mesurée dans le sens de l'écoulement est fournie, elle est de 0,85 m, la distance du sur le modèle créé est de 0,875 m. Une autre dimension est donnée permettant seulement de comparer le modèle au sujet, c'est l'aire frontale du cycliste, elle est de 0,41 m² pour le sujet et de 0,388 m² pour le modèle. La position est donc une position intermédiaire entre la position haute et la position aérodynamique de Defraeye. Une interpolation des résultats sera nécessaire pour déterminer les résultats obtenus.

Les mêmes conditions de vents sont appliquées. La direction du vent est frontale au cycliste et la vitesse du vent étant de 10m/s dans la partie simulation de l'article de Thijs Defraeye. L'extraction des résultats permettra donc de valider la simulation grâce au coefficient de traînée.

La deuxième position (Figure 3-34-b) à une aire frontale de 0,371 m² au niveau du deuxième modèle créé, qui correspond à l'aire de 0,37 m² de la position aérodynamique de l'article.

CHAPITRE 4 RÉSULTATS

Dans ce chapitre, les résultats de la validation géométrique formeront la première partie du chapitre alors que ceux permettant la validation de la simulation CFD se trouvent dans la deuxième partie du chapitre.

4.1 Validation géométrique

La validation géométrique est, comme expliqué dans la section 3.5.1, une mesure de la distance normale entre deux surfaces permise par le logiciel Catia. Pour cela, les acquisitions des cinq sujets sont traitées pour obtenir les paramètres de création du modèle CAO. Ainsi cinq modèles différents sont obtenus. D'un autre côté, les acquisitions du scanner Inspeck sont traitées pour obtenir les cinq maillages. Le maillage de point importé dans Catia est superposé au modèle développé (exemple figure 4-1).



Figure 4-1 : Superposition du scan et du modèle.

Les cartographies des mesures de distances prennent des allures comme celles des figures 4-2 et 4-3.

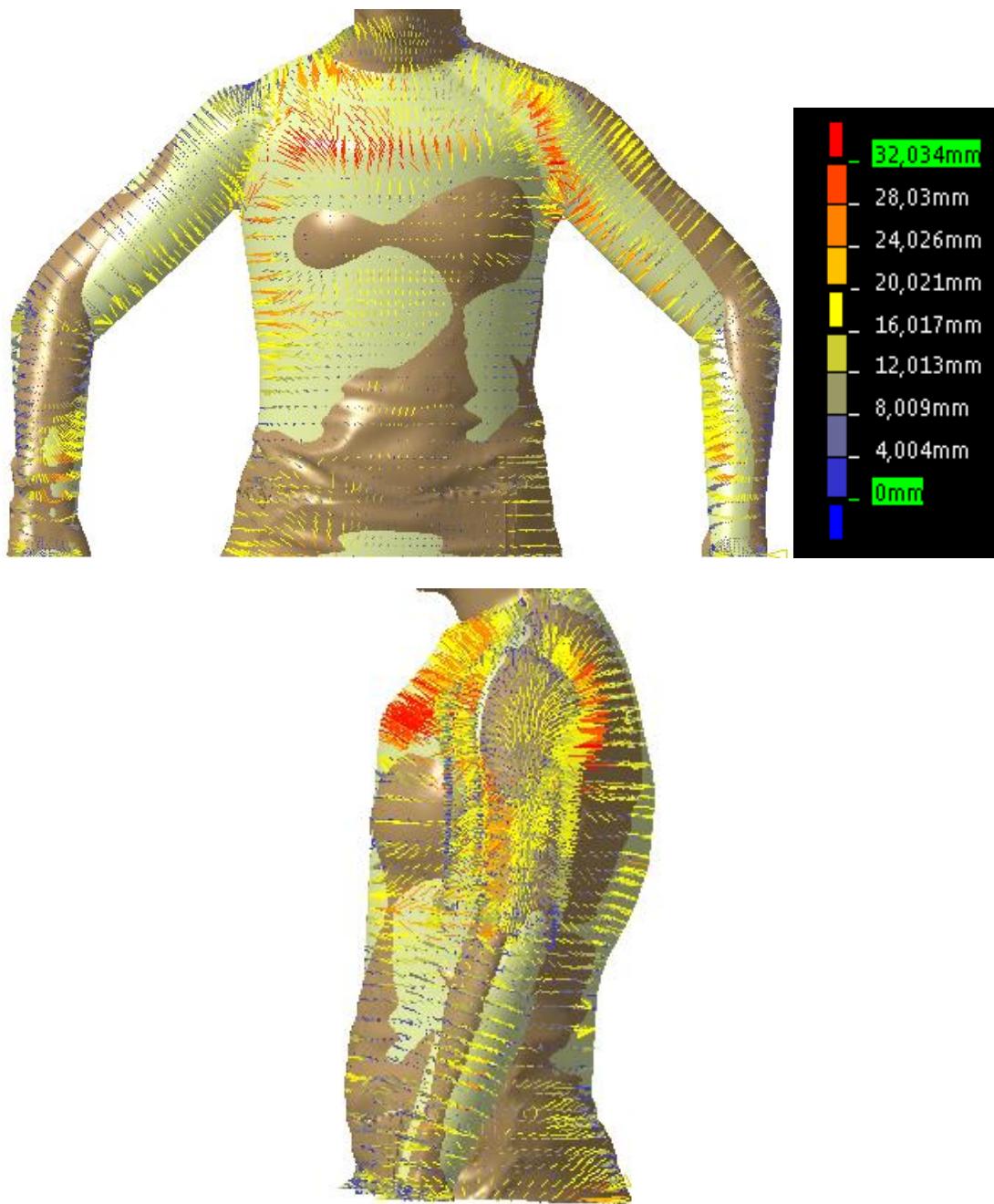


Figure 4-2 : Mesure normale à la surface du modèle entre les deux surfaces obtenues sur le haut du corps.

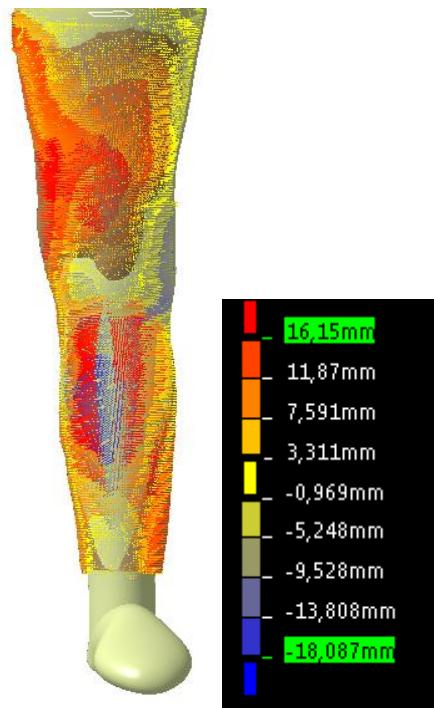


Figure 4-3 : Mesure normale à la surface du modèle entre les deux surfaces obtenues sur une jambe.

Grâce à ces cartographies, la distance maximale peut être récupérée. Ainsi en réalisant une mesure du haut du corps et une autre d'une jambe le tableau 4-1 regroupant ces différentes valeurs est créé.

Lors des mesures, un problème a été repéré. Lors des comparaisons des deux surfaces, une différence importante en taille est apparue. Un rapport de 0,9 existe entre les deux surfaces, ainsi la taille du maillage a été multipliée par 0,9 pour tous les sujets afin de pouvoir continuer les comparaisons. Cette différence provient de l'acquisition servant à la calibration, le calibre est un disque compact de rayon de 12cm, or une erreur sur la segmentation de 2 pixels de chaque côté entraîne une erreur de 10 % sur la mesure, d'où la différence de 0,9. La technique de segmentation à tendance à sous-estimer la taille réelle de l'objet mesuré.

Tableau 4-1 : Distance maximale entre le scan et le modèle.

Sujet	Haut du corps	Jambe
1	32 mm	18 mm
2	36 mm	35 mm
3	33 mm	39 mm
4	25 mm	35 mm
5	56 mm	24 mm

Les distances maximales sont élevées, mais ne reflètent pas le résultat final. Catia ne permet pas d'obtenir de valeur moyenne ou l'écart-type. Ainsi seule une analyse visuelle des résultats est possible. La figure 4-2 montre la répartition des distances de la partie supérieure du corps entre les deux surfaces, et la figure 4-3 pour une jambe. Plusieurs zones ont des distances plus élevées que les autres sur l'ensemble des cinq sujets :

- Au niveau du haut du torse antérieur et postérieur
- Au niveau des épaules
- Au niveau du genou

Ainsi la distance maximale permet de détecter les zones problématiques. Mais cette distance n'est présente que sur certaines zones, la distance moyenne entre les deux surfaces se trouvent être entre 50 % et 66 % de la distance maximale. Cette valeur est obtenue par observation des différentes cartographies des distances. Régulièrement, une couleur prédomine, comme la couleur jaune sur la figure 4-2. Donc sur les différentes prises de mesures, la distance moyenne est de 15 à 20 mm.

La distance maximale au-dessus des autres du sujet 5 provient d'une position non verticale du tronc, la courbure du ventre à fausser les valeurs (Figure 4-4). La modélisation permet de réaliser un ventre plat, ainsi la surface à la même profondeur, mais est décalée vers la partie postérieure du sujet.

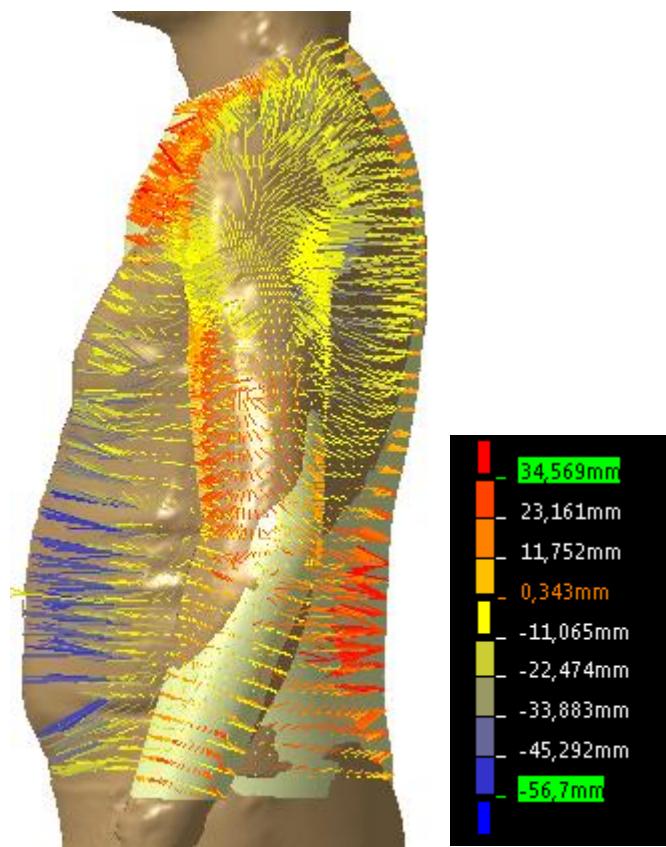


Figure 4-4 : Mesure de haut du corps du sujet 5.

4.2 Validation CFD

La première position utilisée a été une position intermédiaire entre la position haute et la position aérodynamique utilisées par Defraeye. Car l'aire frontale de la position haute est de $0,41 \text{ m}^2$ et celle de l'intermédiaire est de $0,37 \text{ m}^2$ alors que le modèle simulé est de $0,388 \text{ m}^2$. Ainsi la comparaison va se faire grâce aux deux positions.

Tableau 4-2 : Valeurs des coefficients AC_D simulés et de l'article de Defraeye.

	Position du cycliste	$AC_D (\text{m}^2)$	Aire (m^2)
Defraeye	Position haute en soufflerie	0,193	0,41
	Position aérodynamique en soufflerie	0,165	0,37
	Position haute simulée	0,219	0,41
	Position aérodynamique simulée	0,179	0,37
Interpolation	Position interpolée en soufflerie	0,175	0,39
	Position interpolée simulée	0,197	0,39
Modèle	Première Position simulée	0,177	0,39
	Deuxième Position simulée	0,170	0,37

La simulation CFD de la première position donne un AC_D (produit de l'aire frontale par le coefficient de la force de traînée) de $0,177 \text{ m}^2$. Or cette position à une aire frontale intermédiaire entre la position haute et la position aérodynamique. En utilisant une interpolation entre les deux positions de la littérature, le AC_D correspondant est de $0,175 \text{ m}^2$ pour les résultats en soufflerie et $0,197 \text{ m}^2$ pour ceux venant de la CFD.

Ainsi la simulation CFD donne des résultats concluants et proches de la littérature. Car pour les deux positions simulées, le coefficient calculé se trouve être compris entre les valeurs simulées et les valeurs mesurées en soufflerie par Defraeye.

De nombreux autres résultats peuvent être créés à partir des simulations CFD. Tel que le coefficient de pression créée par le flux sur le cycliste (Figure 4-5). La pression est bien plus importante sur les zones où le fluide est en incidence normale à la surface. Alors que les zones où le flux est perpendiculaire à la normale de la surface, une dépression apparaît.

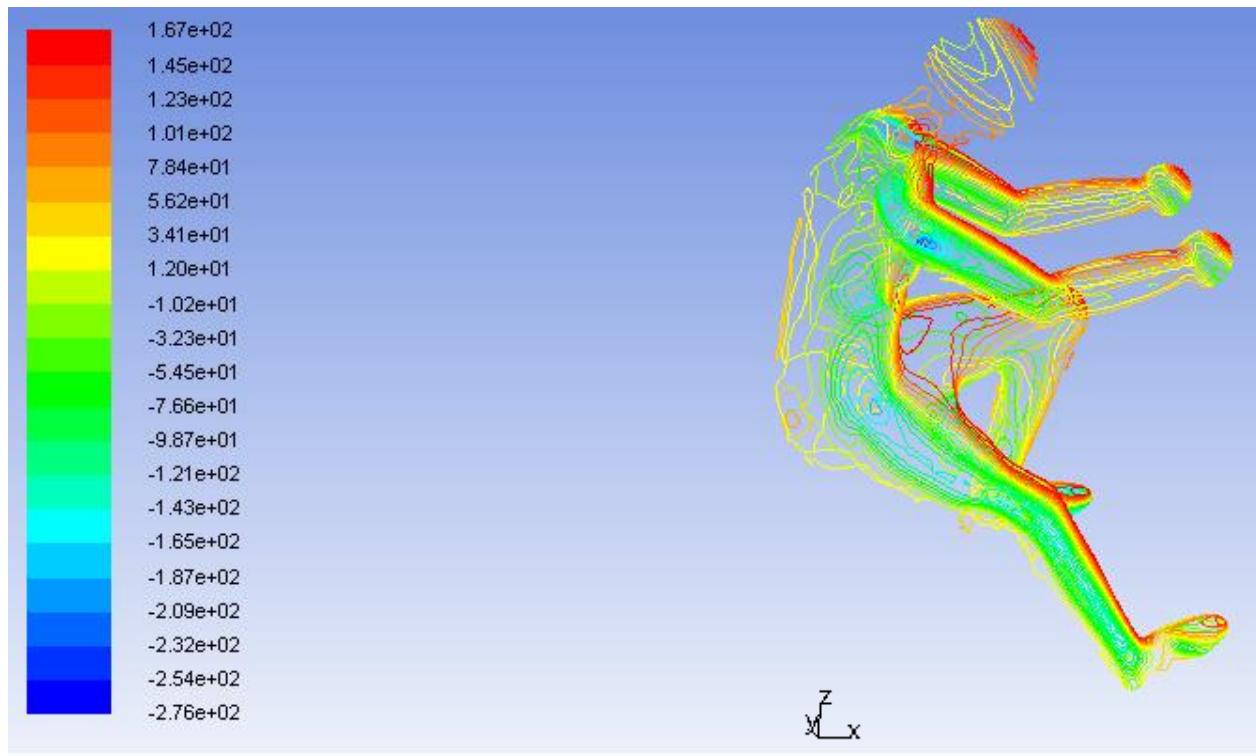


Figure 4-5 : Coefficient de pression de la position haute (sans unité).

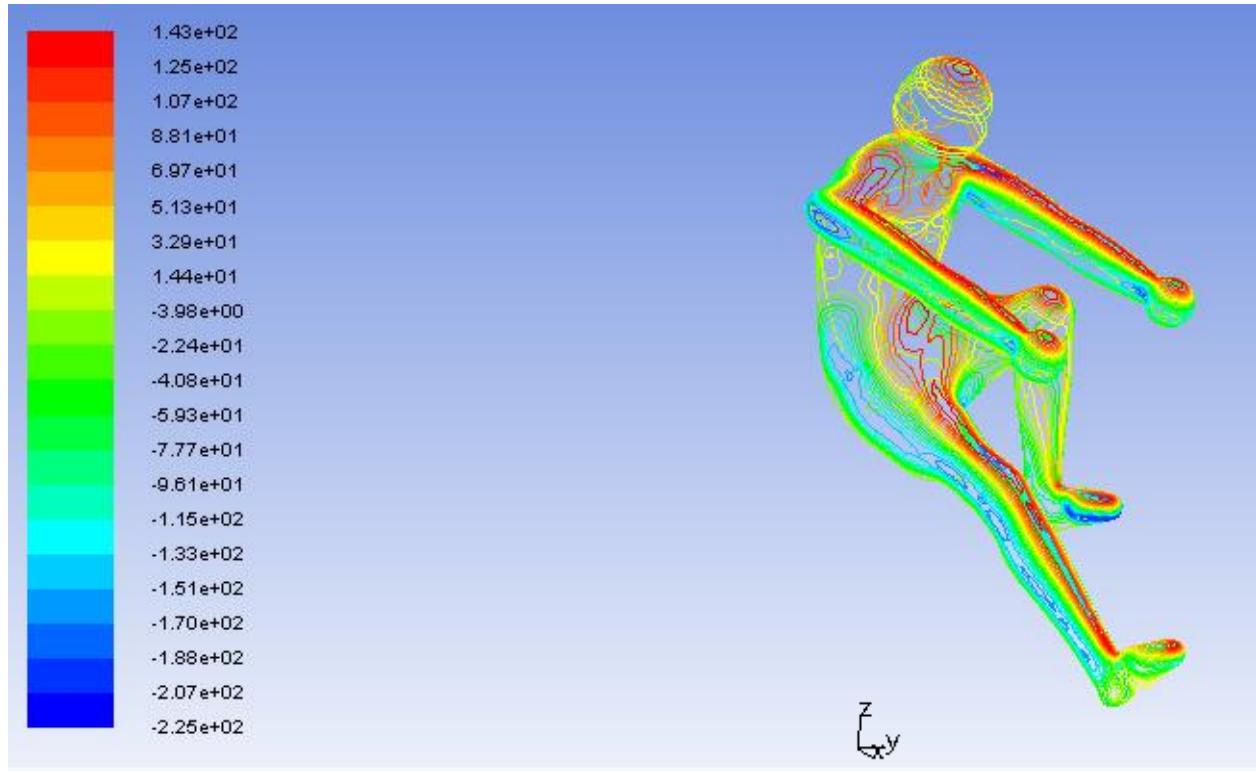


Figure 4-6 : Coefficient de pression de la position aérodynamique.

Entre les deux positions, une plus forte pression est présente dans le cas de la position haute par rapport à la position aérodynamique. Cela vient du fait que la position haute a le haut du corps plus droit, ainsi le flux à une incidence qui se rapproche plus d'une incidence normale par rapport à la position aérodynamique où l'incidence normale du fluide est évitée. La pression maximale est créée par un flux en incidence normale sur une surface, la minimale est quand l'incidence est parallèle à la surface.

CHAPITRE 5 DISCUSSION

Cette partie a pour but d'apporter un regard critique sur le projet en passant par la technique de modélisation puis par celle de mesure et enfin par la modélisation CFD. Pour chaque partie, la validation correspondante ainsi que la littérature sera utilisée pour appuyer la discussion. La création du mannequin 3D et sa validation constitue la première partie. La deuxième partie concerne le système de mesure, alors que la troisième partie est sur la simulation CFD et sa validation.

5.1 Mannequin

La création du mannequin est une avancée dans le domaine du cycliste. Beaucoup d'études portent sur un scan du cycliste. Ce scan exige une part importante dans le temps alloué à la création de la surface utilisée lors de la CFD. La création d'un modèle simplifié du cycliste permet d'avoir une rapidité d'acquisition et de création de la surface. Ainsi l'utilisation du mannequin demande peu de temps au cycliste en prise de mesure par rapport à l'acquisition d'un nuage de point par scanner. Dans le cas de plusieurs positions à analyser, comme fait par Defraeye et coll. (Defraeye et al., 2010a), chaque position du cycliste doit être scannée, alors que l'utilisation de notre solution ne demande qu'une acquisition, les différentes positions sont créées en modifiant les paramètres d'angles du modèle. Ainsi notre modèle permet un gain de temps considérable pour créer les géométries nécessaires aux simulations CFD.

Le côté paramétrique du modèle apporte de nombreuses possibilités de développement tel que l'implantation des données physiologiques pour prévenir des positions pouvant indisposer le cycliste, comme le module ergonomique de Catia et le mannequin. Mais le mannequin de Catia ne permet pas d'extraire facilement la surface pour réaliser la simulation CFD. Aussi, la possibilité d'utiliser le modèle dans d'autres domaines, tels que le saut à ski qui repose aussi sur la position du sujet pour améliorer ses performances aérodynamiques (Meile et al., 2006).

Du point de vue du choix de modélisation du mannequin, la précision du modèle sur les cinq sujets donne une erreur de 3,5 cm au maximum, mais une valeur moyenne dans les environs de 1,5-2,0 cm. Par rapport au 20 cm en moyenne de diamètre des jambes ou des 30 cm de largeur du tronc, cette valeur est acceptable. Mais elle peut être améliorée dans le futur par l'amélioration

des zones à problème telles que le tronc qui repose sur peu de mesures par rapport aux autres membres.

La surface du corps humain est difficilement modélisable par des règles simples, car le corps humain est différent pour chacun. Par exemple la présence de la poitrine entre les hommes et les femmes. Ainsi la modélisation des spécificités de chacun est chose difficile avec la modélisation utilisée, car elle est orientée vers des personnes ayant un physique sportif. Par exemple, le choix de modélisation du tronc a été de la modéliser par un ventre plat. Ainsi dans le cas de présence de masse graisseuse au niveau du ventre, une importante différence entre la réalité et la modélisation sera présente.

La modélisation du mannequin utilise un logiciel largement utilisé dans le domaine aéronautique. Mais de nombreux obstacles de modélisation sont apparus provenant du logiciel.

La paramétrisation du modèle ainsi que son articulation sont source de modification de nombreuses surfaces et esquisses intermédiaires lors de changement de dimension. Ainsi des droites et des ellipses se retrouvent être recréées. Or ces deux entités ont une orientation ne pouvant pas être contrôlée. Cette orientation peut donc s'inverser lors de ces mises à jour et créer des erreurs empêchant la reconstruction globale du modèle. Par exemple, l'extrémité droite d'une droite peut être utilisée pour la construction d'une entité la précédent. Ensuite, lors de la mise à jour, ce point peut se retrouver à gauche de la droite, malgré l'apparence conservée de la droite. Le problème a été contourné en modifiant la modélisation pour avoir à éviter d'utiliser les extrémités des segments. Ainsi comme le modèle surfacique repose essentiellement sur des droites et des ellipses, le logiciel Catia comporte de nombreuses contraintes de construction.

5.2 Kinect

Le système Kinect est un système peu onéreux permettant une acquisition rapide. Lors du projet, la version 1 de la Kinect a été utilisée. Cependant, une version 2 a été mise sur le marché pendant le projet et présente de meilleures performances. Microsoft ainsi affirme que la détection des joints se trouve être trois fois plus précise, et sa résolution passe à 1920x1080 pour la caméra couleur.

Les mesures sont localisées, mais il est possible d'augmenter ce nombre pour améliorer la précision du modèle. Par exemple, une augmentation des mesures peut permettre de déterminer l'emplacement exact de la lordose et de la cyphose de la colonne et ainsi pouvoir mieux modéliser la courbure de la colonne par la sinusoïde en modifiant des paramètres tels que son déphasage et sa période.

La Kinect est un outil vers lequel plusieurs chercheurs se tournent de plus en plus (Rougier, Auvinet, Rousseau, Mignotte, & Meunier, 2011). Une des raisons qui la rend attractive est son prix modique. Mais ce prix est compréhensible au vu de la qualité de certains éléments du système. Un des points de qualité étant limité est la résolution de la caméra qui limite la précision des mesures. La résolution de 640x480 couplée avec l'emplacement en profondeur des sujets (environ 2 m) a permis de déterminer que le facteur de conversion de 0,4 cm/pixel est l'ordre de grandeur, ce qui est loin de ce qu'une caméra peut offrir en comparaison.

Le contraste de la caméra a aussi été un problème. Malgré un fond uni et contrastant bien avec le sujet placé au premier plan, le gradient des niveaux de gris (après transformation par le logiciel Matlab) n'était pas suffisant pour la détection de contour. Ainsi la détection demande des conditions d'éclairage difficiles à remplir. Lors de la validation, la segmentation a dû être faite manuellement.

Le dernier point lié à la Kinect concerne la détection des articulations. Cette détection est réalisée pour détecter les mouvements du corps et plus particulièrement ceux des membres supérieurs. Par contre, la détection des membres inférieurs a une précision moindre. Ainsi une des perspectives d'amélioration est de créer un programme pour replacer manuellement les articulations.

Il est aussi possible de modifier le programme de manière à pouvoir l'utiliser pour d'autres projets demandant des mesures du corps humain. Par exemple, les compagnies utilisent des systèmes de mesures des angles du corps humain pour obtenir la position exacte d'un cycliste sur son vélo. Ainsi l'utilisation de la Kinect peut être une solution viable grâce à l'utilisation de la détection des articulations intégrée pour réaliser ce point.

5.3 Choix de la modélisation CFD

La simulation CFD donne de bons résultats, car la valeur recherchée est de $0,167 \text{ m}^2$, et $0,165 \text{ m}^2$ a été obtenu. De plus, la convergence de la simulation se trouve être rapide (environ 10 min).

Aucune comparaison n'a été effectuée entre différentes positions d'un même sujet. La modification de la position du sujet demande seulement une recréation du maillage et de faire une nouvelle simulation. Un script de création du maillage a été réalisé qui permet de recréer rapidement le maillage. La nouvelle simulation garde les mêmes paramètres, seul le maillage d'entrée se trouve changé. Ainsi réaliser de nouvelles simulations permettant de comparer différentes positions du cycliste est chose aisée.

Le choix de la modélisation de la turbulence a été de prendre la plus couramment utilisée dans le domaine sportif et une qui donne de bons résultats dans le cas d'un cycliste selon Defraeye et coll. (Defraeye et al., 2010b). Ainsi le choix du type de simulation peut être un des points à améliorer, car il n'est pas non plus le meilleur selon le même article. Le meilleur étant une simulation LES, mais ce type de simulation demande un temps de calcul plus long. Ainsi une série de position et cycliste pourrait être simulée pour permettre de comparer les différences de résultats entre plusieurs types de modélisation ainsi que leurs temps de convergence. Un compromis entre le temps de simulation et la précision des résultats pourra être fait grâce à cela et un choix de modélisation permettant une optimisation du temps de calcul en découlera.

CONCLUSION

Le travail présenté dans ce mémoire consiste en la création d'outils permettant de réaliser des simulations CFD pour déterminer les performances aérodynamiques de n'importe quel cycliste.

Dans un premier temps, un modèle surfacique a été créé à l'aide du logiciel Catia. Ce modèle articulé autour d'un modèle multicorps est une modélisation simplifiée de la surface extérieure du corps humain. Sa création passe par la définition d'ellipses servant de support pour la surface. Le modèle est paramétré par une multitude de longueurs permettant de le modifier et ainsi pouvoir représenter un large éventail de morphologies.

Dans un deuxième temps, l'utilisation de la Kinect a permis de mesurer différents paramètres nécessaires à la mise sur mesure du modèle. L'utilisation du squelette permet d'obtenir les dimensions du modèle multicorps qui fait office de squelette, malgré une détection d'articulation nécessitant de temps en temps un recalage manuel de certaines mesures, dont les membres inférieurs. L'utilisation de deux photos, une de face et une de profil, permet de déterminer les dimensions des ellipses du modèle et donc permettre de modifier la corpulence du modèle et ainsi pouvoir représenter le mieux possible le sujet. La validation du système de mesure et du mannequin a montré que des écarts entre le résultat obtenu et le sujet scanné sont présents, mais que le modèle est validé.

Enfin, une simulation CFD a été réalisée avec la surface obtenue du modèle dans une position similaire à celle utilisé dans l'article de Defraeye et coll.(Defraeye et al., 2010a). Cette simulation est réalisée dans un cas simple de vent de face avec une vitesse de 10m/s, et les résultats obtenus permettent de valider les paramètres choisis ainsi que la construction du maillage.

Les différents objectifs du projet ont ainsi été réalisés. Par contre, certains points du projet peuvent être sujets d'amélioration. Ainsi l'utilisation de la Kinect peut être améliorée avec l'utilisation d'un système plus précis, mais aussi un travail sur les conditions d'exposition peut être amélioré pour permettre une meilleure segmentation avec le logiciel Matlab.

Au niveau du modèle CATIA, certaines zones peuvent être améliorées pour diminuer l'écart avec les dimensions du sujet réel. Un des points d'amélioration est la modélisation du tronc qui ne repose que sur quelques mesures et est source d'une partie de la différence. Aussi, du point de vue

du modèle multicorps, un rajout de degrés de liberté supplémentaires à certaines articulations, pourrait permettre une utilisation du modèle pour d'autres domaines d'étude que le cyclisme.

Au niveau de la modélisation CFD, le choix de la modélisation de la turbulence est matière à amélioration. Malgré les bons résultats obtenus, un choix de modélisation différente peut amener à une optimisation des temps de simulations.

BIBLIOGRAPHIE

- Barber, S., Chin, S., & Carré, M. (2009). Sports ball aerodynamics: a numerical study of the erratic motion of soccer balls. *Computers & Fluids*, 38(6), 1091-1100.
- Bixler, B., Pease, D., & Fairhurst, F. (2007). The accuracy of computational fluid dynamics analysis of the passive drag of a male swimmer. *Sports Biomechanics*, 6(1), 81-98.
- Bixler, B., & Riewald, S. (2002). Analysis of a swimmer's hand and arm in steady flow conditions using computational fluid dynamics. *Journal of Biomechanics*, 35(5), 713-717. doi: [http://dx.doi.org/10.1016/S0021-9290\(01\)00246-9](http://dx.doi.org/10.1016/S0021-9290(01)00246-9)
- Bixler, B., & Schloder, M. (1996). Computational fluid dynamics: an analytical tool for the 21st century swimming scientist. *Journal of swimming research*, 4-22.
- Broker, J. P. (2003). Cycling power: road and mountain. *High-tech cycling. Human Kinetics, Champaign*, 147-174.
- Defraeye, T., Blocken, B., Koninckx, E., Hespel, P., & Carmeliet, J. (2010a). Aerodynamic study of different cyclist positions: CFD analysis and full-scale wind-tunnel tests. *Journal of Biomechanics*, 43(7), 1262-1268. doi: <http://dx.doi.org/10.1016/j.jbiomech.2010.01.025>
- Defraeye, T., Blocken, B., Koninckx, E., Hespel, P., & Carmeliet, J. (2010b). Computational fluid dynamics analysis of cyclist aerodynamics: Performance of different turbulence-modelling and boundary-layer modelling approaches. *Journal of Biomechanics*, 43(12), 2281-2287. doi: <http://dx.doi.org/10.1016/j.jbiomech.2010.04.038>
- Defraeye, T., Blocken, B., Koninckx, E., Hespel, P., & Carmeliet, J. (2011). Computational fluid dynamics analysis of drag and convective heat transfer of individual body segments for different cyclist positions. *Journal of Biomechanics*, 44(9), 1695-1701. doi: <http://dx.doi.org/10.1016/j.jbiomech.2011.03.035>
- García-López, J., Rodríguez-Marroyo, J. A., Juneau, C.-E., Peleteiro, J., Martínez, A. C., & Villa, J. G. (2008). Reference values and improvement of aerodynamic drag in professional cyclists. *Journal of sports sciences*, 26(3), 277-286.
- Grappe, F., Candau, R., Belli, A., & Rouillon, J. D. (1997). Aerodynamic drag in field cycling with special reference to the Obree's position. *Ergonomics*, 40(12), 1299-1311.
- Gray, H. (1918). *Anatomy of the human body*: Lea & Febiger.
- Gross, A. C., Kyle, C. R., & Malewicki, D. J. (1983). The aerodynamics of human-powered land vehicles. *Scientific American*, 249(6), 142-152.
- Hanavan Jr, E. P. (1964). A mathematical model of the human body: DTIC Document.
- Hatze, H. (1980). A mathematical model for the computational determination of parameter values of anthropomorphic segments. *Journal of Biomechanics*, 13(10), 833-843. doi: [http://dx.doi.org/10.1016/0021-9290\(80\)90171-2](http://dx.doi.org/10.1016/0021-9290(80)90171-2)

- Jensen, R. K. (1978). Estimation of the biomechanical properties of three body types using a photogrammetric method. *Journal of Biomechanics*, 11(8–9), 349-358. doi: [http://dx.doi.org/10.1016/0021-9290\(78\)90069-6](http://dx.doi.org/10.1016/0021-9290(78)90069-6)
- Jeukendrup, A. E., & Martin, J. (2001). Improving cycling performance. *Sports Medicine*, 31(7), 559-569.
- Kasap, M. (2011). *Generating and customizing virtual human body models*. University of Geneva.
- Kyle, C. R., & Burke, E. R. (1984). Improving the racing bicycle. *Mechanical engineering*, 106(9), 34-45.
- MARIEB, E. N. (2004). Anatomie et physiologie humaines [Livre]: Paris: PEARSON.
- Meile, W., Reisenberger, E., Mayer, M., Schmölzer, B., Müller, W., & Brenn, G. (2006). Aerodynamics of ski jumping: experiments and CFD simulations. *Experiments in Fluids*, 41(6), 949-964.
- Nissan, M., & Gilad, I. (1986). Dimensions of human lumbar vertebrae in the sagittal plane. *Journal of Biomechanics*, 19(9), 753-758.
- Rougier, C., Auvinet, E., Rousseau, J., Mignotte, M., & Meunier, J. (2011). Fall detection from depth map video sequences *Toward Useful Services for Elderly and People with Disabilities* (pp. 121-128): Springer.
- Yeadon, M. R. (1990). The simulation of aerial movement—II. A mathematical inertia model of the human body. *Journal of Biomechanics*, 23(1), 67-74. doi: [http://dx.doi.org/10.1016/0021-9290\(90\)90370-I](http://dx.doi.org/10.1016/0021-9290(90)90370-I)

ANNEXE A – Listes des paramètres

Angles :

- Angle_Genou_G
- Angle_Pied_G
- Angle_Genou_D
- Angle_Pied_D
- Angle_Jambe_D
- Angle_Jambe_G
- Angle_Thoracique
- Angle_Lombaire
- Angle_Abduction_Bras
- Angle_RotationLateral_Bras
- Angle_Flexion_AvantBras
- Angle_Tete
- Inclinaison_Globale

Longueurs :

- Longueur_Cuisse
- Longueur_Tibia
- Longueur_Pied
- Longueur_Bras
- Longueur_AvantBras
- Longueur_Torce
- Longueur_Tete
- Largeur_Epaule
- Largeur_Hanche

Ellipses :

- Bras_E1_r
- Bras_E2_a
- Bras_E2_b
- Bras_E3_a
- Bras_E3_b
- Avant-Bras_E2_a
- Avant-Bras_E2_b
- Avant-Bras_E3_a
- Avant-Bras_E3_b
- Avant-Bras_Main_r

- Cuisse_E2_r
- Cuisse_E3_r
- Cuisse_E4_r
- Tibia_E2_r
- Tibia_E3_r
- Tibia_E4_a
- Tibia_E4_b
- Tronc_Amplitude_a
- Tronc_Moyenne_a
- Tronc_Amplitude_b
- Tronc_Moyenne_b
- Tronc_Tete_a
- Tronc_Tete_b

ANNEXE B – Code du logiciel

Code du comportement du logiciel (Main Windows)

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Navigation;
using System.Windows.Shapes;
using Microsoft.Kinect;
using System.IO;

namespace KinectPhoto
{
    /// <summary>
    /// Interaction logic for MainWindow.xaml
    /// </summary>
    public partial class MainWindow : Window
    {
        #region Member Variables
        private KinectSensor _KinectDevice;
        private readonly Brush[] _SkeletonBrushes;
        private Skeleton[] _FrameSkeletons;
        #endregion Member Variables

        #region Constructor
        public MainWindow()
        {
            InitializeComponent();

            this._SkeletonBrushes = new[] { Brushes.Black, Brushes.Crimson,
Brushes.Indigo, 
                                         Brushes.DodgerBlue, Brushes.Purple,
Brushes.Pink };

            KinectSensor.KinectSensors.StatusChanged += KinectSensors_StatusChanged;
            this.KinectDevice = KinectSensor.KinectSensors.FirstOrDefault(x => x.Status == KinectStatus.Connected);
        }
        #endregion Constructor

        #region Methods
        private void KinectSensors_StatusChanged(object sender, StatusChangedEventArgs e)
        {
            switch (e.Status)
            {
                case KinectStatus.Initializing:

```

```

        case KinectStatus.Connected :
            this._KinectDevice = e.Sensor;
            break;
        case KinectStatus.Disconnected:
            //TODO: Give the user feedback to plug-in a Kinect Device.
            this._KinectDevice = null;
            break;
        default:
            //TODO: Show an error state
            break;
    }
}
#endregion Methods

#region Properties
public KinectSensor KinectDevice
{
    get { return this._KinectDevice; }
    set
    {
        if (this._KinectDevice != value)
        {
            //Uninitialize
            if (this._KinectDevice != null)
            {
                this._KinectDevice.Stop();
                this._KinectDevice.SkeletonFrameReady -=
KinectDevice_SkeletonFrameReady;
                this._KinectDevice.SkeletonStream.Disable();
                this._KinectDevice.ColorFrameReady -=
KinectDevice_ColorFrameReady;
                this._KinectDevice.ColorStream.Disable();
                //this._KinectDevice.DepthFrameReady -=
KinectDevice_DepthFrameReady;
                //this._KinectDevice.DepthStream.Disable();

                this._FrameSkeletons = null;
            }

            this._KinectDevice = value;

            //Initialize
            if (this._KinectDevice != null)
            {
                if (this._KinectDevice.Status == KinectStatus.Connected)
                {
                    this._KinectDevice.SkeletonStream.Enable();
                    this._FrameSkeletons = new
Skeleton[this._KinectDevice.SkeletonStream.FrameSkeletonArrayLength];
                    this._KinectDevice.SkeletonFrameReady +=

KinectDevice_SkeletonFrameReady;
                    this._KinectDevice.ColorStream.Enable();
                    this._KinectDevice.ColorFrameReady +=

KinectDevice_ColorFrameReady;
                    //this._KinectDevice.DepthStream.Enable();
                    //this._KinectDevice.DepthFrameReady +=

KinectDevice_DepthFrameReady;
                }
            }
        }
    }
}

```

```

                this._KinectDevice.Start();
            }
        }
    }
}

#endregion Properties

private void KinectDevice_ColorFrameReady(object sender,
ColorImageFrameReadyEventArgs e)
{
    using (ColorImageFrame frame = e.OpenColorImageFrame())
    {
        if (frame != null)
        {
            byte[] pixelData = new byte[frame.PixelDataLength];
            frame.CopyPixelDataTo(pixelData);

            ColorImageElement.Source = BitmapImage.Create(frame.Width,
frame.Height, 96, 96, PixelFormats.Bgr32, null, pixelData, frame.Width *
frame.BytesPerPixel);
        }
    }
}

//private void KinectDevice_DepthFrameReady(object sender,
DepthImageFrameReadyEventArgs e)
//{
//    using (DepthImageFrame frame = e.OpenDepthImageFrame())
//    {
//        if (frame != null)
//        {
//            short[] pixelData = new short[frame.PixelDataLength];
//            frame.CopyPixelDataTo(pixelData);
//            int stride = frame.Width * frame.BytesPerPixel;
//            DepthImage.Source = BitmapSource.Create(frame.Width, frame.Height,
96, 96, PixelFormats.Gray16, null, pixelData, stride);
//        }
//    }
//}

private void KinectDevice_SkeletonFrameReady(object sender,
SkeletonFrameReadyEventArgs e)
{
    using (SkeletonFrame frame = e.OpenSkeletonFrame())
    {
        if (frame != null)
        {
            Polyline figure;
            Brush userBrush;
            Skeleton skeleton;

            LayoutRoot.Children.Clear();
            frame.CopySkeletonDataTo(this._FrameSkeletons);

            Skeleton[] dataSet2 = new Skeleton[this._FrameSkeletons.Length];
            frame.CopySkeletonDataTo(dataSet2);
        }
    }
}

```

```

        for (int i = 0; i < this._FrameSkeletons.Length; i++)
        {
            skeleton = this._FrameSkeletons[i];

            if (skeleton.TrackingState == SkeletonTrackingState.Tracked)
            {
                userBrush = this._SkeletonBrushes[i % this._SkeletonBrushes.Length];

                //Draw head and torso
                figure = CreateFigure(skeleton, userBrush, new[] {
                    JointType.Head, JointType.ShoulderCenter, JointType.HipCenter
                });
                LayoutRoot.Children.Add(figure);

                figure = CreateFigure(skeleton, userBrush, new[] {
                    JointType.ShoulderLeft, JointType.ShoulderCenter, JointType.ShoulderRight });
                LayoutRoot.Children.Add(figure);

                //Draw left leg
                figure = CreateFigure(skeleton, userBrush, new[] {
                    JointType.HipCenter, JointType.HipLeft, JointType.KneeLeft, JointType.AnkleLeft,
                    JointType.FootLeft });
                LayoutRoot.Children.Add(figure);

                //Draw right leg
                figure = CreateFigure(skeleton, userBrush, new[] {
                    JointType.HipCenter, JointType.HipRight, JointType.KneeRight, JointType.AnkleRight,
                    JointType.FootRight });
                LayoutRoot.Children.Add(figure);

                //Draw left arm
                figure = CreateFigure(skeleton, userBrush, new[] {
                    JointType.ShoulderLeft, JointType.ElbowLeft, JointType.WristLeft, JointType.HandLeft });
                LayoutRoot.Children.Add(figure);

                //Draw right arm
                figure = CreateFigure(skeleton, userBrush, new[] {
                    JointType.ShoulderRight, JointType.ElbowRight, JointType.WristRight, JointType.HandRight
                });
                LayoutRoot.Children.Add(figure);

                Head.Text = string.Format("{0}",
                GetJointPoint(skeleton.Joints[JointType.Head]));
                ShoulderCenter.Text = string.Format("{0}",
                GetJointPoint(skeleton.Joints[JointType.ShoulderCenter]));
                Spine.Text = string.Format("{0}",
                GetJointPoint(skeleton.Joints[JointType.Spine]));
                HipCenter.Text = string.Format("{0}",
                GetJointPoint(skeleton.Joints[JointType.HipCenter]));

                ShoulderRight.Text = string.Format("{0}",
                GetJointPoint(skeleton.Joints[JointType.ShoulderRight]));
                ElbowRight.Text = string.Format("{0}",
                GetJointPoint(skeleton.Joints[JointType.ElbowRight]));
                WristRight.Text = string.Format("{0}",
                GetJointPoint(skeleton.Joints[JointType.WristRight]));
            }
        }
    }
}

```

```

        HandRight.Text = string.Format("{0}",
GetJointPoint(skeleton.Joints[JointType.HandRight]));

        ShoulderLeft.Text = string.Format("{0}",
GetJointPoint(skeleton.Joints[JointType.ShoulderLeft]));
        ElbowLeft.Text = string.Format("{0}",
GetJointPoint(skeleton.Joints[JointType.ElbowLeft]));
        WristLeft.Text = string.Format("{0}",
GetJointPoint(skeleton.Joints[JointType.WristLeft]));
        HandLeft.Text = string.Format("{0}",
GetJointPoint(skeleton.Joints[JointType.HandLeft]));

        HipRight.Text = string.Format("{0}",
GetJointPoint(skeleton.Joints[JointType.HipRight]));
        KneeRight.Text = string.Format("{0}",
GetJointPoint(skeleton.Joints[JointType.KneeRight]));
        AnkleRight.Text = string.Format("{0}",
GetJointPoint(skeleton.Joints[JointType.AnkleRight]));
        FootRight.Text = string.Format("{0}",
GetJointPoint(skeleton.Joints[JointType.FootRight]));

        HipLeft.Text = string.Format("{0}",
GetJointPoint(skeleton.Joints[JointType.HipLeft]));
        KneeLeft.Text = string.Format("{0}",
GetJointPoint(skeleton.Joints[JointType.KneeLeft]));
        AnkleLeft.Text = string.Format("{0}",
GetJointPoint(skeleton.Joints[JointType.AnkleLeft]));
        FootLeft.Text = string.Format("{0}",
GetJointPoint(skeleton.Joints[JointType.FootLeft]));
    }
}
}

private Polyline CreateFigure(Skeleton skeleton, Brush brush, JointType[] joints)
{
    Polyline figure = new Polyline();
    figure.StrokeThickness = 4;
    figure.Stroke = brush;

    for (int i = 0; i < joints.Length; i++)
    {
        figure.Points.Add(GetJointPoint(skeleton.Joints[joints[i]]));
    }

    return figure;
}

private Point GetJointPoint(Joint joint)
{
    //DepthImagePoint point =
this.KinectDevice.MapSkeletonPointToDepth(joint.Position,
this.KinectDevice.DepthStream.Format);
    ColorImagePoint point =
this.KinectDevice.MapSkeletonPointToColor(joint.Position,
this.KinectDevice.ColorStream.Format);
}

```

```

        point.X *= (int)this.LayoutRootMain.ActualWidth /
this.KinectDevice.ColorStream.FrameWidth;
        point.Y *= (int)this.LayoutRootMain.ActualHeight /
this.KinectDevice.ColorStream.FrameHeight;

        return new Point(point.X, point.Y);
    }

    private void CalibrationButton_Click(object sender, RoutedEventArgs e)
{
    string fileName1 = "Calibre.jpg";
    //string fileName2 = "CalibreDepth.jpg";

    if (File.Exists(fileName1))
    {
        File.Delete(fileName1);
        //File.Delete(fileName2);
    }

    using (FileStream savedSnapshot = new FileStream(fileName1,
 FileMode.CreateNew))
    {
        BitmapSource image = (BitmapSource)ColorImageElement.Source;

        JpegBitmapEncoder jpgEncoder = new JpegBitmapEncoder();
        jpgEncoder.QualityLevel = 70;
        jpgEncoder.Frames.Add(BitmapFrame.Create(image));
        jpgEncoder.Save(savedSnapshot);

        savedSnapshot.Flush();
        savedSnapshot.Close();
        savedSnapshot.Dispose();
    }

    //using (FileStream savedSnapshot = new FileStream(fileName2,
 FileMode.CreateNew))
    //{
    //    BitmapSource image = (BitmapSource)DepthImage.Source;

    //    JpegBitmapEncoder jpgEncoder = new JpegBitmapEncoder();
    //    jpgEncoder.QualityLevel = 70;
    //    jpgEncoder.Frames.Add(BitmapFrame.Create(image));
    //    jpgEncoder.Save(savedSnapshot);

    //    savedSnapshot.Flush();
    //    savedSnapshot.Close();
    //    savedSnapshot.Dispose();
    //}
}

private void Acquisition1Button_Click(object sender, RoutedEventArgs e)
{
    string fileName1 = "Sujet1.jpg";
    string fileNameT = "Joints1.txt";
    //string fileName2 = "SujetDepth.jpg";

    if (File.Exists(fileName1) | File.Exists(fileNameT))
{

```

```

        File.Delete(fileName1);
        File.Delete(fileNameT);
        //File.Delete(fileName2);
    }

    using (FileStream savedSnapshot = new FileStream(fileName1,
 FileMode.CreateNew))
    {
        BitmapSource image = (BitmapSource)ColorImageElement.Source;

        JpegBitmapEncoder jpgEncoder = new JpegBitmapEncoder();
        jpgEncoder.QualityLevel = 70;
        jpgEncoder.Frames.Add(BitmapFrame.Create(image));
        jpgEncoder.Save(savedSnapshot);

        savedSnapshot.Flush();
        savedSnapshot.Close();
        savedSnapshot.Dispose();
    }

    //    using (FileStream savedSnapshot = new FileStream(fileName4,
 FileMode.CreateNew))
    //    {
    //        BitmapSource image = (BitmapSource)DepthImage.Source;

    //        JpegBitmapEncoder jpgEncoder = new JpegBitmapEncoder();
    //        jpgEncoder.QualityLevel = 70;
    //        jpgEncoder.Frames.Add(BitmapFrame.Create(image));
    //        jpgEncoder.Save(savedSnapshot);

    //        savedSnapshot.Flush();
    //        savedSnapshot.Close();
    //        savedSnapshot.Dispose();
    //    }
    using (TextWriter w = File.CreateText(fileNameT))
    {
        w.WriteLine("{0}", Head.Text);
        w.WriteLine("{0}", ShoulderCenter.Text);
        w.WriteLine("{0}", Spine.Text);
        w.WriteLine("{0}", HipCenter.Text);
        w.WriteLine("{0}", ShoulderRight.Text);
        w.WriteLine("{0}", ElbowRight.Text);
        w.WriteLine("{0}", WristRight.Text);
        w.WriteLine("{0}", HandRight.Text);
        w.WriteLine("{0}", ShoulderLeft.Text);
        w.WriteLine("{0}", ElbowLeft.Text);
        w.WriteLine("{0}", WristLeft.Text);
        w.WriteLine("{0}", HandLeft.Text);
        w.WriteLine("{0}", HipRight.Text);
        w.WriteLine("{0}", KneeRight.Text);
        w.WriteLine("{0}", AnkleRight.Text);
        w.WriteLine("{0}", FootRight.Text);
        w.WriteLine("{0}", HipLeft.Text);
        w.WriteLine("{0}", KneeLeft.Text);
        w.WriteLine("{0}", AnkleLeft.Text);
        w.WriteLine("{0}", FootLeft.Text);
    }
}

```

```

private void Acquisition2Button_Click(object sender, RoutedEventArgs e)
{
    string fileName1 = "Sujet2.jpg";
    string fileNameT = "Joints2.txt";
    //string fileName2 = "SujetDepth.jpg";

    if (File.Exists(fileName1) | File.Exists(fileNameT))
    {
        File.Delete(fileName1);
        File.Delete(fileNameT);
        //File.Delete(fileName2);
    }

    using (FileStream savedSnapshot = new FileStream(fileName1,
    FileMode.CreateNew))
    {
        BitmapSource image = (BitmapSource)ColorImageElement.Source;

        JpegBitmapEncoder jpgEncoder = new JpegBitmapEncoder();
        jpgEncoder.QualityLevel = 70;
        jpgEncoder.Frames.Add(BitmapFrame.Create(image));
        jpgEncoder.Save(savedSnapshot);

        savedSnapshot.Flush();
        savedSnapshot.Close();
        savedSnapshot.Dispose();
    }

    using (TextWriter w = File.CreateText(fileNameT))
    {
        w.WriteLine("{0}", Head.Text);
        w.WriteLine("{0}", ShoulderCenter.Text);
        w.WriteLine("{0}", Spine.Text);
        w.WriteLine("{0}", HipCenter.Text);
        w.WriteLine("{0}", ShoulderRight.Text);
        w.WriteLine("{0}", ElbowRight.Text);
        w.WriteLine("{0}", WristRight.Text);
        w.WriteLine("{0}", HandRight.Text);
        w.WriteLine("{0}", ShoulderLeft.Text);
        w.WriteLine("{0}", ElbowLeft.Text);
        w.WriteLine("{0}", WristLeft.Text);
        w.WriteLine("{0}", HandLeft.Text);
        w.WriteLine("{0}", HipRight.Text);
        w.WriteLine("{0}", KneeRight.Text);
        w.WriteLine("{0}", AnkleRight.Text);
        w.WriteLine("{0}", FootRight.Text);
        w.WriteLine("{0}", HipLeft.Text);
        w.WriteLine("{0}", KneeLeft.Text);
        w.WriteLine("{0}", AnkleLeft.Text);
        w.WriteLine("{0}", FootLeft.Text);
    }
}
}
}

```

Code de l'affichage du squelette (Skeleton viewer)

```

/*
 *
 * Copyright (c) 2012 Jarrett Webb & James Ashley
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy of this
software and associated
 * documentation files (the "Software"), to deal in the Software without restriction,
including without limitation
 * the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software,
 * and to permit persons to whom the Software is furnished to do so, subject to the
following conditions:
 *
 * The above copyright notice and this permission notice shall be included in all copies
or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED
 * TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL
 * THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF
 * CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE
OR THE USE OR OTHER DEALINGS
 * IN THE SOFTWARE.
 *
 */

```

```

using System.Windows;
using System.Windows.Controls;
using System.Windows.Media;
using System.Windows.Shapes;

using Microsoft.Kinect;

namespace KinectPhoto
{
    /// <summary>
    /// Interaction logic for SkeletonViewer.xaml
    /// </summary>
    public partial class SkeletonViewer : UserControl
    {
        #region Member Variables
        private const float FeetPerMeters = 3.2808399f;
        private readonly Brush[] _SkeletonBrushes = new Brush[] { Brushes.Black,
Brushes.Crimson, Brushes.Indigo, Brushes.DodgerBlue, Brushes.Purple, Brushes.Pink };
        private Skeleton[] _FrameSkeletons;
        #endregion Member Variables

        #region Constructor
        public SkeletonViewer()
        {

```

```

{
    InitializeComponent();
}
#endregion Constructor

#region Methods
private void KinectDevice_SkeletonFrameReady(object sender,
SkeletonFrameReadyEventArgs e)
{
    SkeletonsPanel.Children.Clear();
    JointInfoPanel.Children.Clear();

    using (SkeletonFrame frame = e.OpenSkeletonFrame())
    {
        if (frame != null)
        {
            if (this.IsEnabled)
            {
                frame.CopySkeletonDataTo(this._FrameSkeletons);

                for (int i = 0; i < this._FrameSkeletons.Length; i++)
                {
                    DrawSkeleton(this._FrameSkeletons[i],
this._SkeletonBrushes[i]);

                    TrackJoint(this._FrameSkeletons[i].Joints[JointType.HandLeft],
this._SkeletonBrushes[i]);

                    TrackJoint(this._FrameSkeletons[i].Joints[JointType.HandRight], this._SkeletonBrushes[i]);
                }
            }
        }
    }
}

private void DrawSkeleton(Skeleton skeleton, Brush brush)
{
    if (skeleton != null && skeleton.TrackingState ==
SkeletonTrackingState.Tracked)
    {
        //Draw head and torso
        Polyline figure = CreateFigure(skeleton, brush, new[] { JointType.Head,
JointType.ShoulderCenter, JointType.ShoulderLeft, JointType.Spine,
JointType.ShoulderRight, JointType.ShoulderCenter, JointType.HipCenter});
        SkeletonsPanel.Children.Add(figure);

        figure = CreateFigure(skeleton, brush, new[] { JointType.HipLeft,
JointType.HipRight });
        SkeletonsPanel.Children.Add(figure);

        //Draw left leg
        figure = CreateFigure(skeleton, brush, new[] { JointType.HipCenter,
JointType.HipLeft, JointType.KneeLeft, JointType.AnkleLeft, JointType.FootLeft });
        SkeletonsPanel.Children.Add(figure);
    }
}

```

```

        //Draw right leg
        figure = CreateFigure(skeleton, brush, new[] { JointType.HipCenter,
JointType.HipRight, JointType.KneeRight, JointType.AnkleRight, JointType.FootRight });
        SkeletonsPanel.Children.Add(figure);

        //Draw left arm
        figure = CreateFigure(skeleton, brush, new[] { JointType.ShoulderLeft,
JointType.ElbowLeft, JointType.WristLeft, JointType.HandLeft });
        SkeletonsPanel.Children.Add(figure);

        //Draw right arm
        figure = CreateFigure(skeleton, brush, new[] { JointType.ShoulderRight,
JointType.ElbowRight, JointType.WristRight, JointType.HandRight });
        SkeletonsPanel.Children.Add(figure);
    }
}

private Polyline CreateFigure(Skeleton skeleton, Brush brush, JointType[] joints)
{
    Polyline figure = new Polyline();

    figure.StrokeThickness = 18;
    figure.Stroke = brush;

    for (int i = 0; i < joints.Length; i++)
    {
        figure.Points.Add(GetJointPoint(skeleton.Joints[joints[i]]));
    }

    return figure;
}

private Point GetJointPoint(Joint joint)
{
    DepthImagePoint point =
this.KinectDevice.MapSkeletonPointToDepth(joint.Position,
this.KinectDevice.DepthStream.Format);
    point.X *= (int)this.LayoutRoot.ActualWidth /
KinectDevice.DepthStream.FrameWidth;
    point.Y *= (int)this.LayoutRoot.ActualHeight /
KinectDevice.DepthStream.FrameHeight;

    return new Point(point.X, point.Y);
}

private void TrackJoint(Joint joint, Brush brush)
{
    if (joint.TrackingState != JointTrackingState.NotTracked)
    {
        Canvas container = new Canvas();
        Point jointPoint = GetJointPoint(joint);

        //FeetPerMeters is a class constant of 3.2808399f;
        double z = joint.Position.Z * FeetPerMeters;
    }
}

```

```

        Ellipse element = new Ellipse();
        element.Height = 15;
        element.Width = 15;
        element.Fill = brush;
        Canvas.SetLeft(element, 0 - (element.Width / 2));
        Canvas.SetTop(element, 0 - (element.Height / 2));
        container.Children.Add(element);

        TextBlock positionText = new TextBlock();
        positionText.Text = string.Format("<{0:0.00}, {1:0.00}, {2:0.00}>",
jointPoint.X, jointPoint.Y, z);
        positionText.Foreground = brush;
        positionText.FontSize = 24;
        positionText.Measure(new Size(double.PositiveInfinity,
double.PositiveInfinity));
        Canvas.SetLeft(positionText, 35);
        Canvas.SetTop(positionText, 15);
        container.Children.Add(positionText);

        Canvas.SetLeft(container, jointPoint.X);
        Canvas.SetTop(container, jointPoint.Y);

        JointInfoPanel.Children.Add(container);
    }
}

#endregion Methods

#region Properties
#region KinectDevice
protected const string KinectDevicePropertyName = "KinectDevice";
public static readonly DependencyProperty KinectDeviceProperty =
DependencyProperty.Register(KinectDevicePropertyName, typeof(KinectSensor),
typeof(SkeletonViewer), new PropertyMetadata(null, KinectDeviceChanged));

private static void KinectDeviceChanged(DependencyObject owner,
DependencyPropertyChangedEventArgs e)
{
    SkeletonViewer viewer = (SkeletonViewer)owner;

    if (e.OldValue != null)
    {
        ((KinectSensor)e.OldValue).SkeletonFrameReady -=
viewer.KinectDevice_SkeletonFrameReady;
        viewer._FrameSkeletons = null;
    }

    if (e.NewValue != null)
    {
        viewer.KinectDevice = (KinectSensor)e.NewValue;
        viewer.KinectDevice.SkeletonFrameReady +=
viewer.KinectDevice_SkeletonFrameReady;
        viewer._FrameSkeletons = new
Skeleton[viewer.KinectDevice.SkeletonStream.FrameSkeletonArrayLength];
    }
}

```

```
public KinectSensor KinectDevice
{
    get { return (KinectSensor)GetValue(KinectDeviceProperty); }
    set { SetValue(KinectDeviceProperty, value); }
}
#endregion KinectDevice
#endregion Properties
}
```

Code de l'interface graphique (Main Windows.xaml)

```

<Window x:Class="KinectPhoto.MainWindow"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        Title="MainWindow" Height="700" Width="1280">
    <Grid Background="GhostWhite">
        <StackPanel Orientation="Vertical" HorizontalAlignment="Center">
            <StackPanel Orientation="Horizontal">
                <Image x:Name="ColorImageElement" Height="480" Width="640"/>
                <Image x:Name="ColorImageElement2" Height="480" Width="640"/>
                <Grid x:Name="LayoutRootMain" Height="480" Width="640">
                    <Grid x:Name="LayoutRoot" Margin="-1280,0,1280,0" />
                </Grid>
            </StackPanel>
            <Grid Height="10"/>
            <StackPanel Orientation="Horizontal" HorizontalAlignment="Center">
                <Button Content="Calibration" Click="CalibrationButton_Click"
Width="200"/>
                <Grid Width="100"/>
                <Button Content="Vue 1" Click="Acquisition1Button_Click" Width="200"/>
                <Grid Width="100"/>
                <Button Content="Vue 2" Click="Acquisition2Button_Click" Width="200"/>
            </StackPanel>
            <Grid Height="10"/>
            <StackPanel Orientation="Horizontal" HorizontalAlignment="Center">
                <StackPanel Orientation="Vertical" HorizontalAlignment="Center">
                    <TextBlock Text="Epaule gauche:" HorizontalAlignment="Left"/>
                    <TextBlock x:Name="ShoulderLeft" Width="125"
HorizontalAlignment="Left" TextAlignment="Left"/>
                    <TextBlock Text="Coude gauche:" HorizontalAlignment="Left"/>
                    <TextBlock x:Name="ElbowLeft" Width="125" HorizontalAlignment="Left"
TextAlignment="Left"/>
                    <TextBlock Text="Poignet gauche:" HorizontalAlignment="Left"/>
                    <TextBlock x:Name="WristLeft" Width="125" HorizontalAlignment="Left"
TextAlignment="Left"/>
                    <TextBlock Text="Main gauche:" HorizontalAlignment="Left"/>
                    <TextBlock x:Name="HandLeft" Width="125" HorizontalAlignment="Left"
TextAlignment="Left"/>
                </StackPanel>
                <Grid Width="50"/>
                <StackPanel Orientation="Vertical" HorizontalAlignment="Center">
                    <TextBlock Text="Hanche gauche:" HorizontalAlignment="Left"/>
                    <TextBlock x:Name="HipLeft" Width="125" HorizontalAlignment="Left"
TextAlignment="Left"/>
                    <TextBlock Text="Genou gauche:" HorizontalAlignment="Left"/>
                    <TextBlock x:Name="KneeLeft" Width="125" HorizontalAlignment="Left"
TextAlignment="Left"/>
                    <TextBlock Text="Mollet gauche:" HorizontalAlignment="Left"/>
                    <TextBlock x:Name="AnkleLeft" Width="125" HorizontalAlignment="Left"
TextAlignment="Left"/>
                    <TextBlock Text="Pied gauche:" HorizontalAlignment="Left"/>
                    <TextBlock x:Name="FootLeft" Width="125" HorizontalAlignment="Left"
TextAlignment="Left"/>
                </StackPanel>
                <Grid Width="50"/>
                <StackPanel Orientation="Vertical" HorizontalAlignment="Center">

```

```

        <TextBlock Text="Tete:" HorizontalAlignment="Center"/>
        <TextBlock x:Name="Head" Width="125" HorizontalAlignment="Center"
TextAlignment="Center"/>
            <TextBlock Text="Epaule Centre:" HorizontalAlignment="Center"/>
            <TextBlock x:Name="ShoulderCenter" Width="125"
HorizontalAlignment="Center" TextAlignment="Center"/>
            <TextBlock Text="Vertebre:" HorizontalAlignment="Center"/>
            <TextBlock x:Name="Spine" Width="125" HorizontalAlignment="Center"
TextAlignment="Center"/>
            <TextBlock Text="Hanche Centre:" HorizontalAlignment="Center"/>
            <TextBlock x:Name="HipCenter" Width="125" HorizontalAlignment="Center"
TextAlignment="Center"/>
        </StackPanel>
        <Grid Width="50"/>
        <StackPanel Orientation="Vertical" HorizontalAlignment="Center">
            <TextBlock Text="Hanche droite:" HorizontalAlignment="Right"/>
            <TextBlock x:Name="HipRight" Width="125" HorizontalAlignment="Right"
TextAlignment="Right"/>
            <TextBlock Text="Genou droit:" HorizontalAlignment="Right"/>
            <TextBlock x:Name="KneeRight" Width="125" HorizontalAlignment="Right"
TextAlignment="Right"/>
            <TextBlock Text="Mollet droit:" HorizontalAlignment="Right"/>
            <TextBlock x:Name="AnkleRight" Width="125" HorizontalAlignment="Right"
TextAlignment="Right"/>
            <TextBlock Text="Pied droit:" HorizontalAlignment="Right"/>
            <TextBlock x:Name="FootRight" Width="125" HorizontalAlignment="Right"
TextAlignment="Right"/>
        </StackPanel>
        <Grid Width="50"/>
        <StackPanel Orientation="Vertical" HorizontalAlignment="Center">
            <TextBlock Text="Epaule droite:" HorizontalAlignment="Right"/>
            <TextBlock x:Name="ShoulderRight" Width="125"
HorizontalAlignment="Right" TextAlignment="Right"/>
            <TextBlock Text="Coude droit:" HorizontalAlignment="Right"/>
            <TextBlock x:Name="ElbowRight" Width="125" HorizontalAlignment="Right"
TextAlignment="Right"/>
            <TextBlock Text="Poignet droit:" HorizontalAlignment="Right"/>
            <TextBlock x:Name="WristRight" Width="125" HorizontalAlignment="Right"
TextAlignment="Right"/>
            <TextBlock Text="Main droite:" HorizontalAlignment="Right"/>
            <TextBlock x:Name="HandRight" Width="125" HorizontalAlignment="Right"
TextAlignment="Right"/>
        </StackPanel>
    </StackPanel>
</Grid>
</Window>

```

ANNEXE C – Code Matlab

Calibration

```

function [Ratio] = Calibration(nomFichier, dCalibre)

%=====
%FONCTION PERMETTANT DE SEGMENTER L'IMAGE DU CALIBRE ET DE CALIBRER LES
%MESURES AVEC LE RATIO CM/PIXELS.
%nomFichier doit inclure l'extension, le tout entre '' (ex : 'Calibre.jpg').
%dCalibre represente le diametre reel du calibre, en cm.
%=====

imageOpen = fopen(nomFichier);

if imageOpen == -1
    Ratio = [];
    disp('L''image permettant de calibrer les mesures n''a pas été ouverte!');
    disp('Assurez-vous que l''image en question est présente!');
    disp(' ');
else
    disp('L''image permettant de calibrer les mesures a été ouverte avec
succès!');
    disp(' ');
    imageCouleur = imread(nomFichier);
    BW = rgb2gray(imageCouleur);
    close all;

    %SEGMENTATION DU CALIBRE :

    %1:Detecter le calibre :
    [junk threshold] = edge(BW, 'sobel');
    fudgeFactor = .5;
    BW1 = edge(BW, 'sobel', threshold * fudgeFactor);
    %figure, imshow(BW1), title('Détection du calibre');

    %2:Dilater l'image :
    se90 = strel('line', 3, 90);
    se0 = strel('line', 3, 0);

    BW2 = imdilate(BW1, [se90 se0]);
    %figure, imshow(BW2), title('Détection dilatée');

    %3:Remplir le calibre et enlever des objets reliés aux bordures :
    BW3 = imfill(BW2, 'holes');
    %figure, imshow(BW3), title('Remplissage du calibre');
    BWnobord = imclearborder(BW3, 4);
    %figure, imshow(BWnobord), title('cleared border image');

    %4:Lisser le corps :
    seD = strel('diamond',1);
    BW4 = imerode(BWnobord,seD);
    BW4 = imerode(BW4,seD);

```

```
BW4 = imerode(BW4,seD); %Erosion supplémentaire
BW4 = imerode(BW4,seD); %Erosion supplémentaire
BW4 = imerode(BW4,seD); %Erosion supplémentaire
correction = 3; %3 Erosion supplémentaires
figure, imshow(BW4), title('Calibre segmenté');
%imtool(BW4);

%COMPTER LES PIXELS SELON LES LIGNES :
for ii = 1:size(BW4,1)
    L(ii) = sum(BW4(ii,:))+2*correction;
end

%COMPTER LES PIXELS SELON LES COLONNES :
for jj = 1:size(BW4,2)
    C(jj) = sum(BW4(:,jj))+2*correction;
end

%CALCUL DU RATIO DE CALIBRATION
dPixels = (max(L)+max(C))/2;
Ratio = dCalibre/dPixels;

%AFFICHAGE DU RESULTATS :
disp('Calibration réussie!');
fprintf('Le ratio est de %7.5f cm/pixel pour une résolution de %d x
%d!\n', Ratio, size(BW4,2),size(BW4,1));
disp(' ');
end

end
```