

Workflow d'animation 3D

Animation 3D à base de capture de mouvements

Pierre Jaffuer



Faculté des Sciences
Département de Mathématiques et Informatique
Université de La Réunion

La Réunion
3 décembre 2021



CC BY-SA 4.0

Table des matières

1	Prérequis	2
1.1	Ressources	2
1.2	Logiciel utilisé	2
1.3	Installation des add-ons	2
1.4	Quelques connaissances sur les armatures	3
1.4.1	Création de l'armature	5
1.4.2	Déformation du modèle par l'armature	9
1.4.3	Orientation des os	16
1.5	Quelques connaissances sur les contraintes	17
1.5.1	Contraintes d'objets	17
1.5.2	Contraintes d'os	18
2	Transformer les données de capture en une armature	19
2.1	Importation des données	19
2.2	Positionnement des joints	21
2.3	Création de l'armature	21
2.3.1	Convention pour l'orientation des os	22
2.4	Contraindre l'armature au mouvement	24
2.4.1	Contraintes principales des os	24
2.4.2	Correction des rotations	25
2.5	Exporter le résultat	26
2.5.1	Pré-calculer l'animation	27
2.5.2	Nettoyer l'armature	29
2.5.3	Export en BVH	30
3	Retargeting manuel	31
3.1	Contraintes à appliquer	32
3.1.1	Spécificité de l'os racine	33
4	Retargeting assisté	33
4.1	Présentation de l'add-on	33
4.2	Utilisation	35
4.2.1	Configurer l'association des os	36
4.2.2	Sauvegarder la configuration	39
4.2.3	Ajustements du retargeting	39
4.3	Pré-calculer l'animation	41
5	Démonstration du workflow	41



1 Prérequis

1.1 Ressources

Cette section regroupe les ressources à télécharger pour pouvoir suivre ce workflow :

- io_anim_c3d add-on : https://github.com/MattiasFredriksson/io_anim_c3d
- SimpleRetargeting add-on : <https://github.com/smallcluster/TER-animation-3d/releases/tag/v0.0.2>
- exemples : <https://github.com/smallcluster/TER-animation-3d/tree/main/workflow>
- Blender : <https://www.blender.org/>

1.2 Logiciel utilisé

Il existe de nombreux logiciels pour l'animation 3D, par exemple, *Maya* d' *Autodesk* qui est un des standards de l'industrie. Cependant, ces logiciels ont un coût élevé. De plus, les étudiants susceptibles de suivre ce workflow n'ont pas été formés sur ces derniers. En effet, ces logiciels sont souvent lourds et il est parfois difficile d'obtenir à temps une licence étudiante. Finalement, les étudiants concernés par ce workflow ont déjà reçu une formation sur *Blender* (*Figure 1*), logiciel 3D libre et open source qui a récemment percé dans le monde professionnel.

Ce workflow utilisera donc *Blender* en sa version v2.8 au minimum. Il est attendu du lecteur d'avoir au moins un niveau débutant avec ce logiciel afin de pouvoir suivre ce document. Un rappel des connaissances nécessaires à ce workflow seront introduites dans les sections *1.4 Quelques connaissances sur les armatures* et *1.5.1 Quelques connaissances sur les contraintes*.



FIGURE 1 – Blender : <https://www.blender.org/>

1.3 Installation des add-ons

Une fois Blender d'installé, il faudra y ajouter comme indiqué sur la *Figure 2* les deux add-ons de la section *1.1 Ressources* :

- io_anim_c3d : un importateur de fichier C3D par *Mattias Fredriksson*
- SimpleRetargeting : un add-on pour faciliter le retargeting



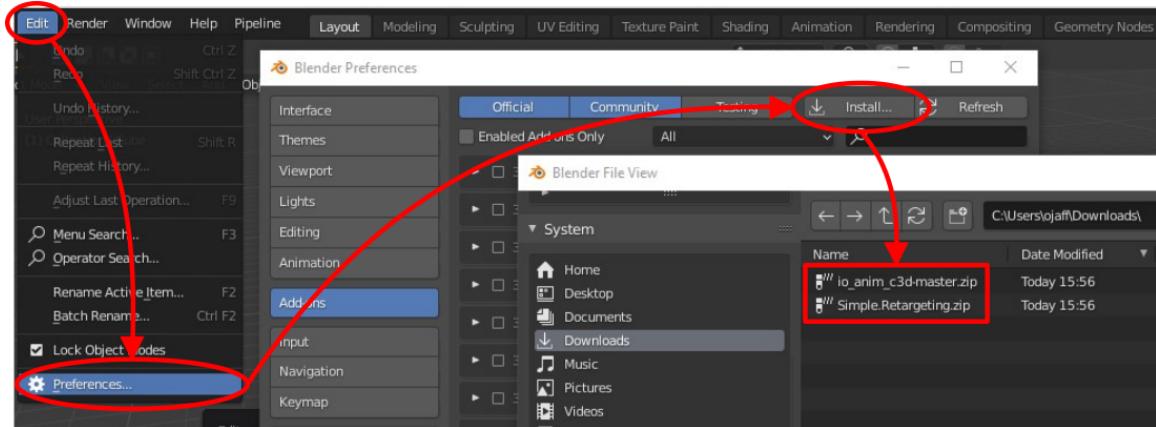
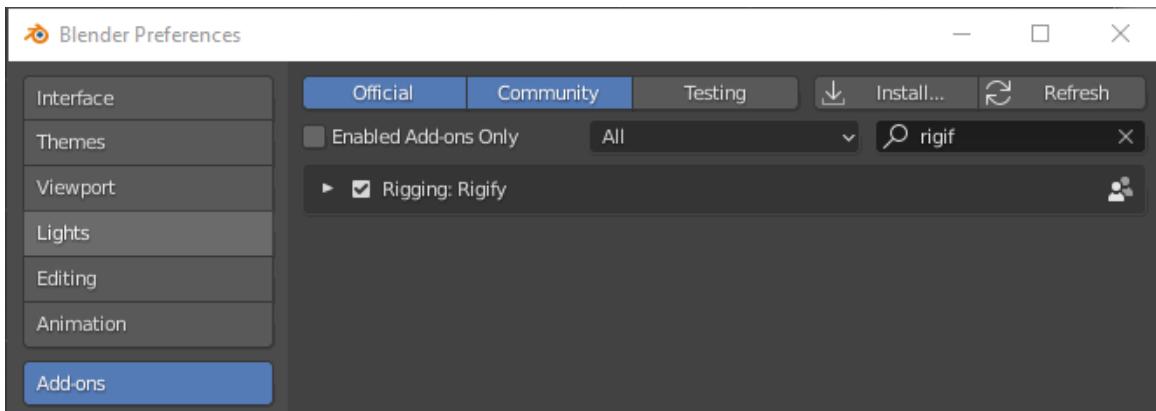


FIGURE 2 – Installation des add-ons

Activer ensuite l'add-on *Rigify* (Figure 3), cet add-on propose des armatures préconfigurées pour des êtres humains. Ce workflow utilisera les mêmes conventions que *Rigify* pour l'orientation des os lors de la section [2.3.1 Convention pour l'orientation des os](#)

FIGURE 3 – Activation de *Rigify*

1.4 Quelques connaissances sur les armatures

En réalité, un animateur n'anime pas à proprement parler le modèle 3D d'un personnage, il anime son armature. Une armature (aussi appelée squelette), permet donc de déformer un modèle 3D en contrôlant ses os (rotations, positions, tailles, étirements, etc.). Ce document propose d'effectuer un rappel sur les armatures de Blender par l'exemple : créer l'armature d'un personnage 3D.

Le fichier Blender *PersonnageExemple.blend* disponible dans les exemples de la section [1.1 Resources](#) contient un personnage 3D visible en Figure 4, qui servira de base pour la suite de ce rappel. Cet exemple contient aussi le résultat final stocké dans la collection *Réultat* afin que le lecteur puisse y comparer son travail avec ce qui est attendu.





FIGURE 4 – Personnage 3D pour cet exemple

1.4.1 Création de l'armature

Une fois le fichier d'exemple téléchargé, puis ouvert dans Blender, il faudra commencer par ajouter une armature. Dans Blender, une armature est un type d'objet, elle peut être créée à partir d'un modèle de base depuis le menu d'ajout. Choisissez l'option *Single Bone* (*Figure 5*).

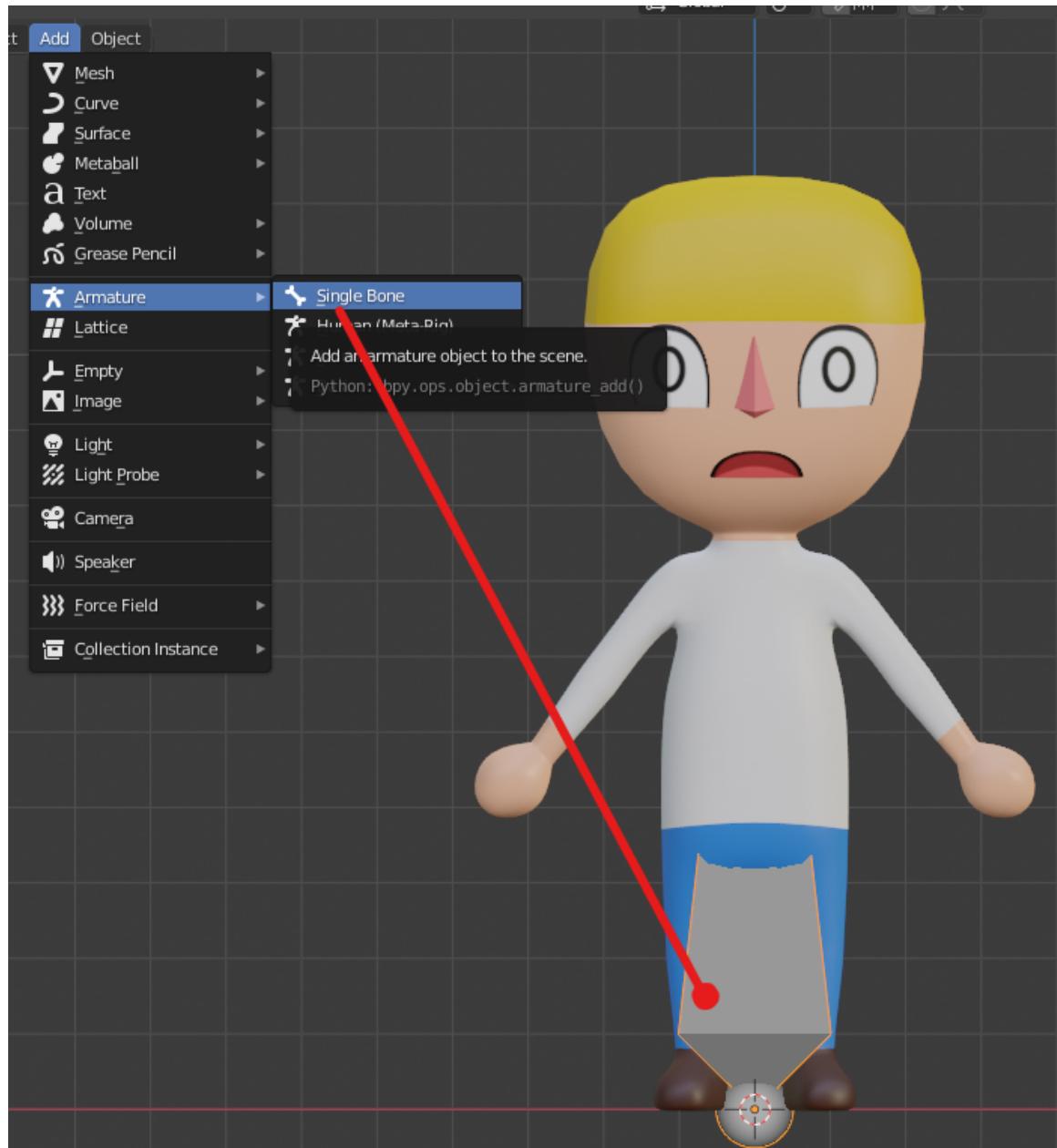


FIGURE 5 – Ajout d'une armature



Afin d'améliorer la visibilité de l'armature, il est possible de l'afficher au premier plan après sélection de celle-ci comme indiqué sur la *Figure 6*.

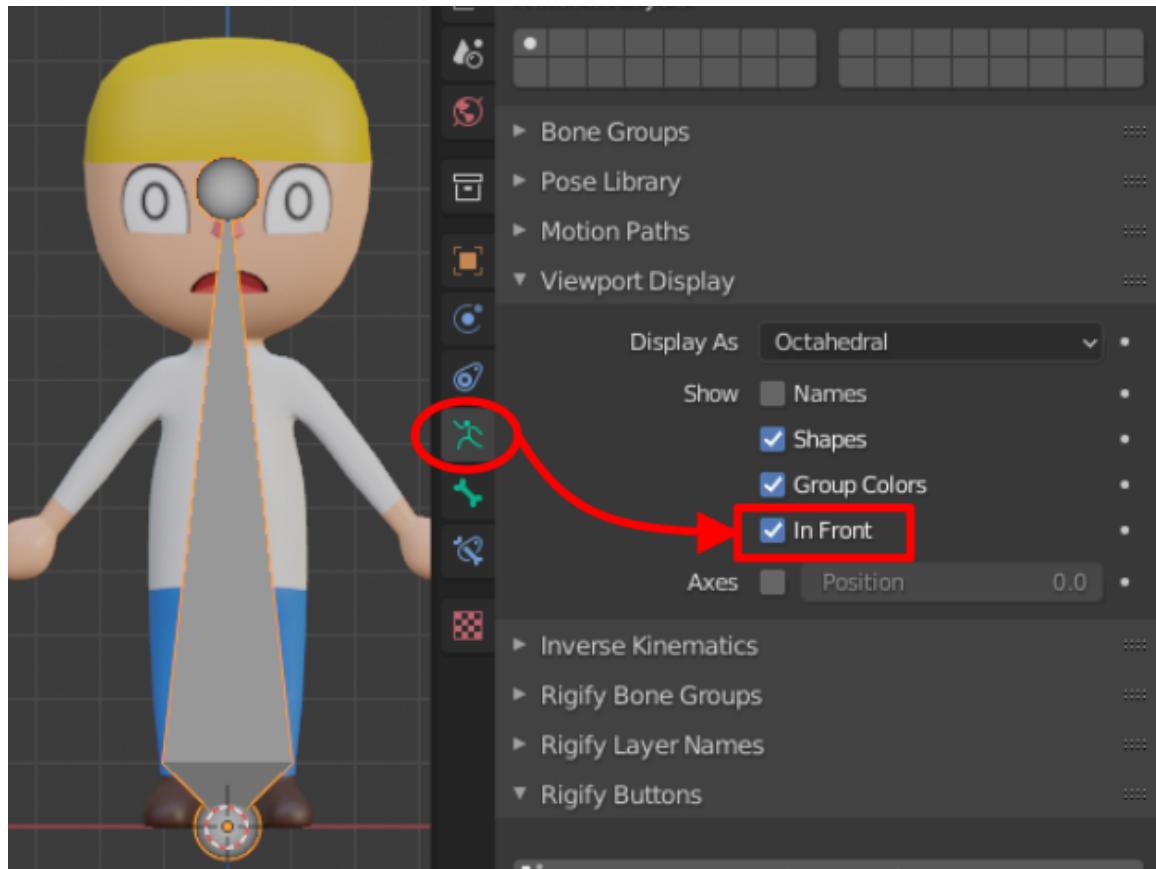


FIGURE 6 – Armature au premier plan

Une armature est une hiérarchie d'os, ses os peuvent être connectés entre eux ou avoir un parentage en conservant une distance fixe. Ce dernier cas sera utilisé pour inclure les membres dans la structure totale. La modification de l'armature se fait en mode *Edit*, les os ont des extrémités sélectionnables et déplaçables, la partie pointue représente la tête de l'os. Pour créer une chaîne d'os, il suffit de sélectionner la tête et de l'extruder (touche *E* par défaut). Il est aussi possible d'ajouter un os libre depuis le menu d'ajout.

Passez maintenant en mode *Edit* et procédez à la création des os en respectant la hiérarchie indiquée sur la *Figure 7*. N'oubliez pas de nommer vos os (*Figure 8*) et d'y ajouter le suffixe *.L* pour les membres de gauche (*.R* pour ceux de droite), ceci est important pour pouvoir effectuer la symétrie en automatique, et donc de ne créer que la partie gauche (ou droite) de l'armature.

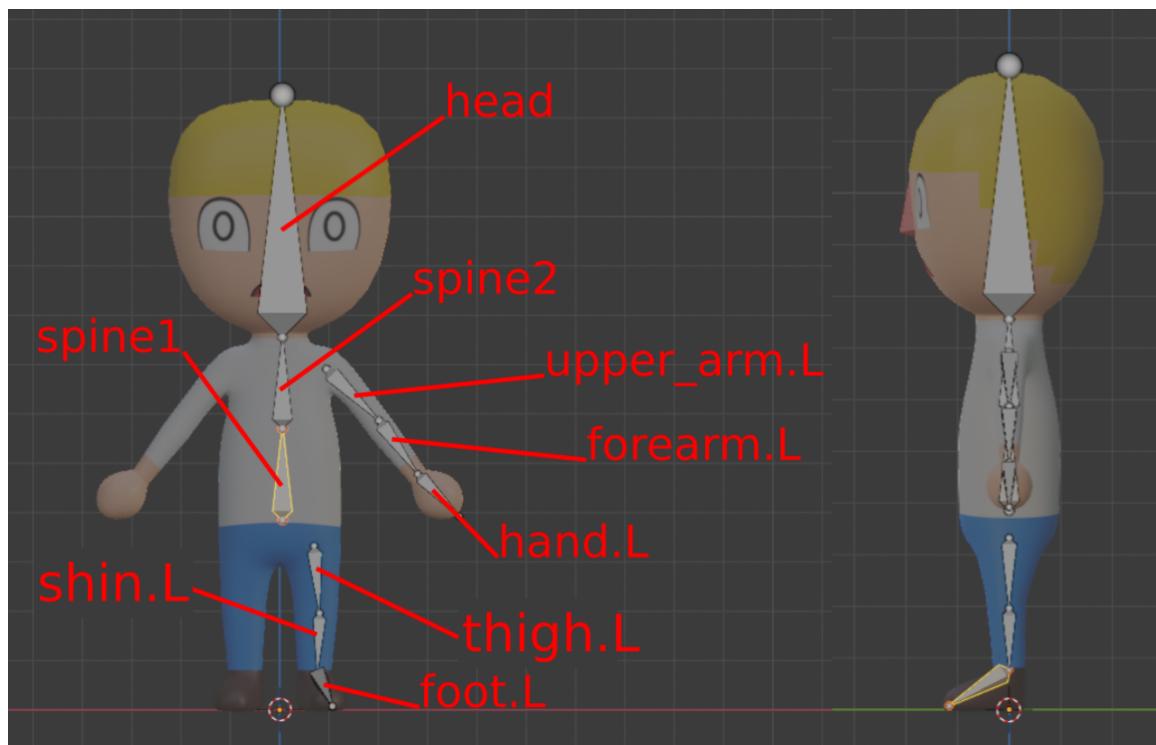


FIGURE 7 – Colonne, bras et jambe gauche

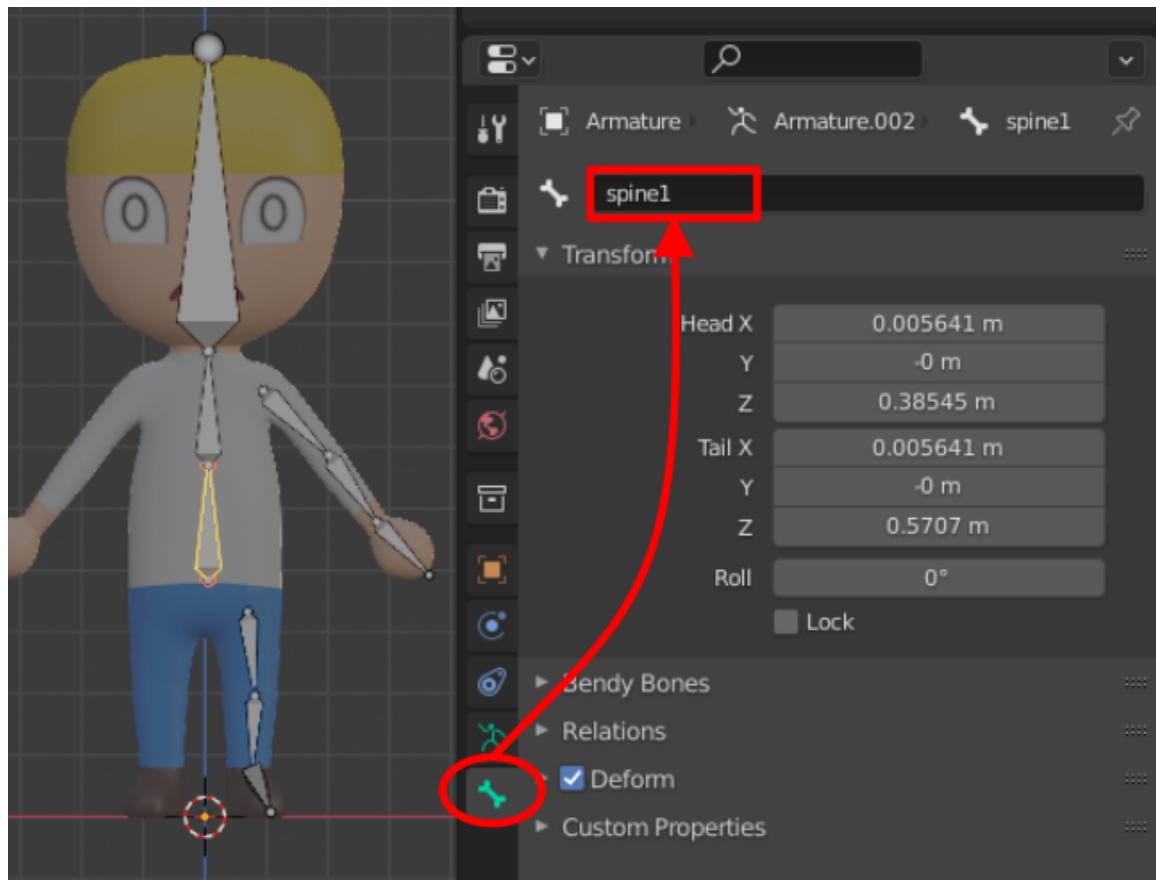


FIGURE 8 – Nommer un os

Il est possible de remarquer la présence de 3 hiérarchies indépendantes :

- celle de la colonne vertébrale
- celle du bras gauche
- celle de la jambe gauche

Pourtant, notre but est de structurer notre armature en une seule hiérarchie, en effet, la jambe se connecte à la hanche, le bras à l'épaule. Pour simuler ces liens, il suffit de parenter 2 os en conservant la distance comme indiqué sur la figure *Figure 9* (*Ctrl+P* avec les 2 os sélectionnés). Une fois le bras et la jambe parentés, il est maintenant possible d'effectuer la symétrie depuis le menu *Armature* avec tous les os sélectionnés.



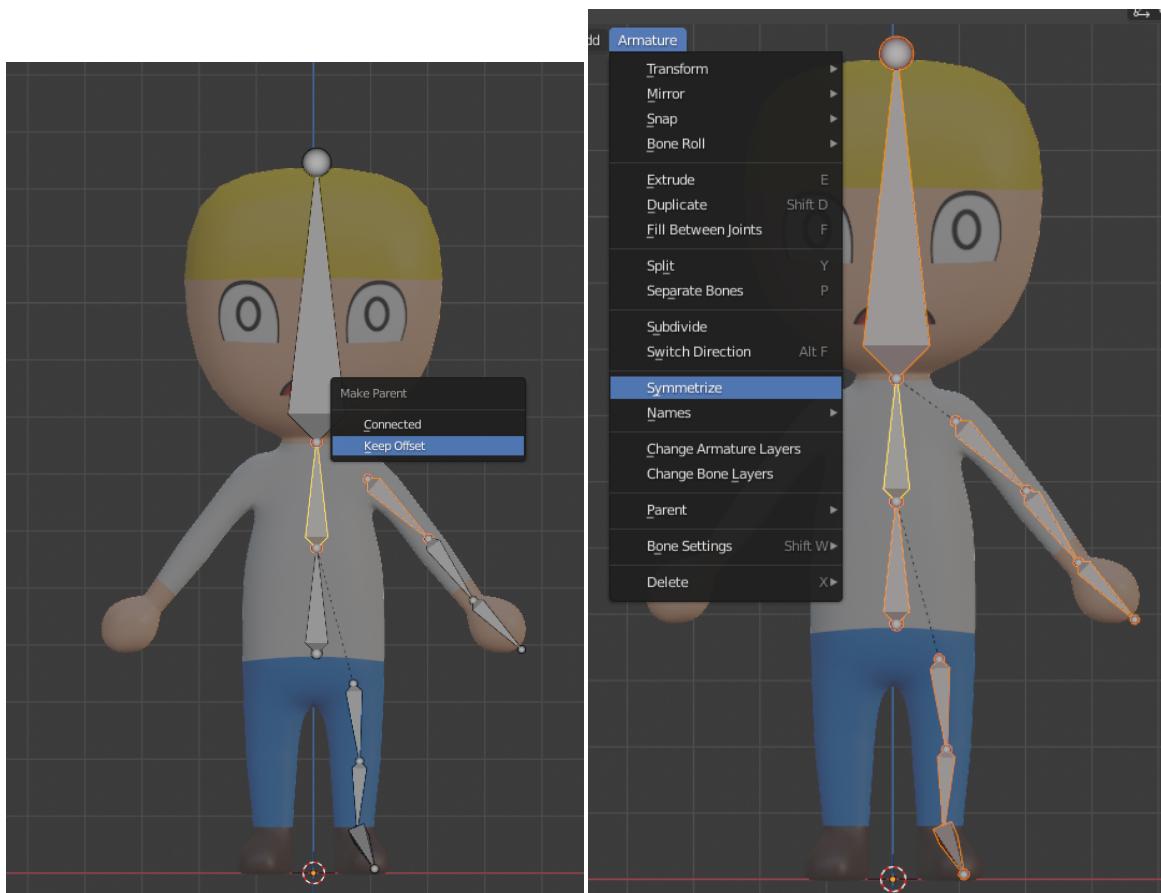


FIGURE 9 – Parentage de 2 os et symétrisation

1.4.2 Déformation du modèle par l'armature

Vous devriez maintenant avoir une armature complète du personnage, mais comment déformer celui-ci grâce à cette armature ? Un modèle 3D n'est qu'un ensemble de sommets connectés entre eux pour former un maillage (*Figure 10*).

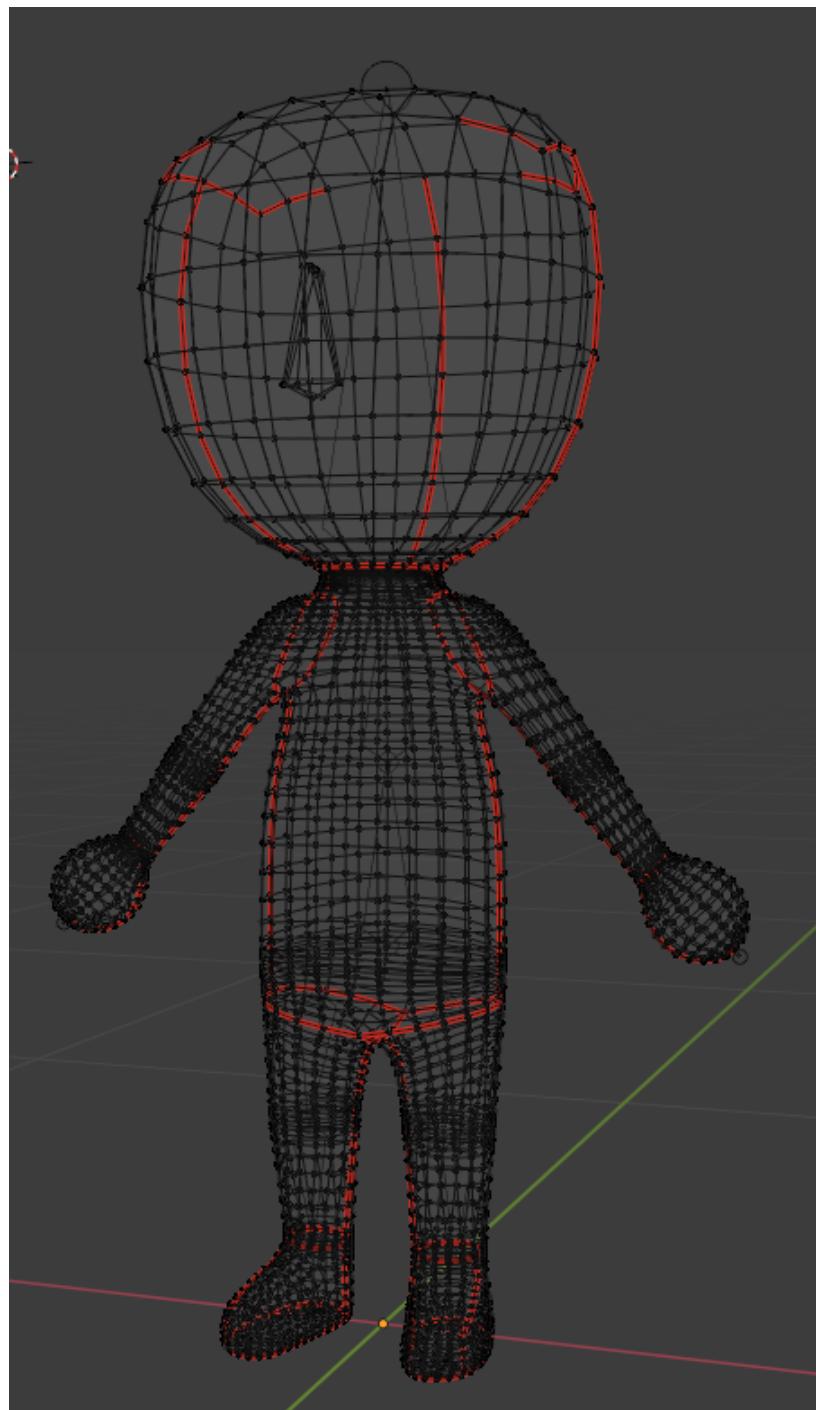


FIGURE 10 – Maillage

Pour déformer ce maillage avec une armature, il faut affecter à chaque sommet du modèle 3D un



poids pour chaque os de l'armature. Affecter manuellement les poids est une tâche très chronophage, ainsi, il existe des systèmes pour attribuer les poids en automatique. Pour déterminer en automatique les poids et ainsi déformer le modèle 3D, il suffit en mode *Object*, de sélectionner dans cet ordre, le modèle du personnage et l'armature créée et de parenter le modèle grâce au raccourci *Ctrl+P* en choisissant l'option indiquée sur la *Figure 11*.

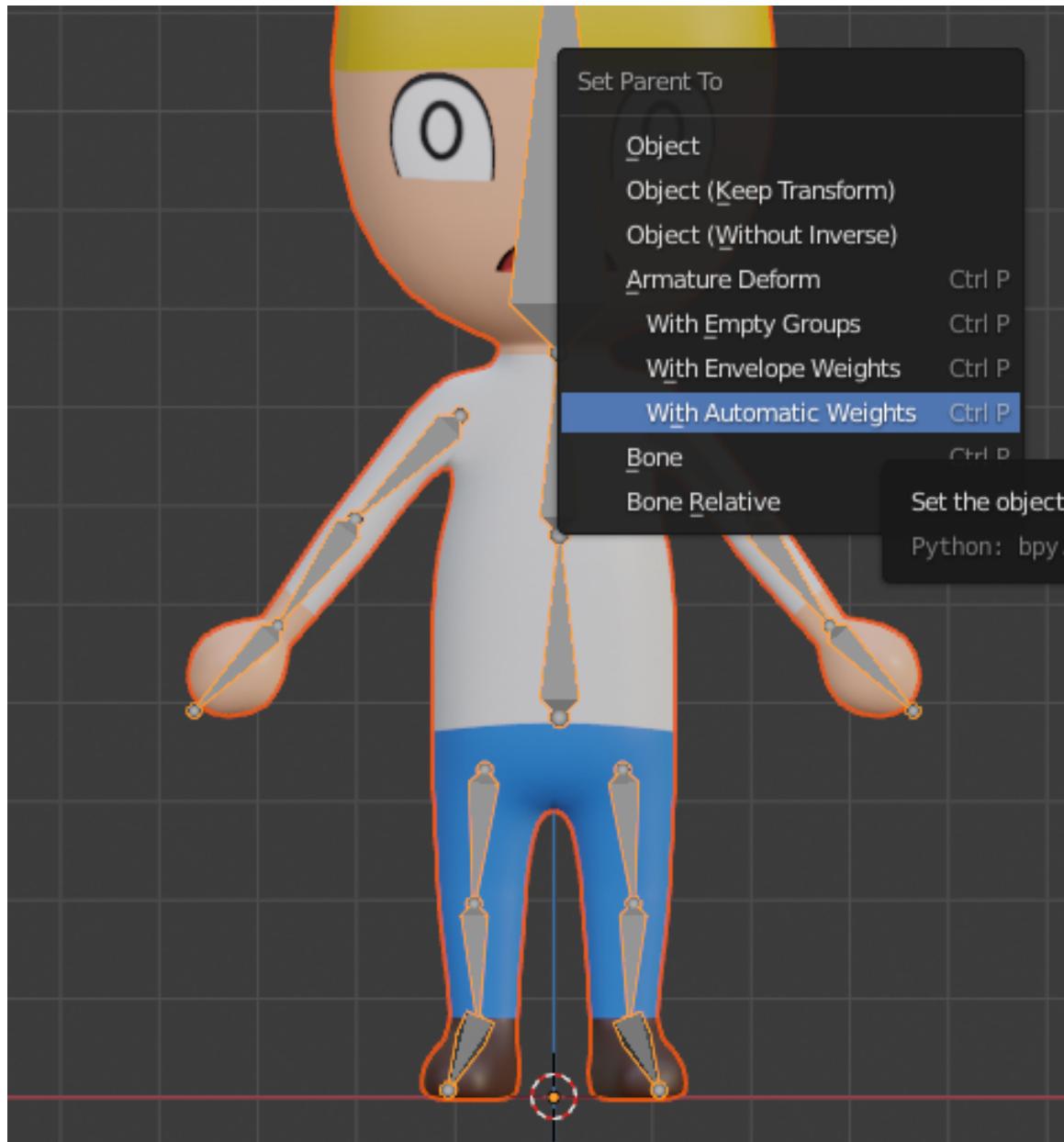


FIGURE 11 – Crédit automatique des poids



Pour tester la déformation, l'armature doit être en mode *Pose*, dans ce mode, les os peuvent être réorientés et les os en début de chaîne peuvent être déplacés. Vous remarquerez que certaines déformations sont incorrectes, mais aussi que certaines parties du modèle ne sont pas influencées par l'armature. L'assignation automatique des poids n'est pas sans failles et c'est à l'utilisateur de corriger la distribution. La [Figure 12](#) montre le résultat d'une mauvaise distribution.

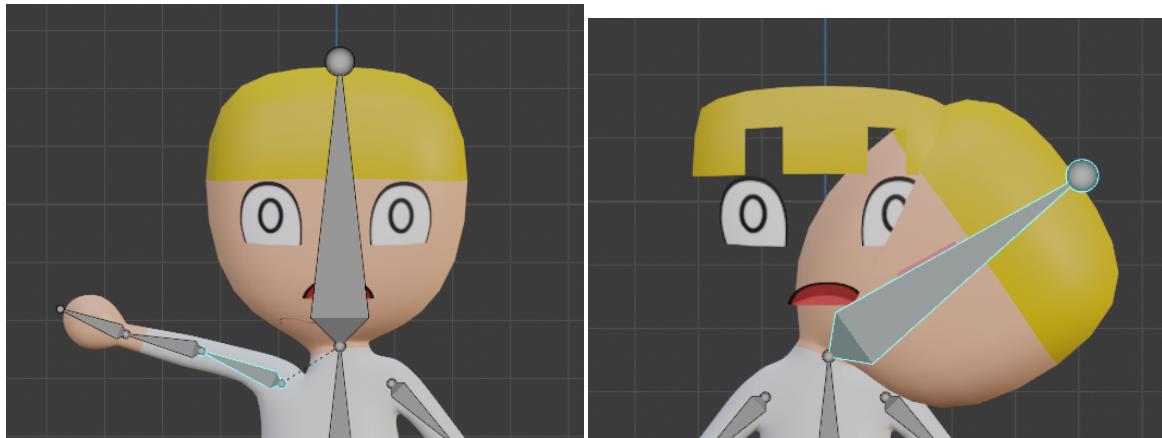


FIGURE 12 – Problèmes de déformations

Dans cet exemple, nous allons corriger la déformation des épaules et rattacher le visage à l'os de la tête. Commençons par les épaules, passez l'armature en mode *Object* et passer le modèle 3D en mode *Weight Paint* en ayant précédemment sélectionné l'armature. Il vous sera possible de sélectionner les os pour voir leur influence sur le maillage, aucune influence pour le bleu et rouge pour une influence complète ([Figure 13](#)). Activer la symétrie sur l'axe X et passer le pinceau en mode *Subtract*, pour l'os de l'épaule, réduisez l'influence de celui-ci pour obtenir une déformation plus naturelle ([Figure 14](#)).

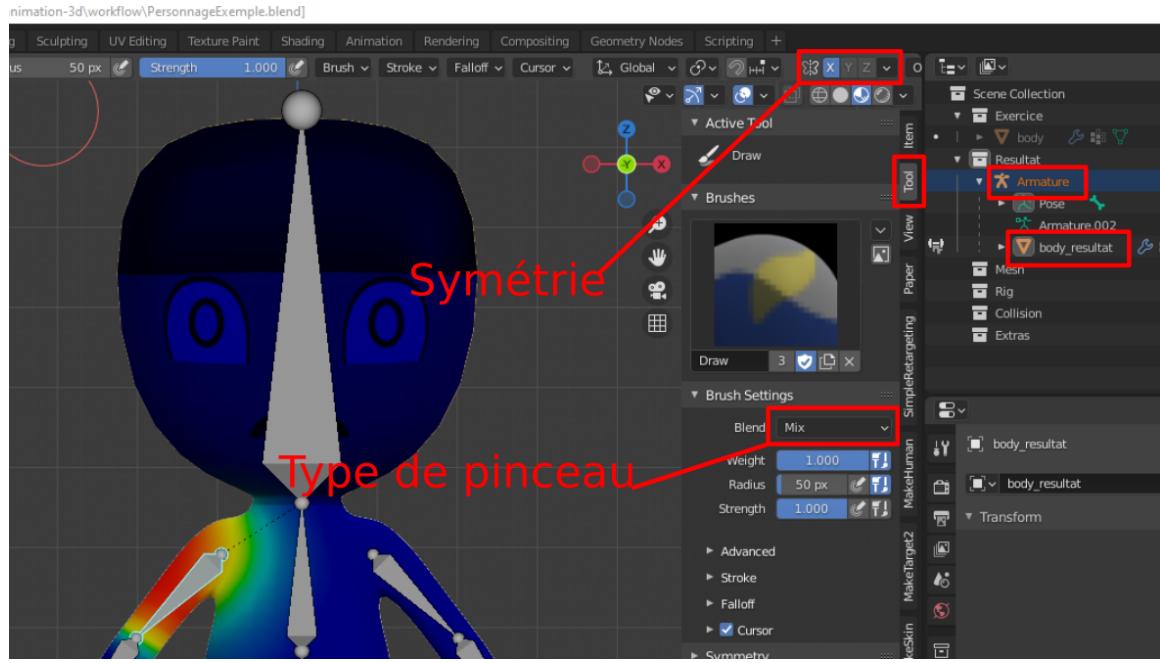


FIGURE 13 – Peinture des poids

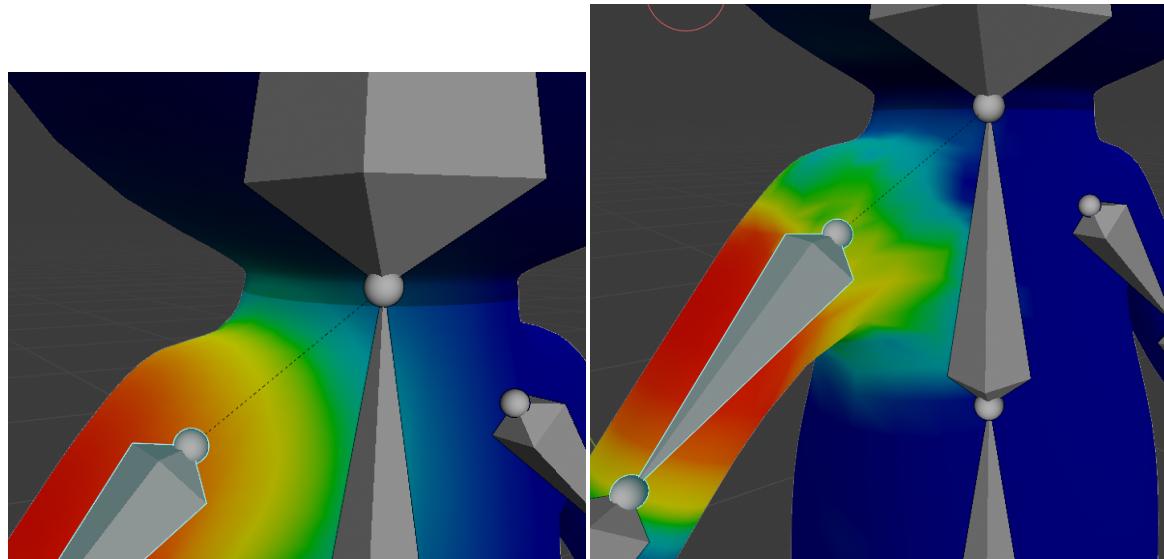


FIGURE 14 – Avant et après modification

Avec l'affichage en mode *Solide* et l'os de la tête sélectionné, vous verrez que le visage est n'est pas du tout affecté par la tête, passez donc le pinceau en mode *Add* avec un poids de 1.0 (*Weight*) et modifier les poids comme visible en [Figure 15](#). Comme pour l'épaule, supprimez l'influence de la



seconde vertèbre pour la tête.

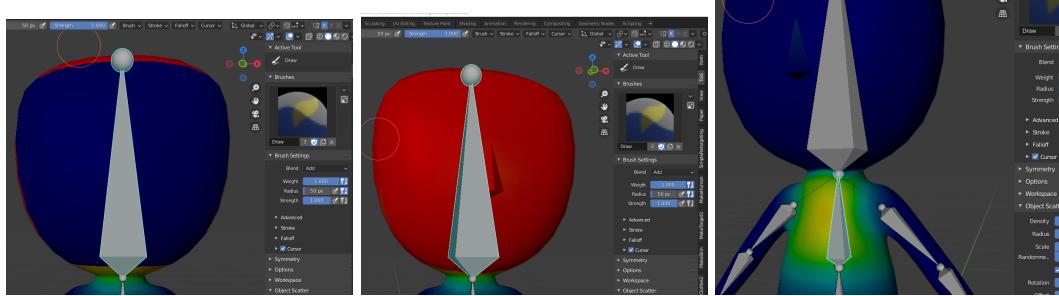


FIGURE 15 – Avant (première image) et après modification

Le personnage peut maintenant être correctement déformé par l'armature. L'animation de celui-ci reviendra à animer son armature.





FIGURE 16 – Personnage animé



1.4.3 Orientation des os

Chaque os possède son propre repère local, pour la suite de ce workflow il est important de comprendre ce concept pour éviter de tomber dans le piège où il est impossible de copier la rotation entre 2 os différents, ce qui est nécessaire pour la solution de retargeting proposée dans ce document. Pour voir les repères des os sur une armature, il faudra procéder comme indiqué sur la [Figure 17](#).

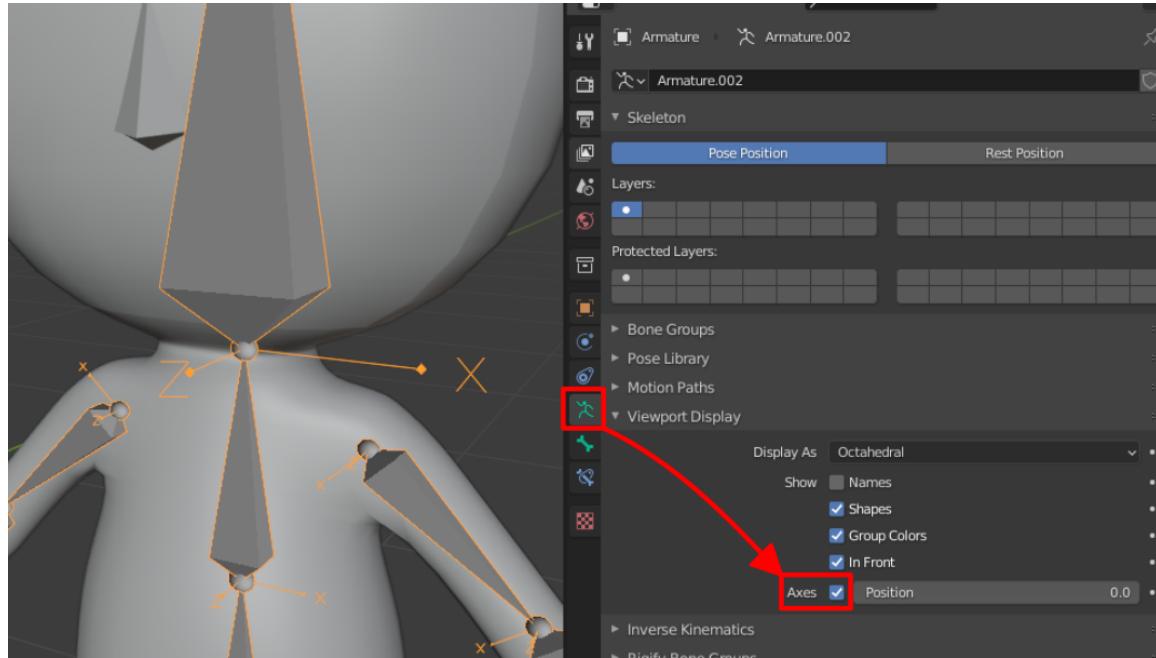
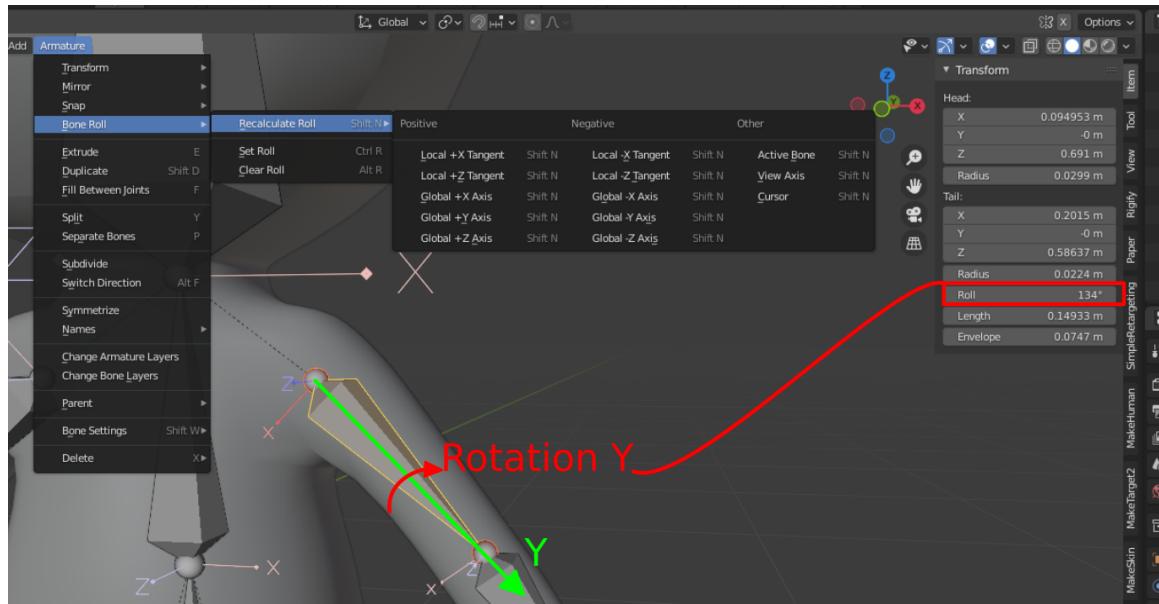


FIGURE 17 – Affichage des repères des os

L'axe de rotation principal d'un os est son axe local *Y*, il est parfois nécessaire d'effectuer une rotation selon cet axe pour éviter les problèmes de retargeting **dès la création de l'armature**. En mode *Edit*, il est possible de modifier l'orientation du repère local d'un os selon son axe *Y* grâce à l'option *Roll* comme visible en [Figure 18](#) ou en utilisant le menu *Recalculate Roll* pour modifier plusieurs os à la fois.

FIGURE 18 – Orientation d'un repère local selon son axe *Y*

1.5 Quelques connaissances sur les contraintes

Cette partie a pour but de rappeler comment accéder au menu des contraintes de *Blender*.

1.5.1 Contraintes d'objets

Les contraintes d'objets sont disponibles (après sélection d'un objet) en mode *Object* dans le menu indiqué sur la *Figure 19*.

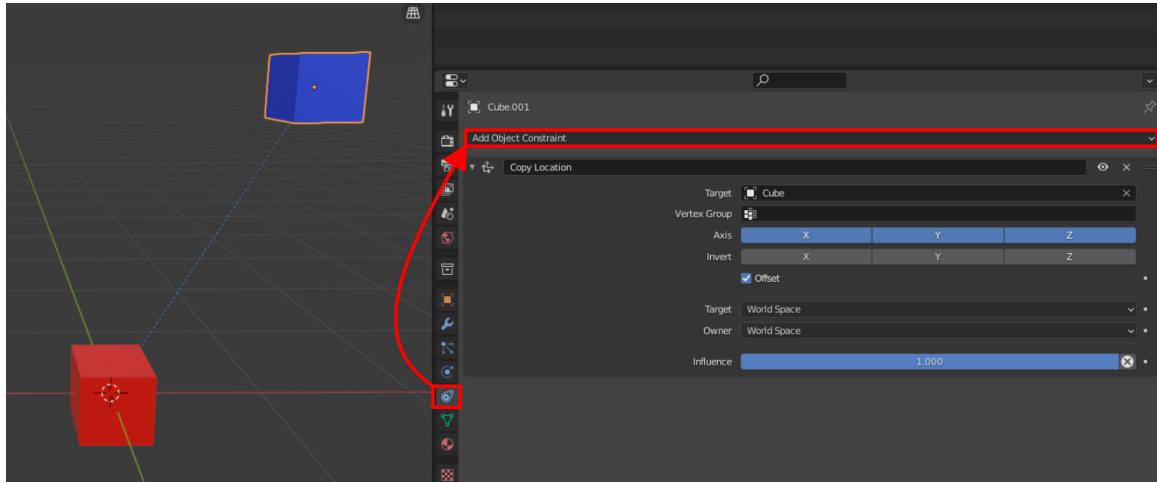


FIGURE 19 – Copie de la position (avec décalage) du cube rouge par le cube bleu via une contrainte d'objet sur ce dernier

1.5.2 Contraintes d'os

Les contraintes d'os sont disponibles (après sélection d'une armature) en mode *Pose* dans le menu indiqué sur la *Figure 20*.

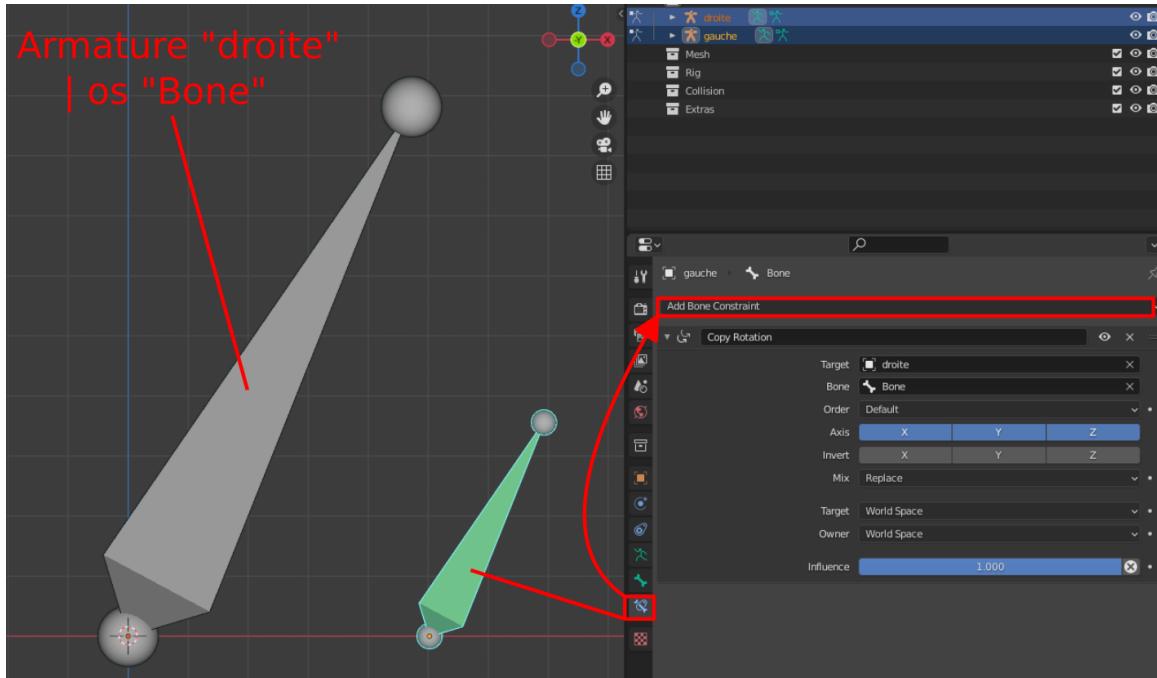


FIGURE 20 – Copie de la rotation de l'os de l'armature de droite sur l'os de l'armature de gauche via une contrainte d'os sur ce dernier



Remarque : un os affecté par une ou plusieurs contraintes est affiché en vert quand son armature est en mode *Pose*.

2 Transformer les données de capture en une armature

Ce workflow s'applique aux données de capture de mouvements fournies par le laboratoire IRISSE de l'île de La Réunion. Ces données représentent le mouvement des capteurs positionnés sur la combinaison de l'acteur réel et sont exportées en format C3D. *Blender* ne gérant pas nativement ce format, il faudra installer et activer l'add-on *io_anim_c3d* comme expliqué dans la section [1.3 Installation des add-ons](#). Le fichier **c3d to rig.blend** est un exemple de résultat possible en appliquant la méthode décrite dans cette section. La méthode présentée ici sera illustrée grâce au jeu de données **Ess.c3d**.

2.1 Importation des données

Une fois l'add-on *io_anim_c3d* activé, il sera possible d'importer les données de capture de mouvements au format C3D (*Figure 21*).

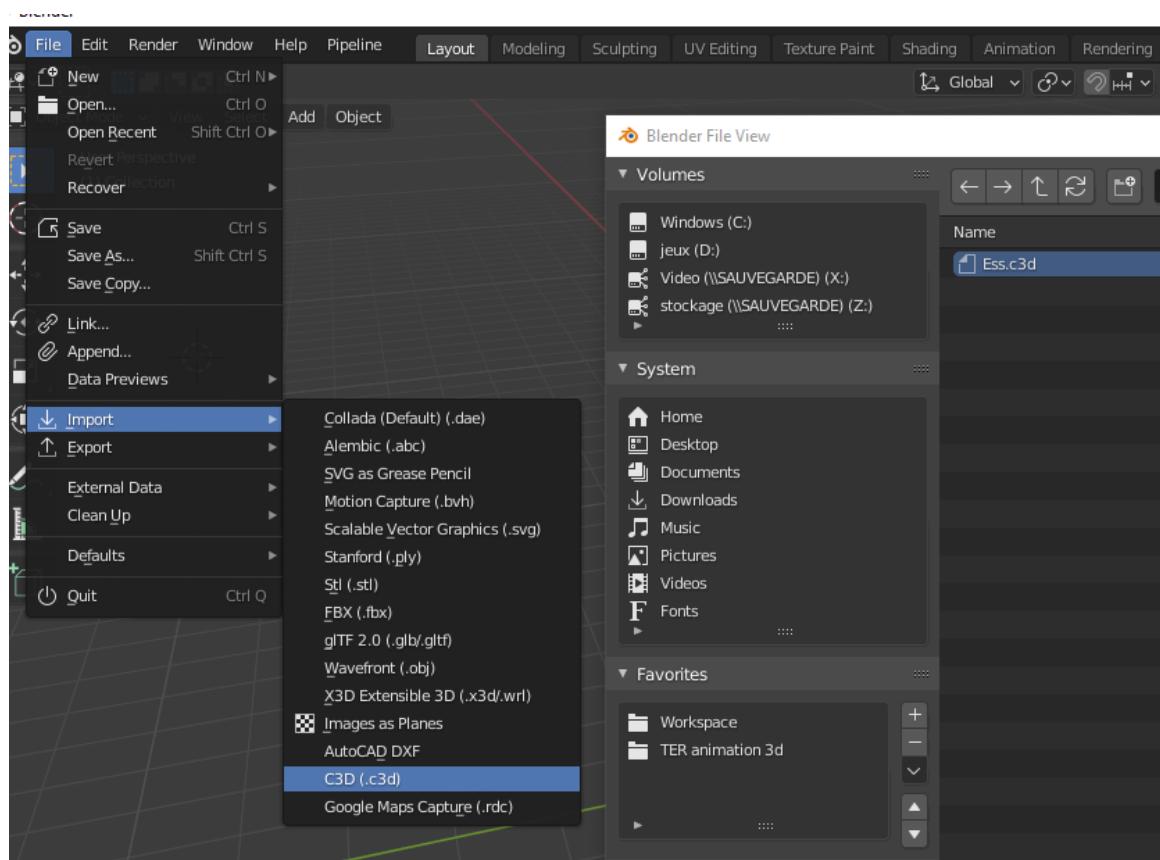


FIGURE 21 – Importation d'un fichier C3D



Comme visible sur la *Figure 22*, après importation, on obtient un nuage de points animé représenté par une armature (sans hiérarchie) portant le nom du fichier et dont ses os représentent la position des différents capteurs.

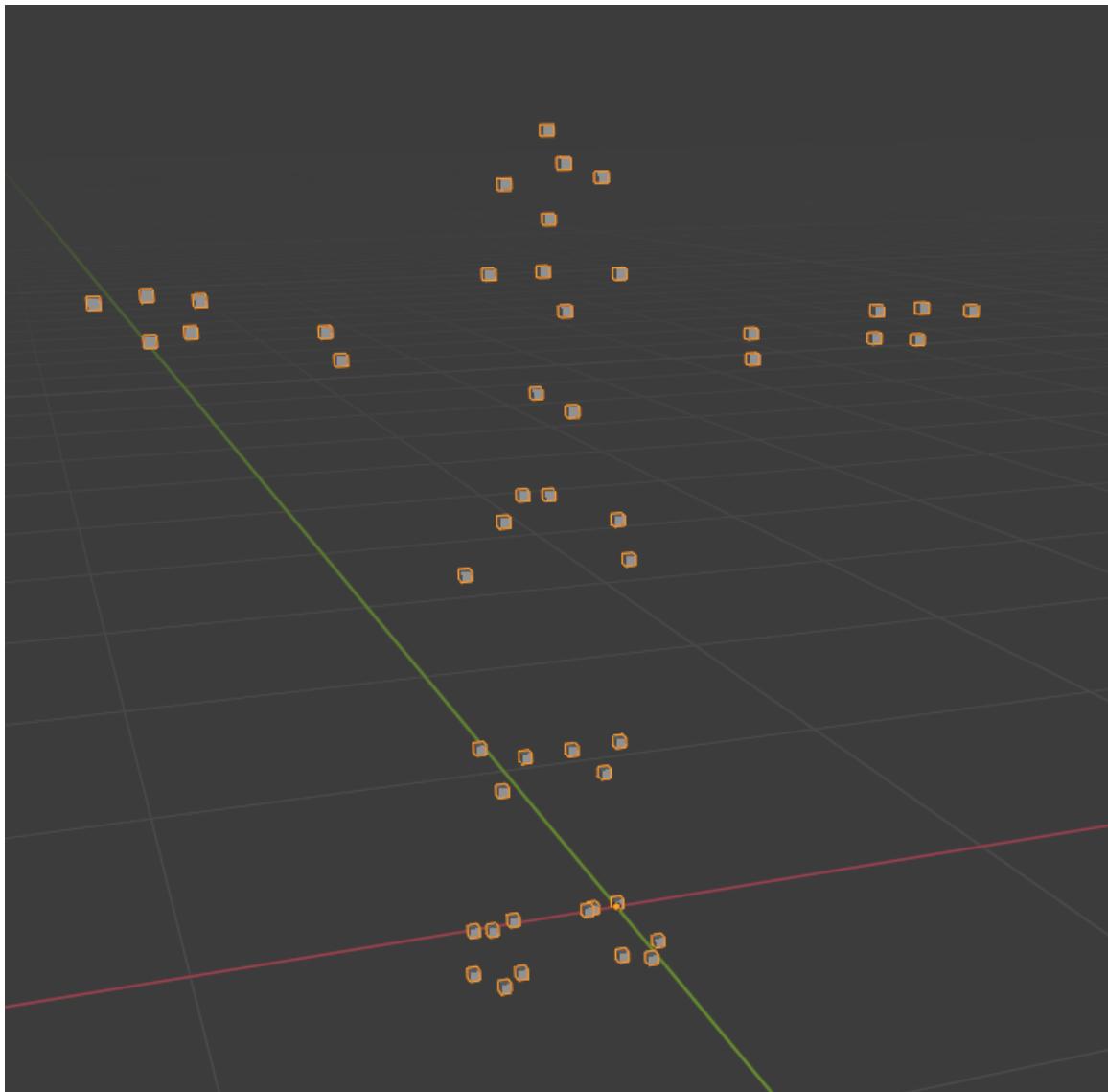


FIGURE 22 – Armature représentant les capteurs

Cependant, l'armature actuelle ne peut être utilisée telle quelle pour animer un personnage, le but de ce document et donc de présenter une méthode pour créer une armature avec une hiérarchie de bipède, qui sera animé selon les rotations des articulations induites par le nuage de points des capteurs.



2.2 Positionnement des joints

Les articulations sont placées dans un voisinage d'une partie du nuage de points représentant le membre souhaité, par exemple, l'os représentant l'avant-bras possède sa base positionnée au centre des 2 capteurs au niveau du coude et sa tête entre les 2 capteurs de la main. Le lecteur est invité à regarder le fichier [c3d to rig.blend](#). Pour ce faire, des objets intermédiaires qui représenteront les centres des articulations seront utilisés : des *Empty*, disponibles dans le menu d'ajout d'objets. Ces objets sont contraints par des constantes de positions visibles en [Figure 23](#), pour qu'ils puissent se déplacer avec le mouvement du nuage de points des capteurs (Se référer à la section [1.5.1 Contraintes d'objets](#)).

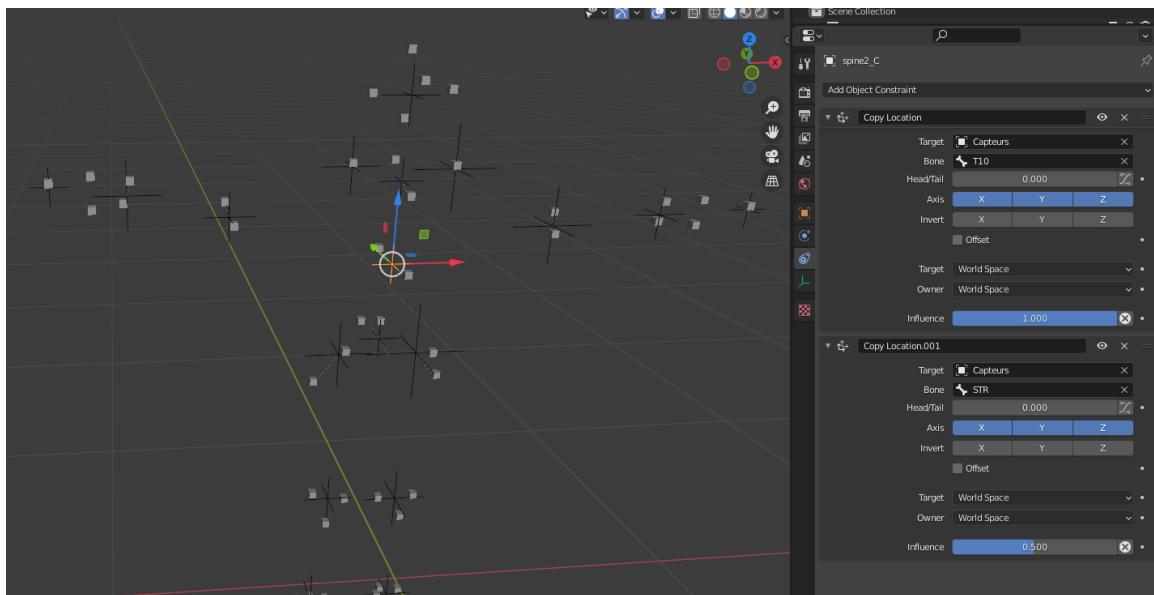


FIGURE 23 – Représentation des centres des articulations

2.3 Crédit de l'armature

Il reste encore à créer l'armature avec les conventions de nommage précisées sur la [Figure 24](#). Pour plus d'informations sur comment créer une armature, le lecteur est invité à lire la sous-section [1.4.1 Crédit de l'armature](#) de la section [1.4 Quelques connaissances sur les armatures](#).

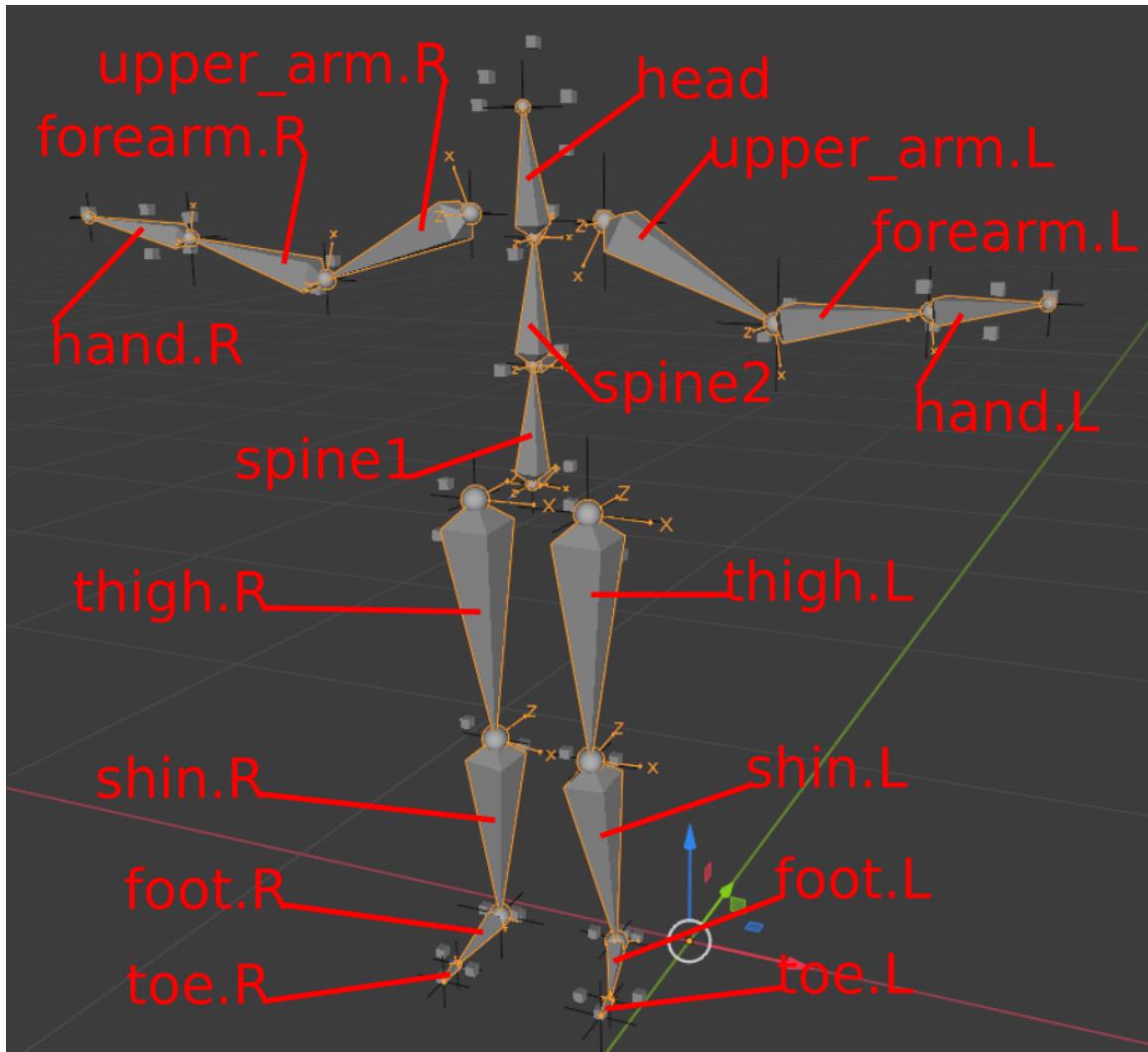


FIGURE 24 – Forme et conventions de nommage de l'armature

2.3.1 Convention pour l'orientation des os

Comme indiqué dans la sous-section [1.4.1 Création de l'armature](#) de la section [1.4 Quelques connaissances sur les armatures](#), chaque os possède un repère local dont l'axe *Y* est son axe de rotation principal. Pour éviter des problèmes de rotations lors du retargeting, il est recommandé de suivre la même convention d'orientation des os que celle de l'add-on *Rigify* (section [1.3 Installation des add-ons](#)). Vous pouvez ajouter un bipède de *Rigify* comme indiqué sur la [Figure 25](#) et observer ses repères ou directement les regarder sur l'exemple [c3d to rig.blend](#).

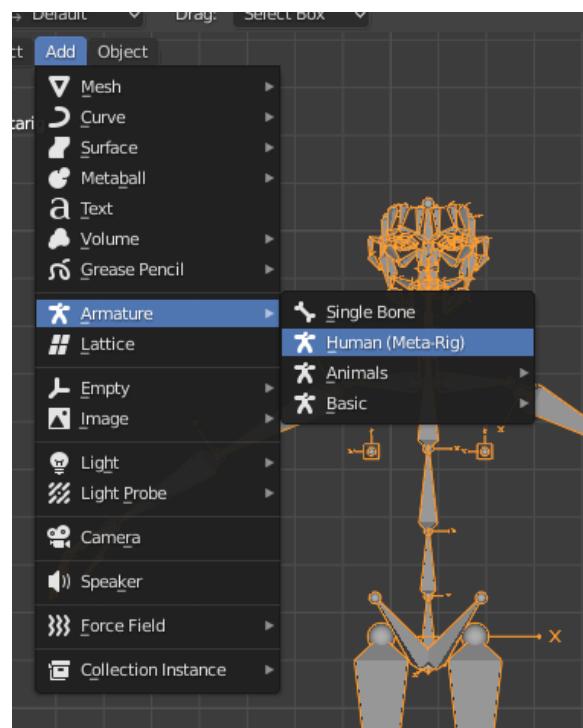


FIGURE 25 – Ajout d'une armature de *Rigify*



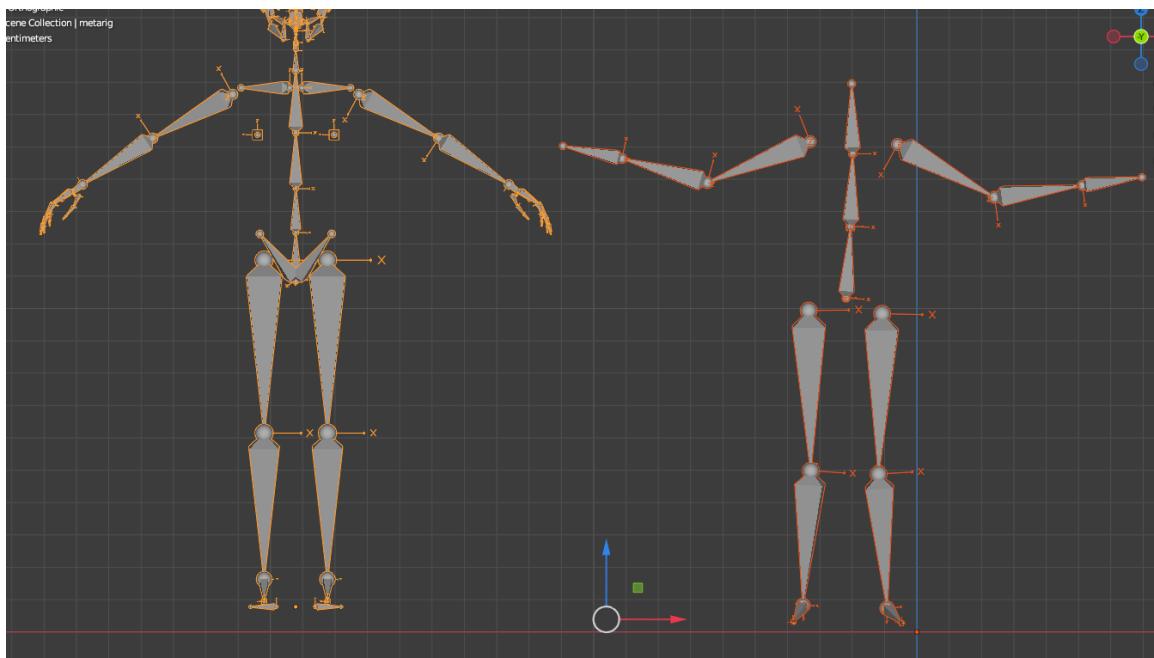


FIGURE 26 – Orientation des os : Rigify à gauche, l'armature précédemment créée à droite

2.4 Contraindre l'armature au mouvement

Pour animer l'armature précédemment créée, il faut contraindre ses os aux centres des articulations.

2.4.1 Contraintes principales des os

Un os doit copier la position du centre de son articulation et il doit s'orienter vers le centre de la prochaine articulation. Les contraintes utilisées sont indiquées sur la *Figure 27*. Ce processus doit être répété pour tous les os de l'armature. Voir la section *1.5.2 Contraintes d'os* pour plus d'informations sur les contraintes d'os.



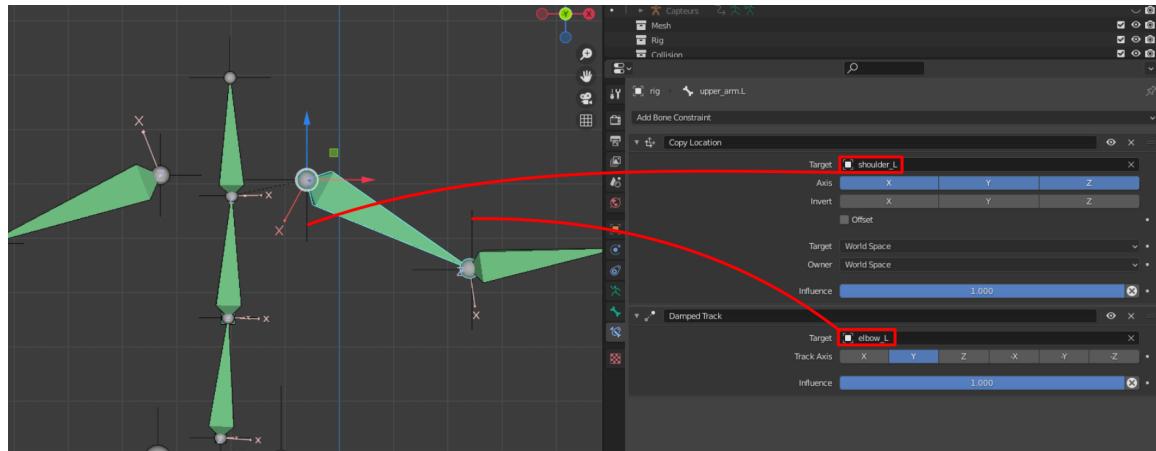


FIGURE 27 – Les contraintes à appliquer pour chaque os

2.4.2 Correction des rotations

Ces deux contraintes ne permettent pas d'en déduire la rotation induite par les capteurs d'un membre selon son axe local Y , en effet, le centre de l'articulation est invariant par rotation en sa position. Prenons l'exemple de la colonne vertébrale (Figure 28), une rotation du buste ne change pas la position de la prochaine vertèbre, la contrainte de suivi n'a donc aucun ajustement à réaliser. Pour palier à ce problème, il faut introduire des os intermédiaires qui eux vont suivre les capteurs dorsaux, ce qui induira la rotation sur l'axe de la colonne. Il faudra finalement copier la rotation de l'axe Z de cet os intermédiaire sur l'os représentant la vertèbre, comme indiqué sur la Figure 29.

Remarque : l'os intermédiaire doit être aligné correctement, c'est-à-dire avoir son repère local Z vers le haut.

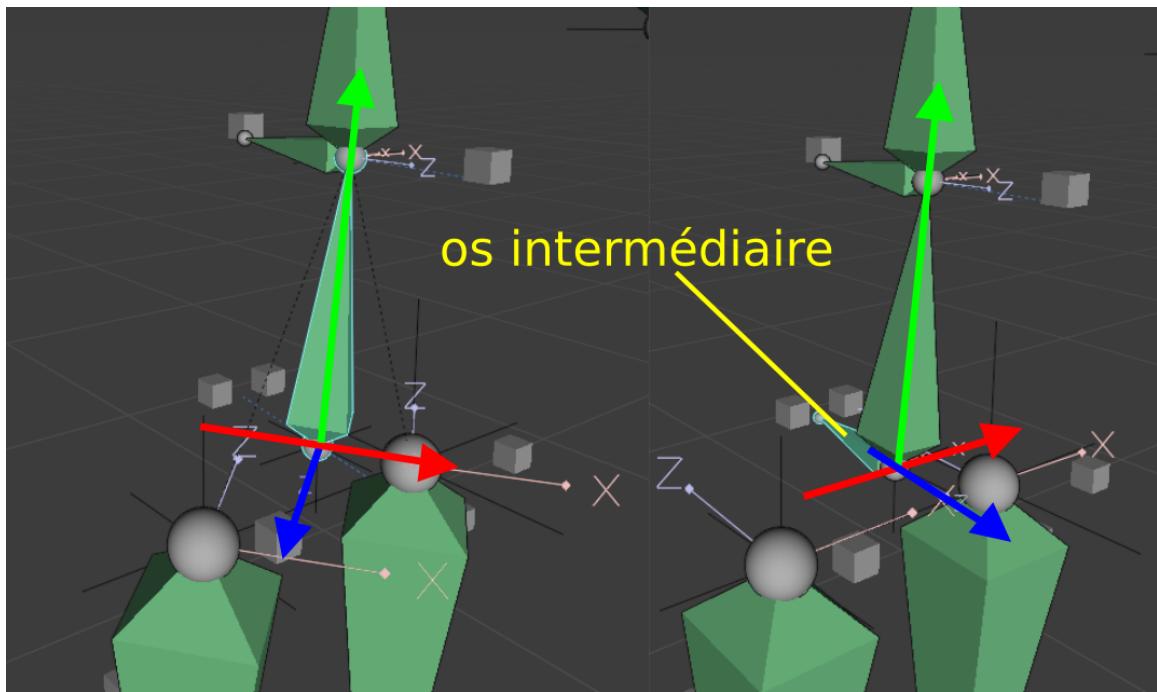


FIGURE 28 – Orientation incorrecte de la première vertèbre. Correction de l'orientation grâce à un os intermédiaire à droite.

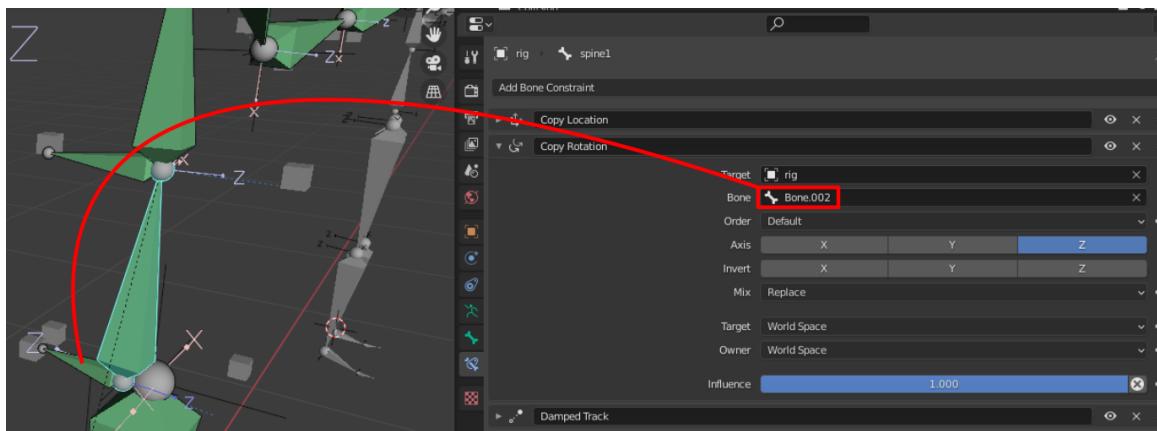


FIGURE 29 – Contrainte de rotation

Après toutes ces étapes, l'armature devrait être correctement animée.

2.5 Exporter le résultat

Il est préférable de ne pas utiliser directement cette armature, en effet, celle-ci est animée grâce aux différentes contraintes appliquées aux os et aux *Empty*. Il est préférable d'utiliser une armature



dont les os sont animés à base de clés, une clé étant un enregistrement à un instant précis d'un ensemble de propriétés (rotation, position, etc.). Cette section explique donc comment exporter l'armature avec son animation au format BVH pour être utilisée lors du retargeting.

2.5.1 Pré-calculer l'animation

Avant de procéder, il est recommandé de dupliquer l'armature actuelle en tant que sauvegarde, en effet, le pré-calculation effacera les contraintes ajoutées précédemment. En mode *Object*, avec l'armature de sélectionnée, il suffit de suivre comme indiqué sur la [Figure 30](#). Un panneau de configuration s'ouvrira alors, il est important d'avoir les mêmes valeurs d'options que celles encadrées sur la [Figure 31](#), laisser les autres options à leur valeur par défaut.



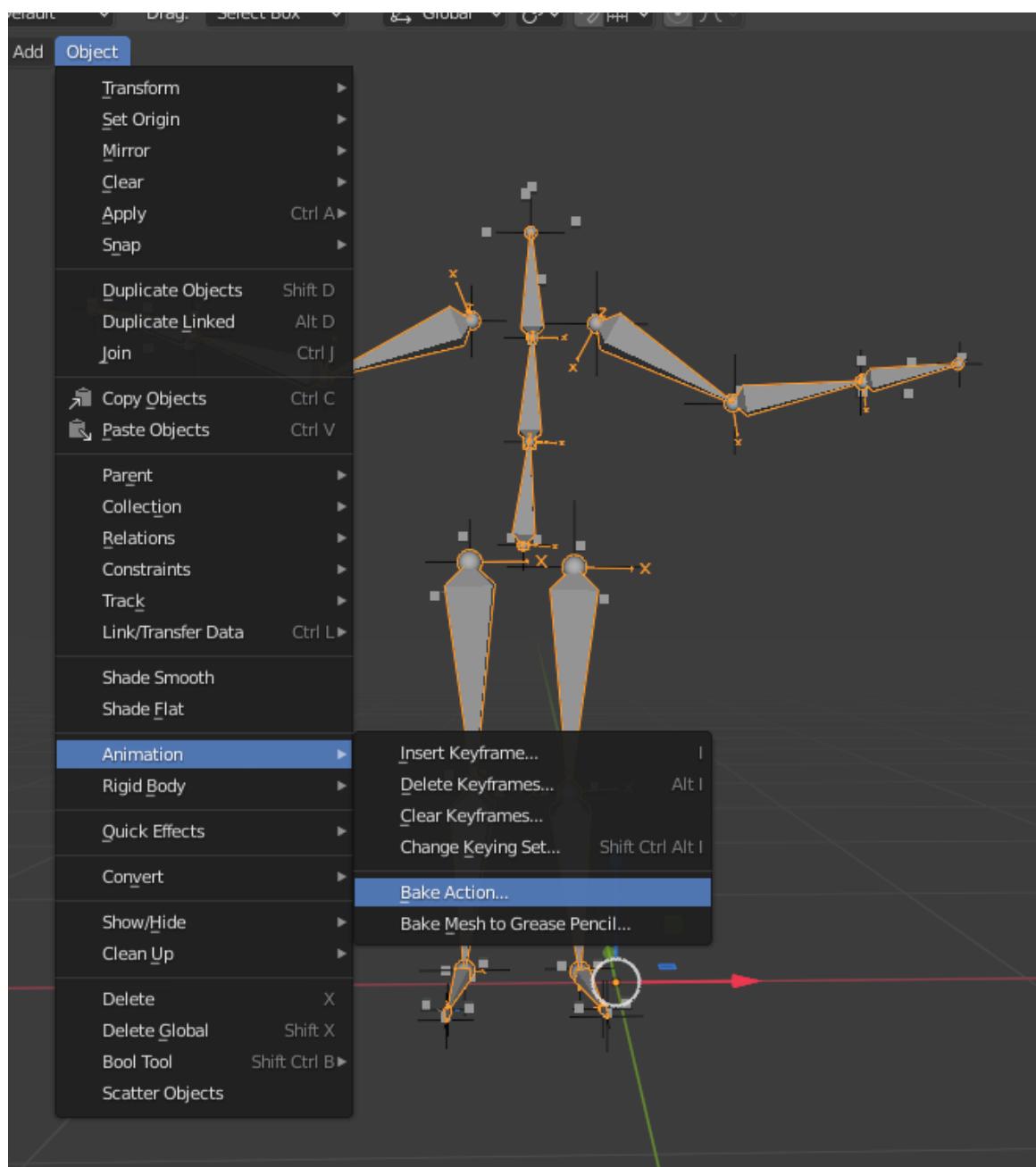


FIGURE 30 – Pré-calculer l'animation



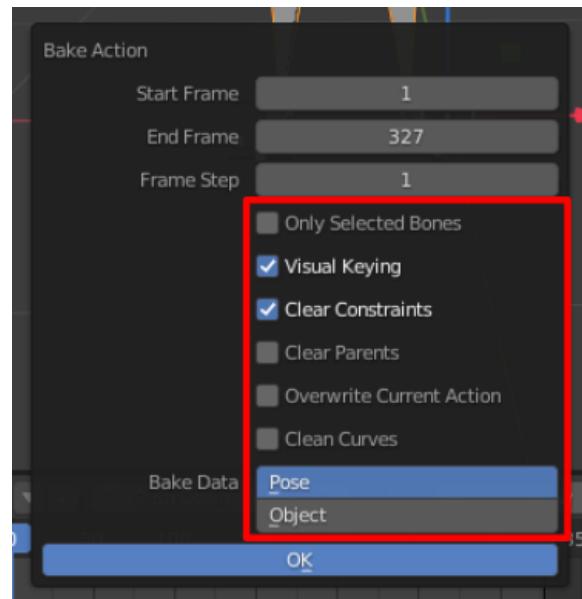


FIGURE 31 – Options du pré-calculation

2.5.2 Nettoyer l'armature

Maintenant l'animation pré-calculée, il est possible de déplacer librement l'armature dans la scène. Les os intermédiaires pour corriger la rotation des vertèbres ne sont plus nécessaires et peuvent être retirés en mode *Edit*. La *Figure 32* montre l'armature nettoyée et animée par clés.



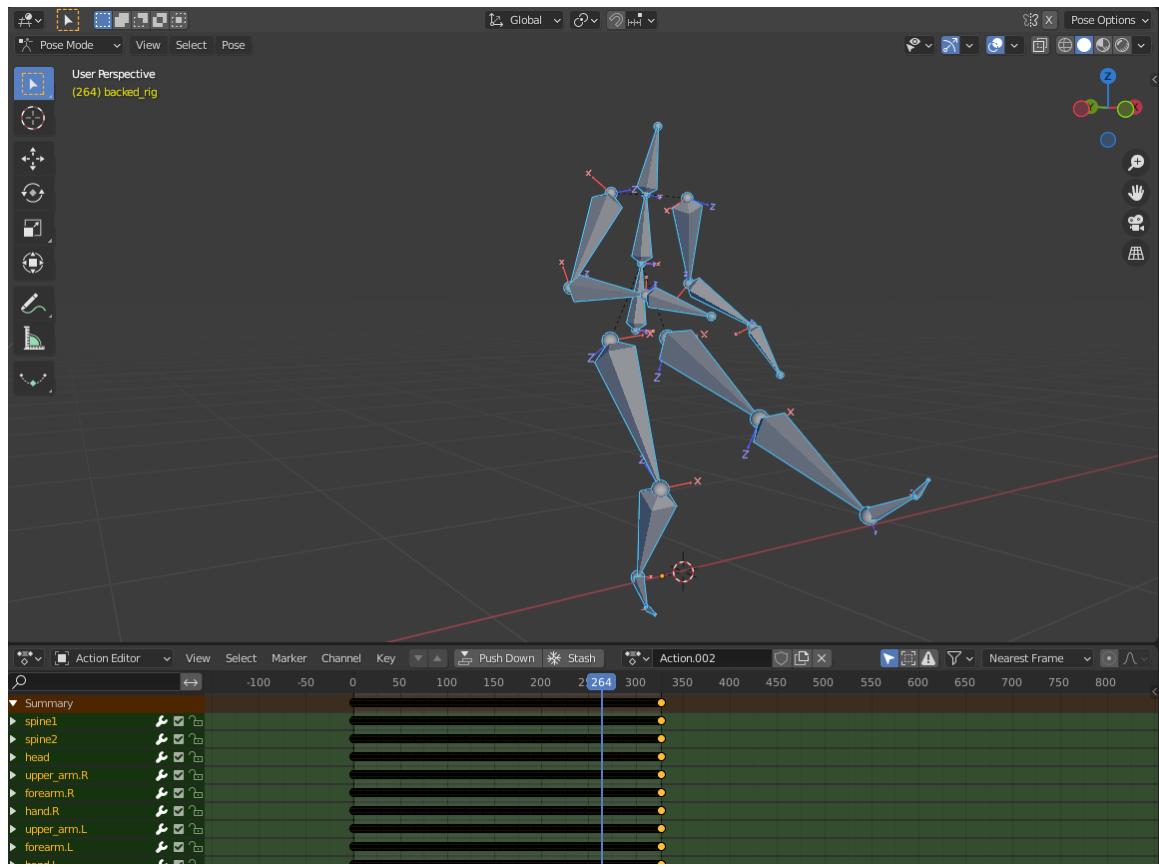


FIGURE 32 – Armature finale avec clés d'animation (panneau du bas)

2.5.3 Export en BVH

L'armature peut enfin être exportée au format BVH comme indiqué sur la *Figure 33* en ayant l'armature finale sélectionnée en mode *Object*.

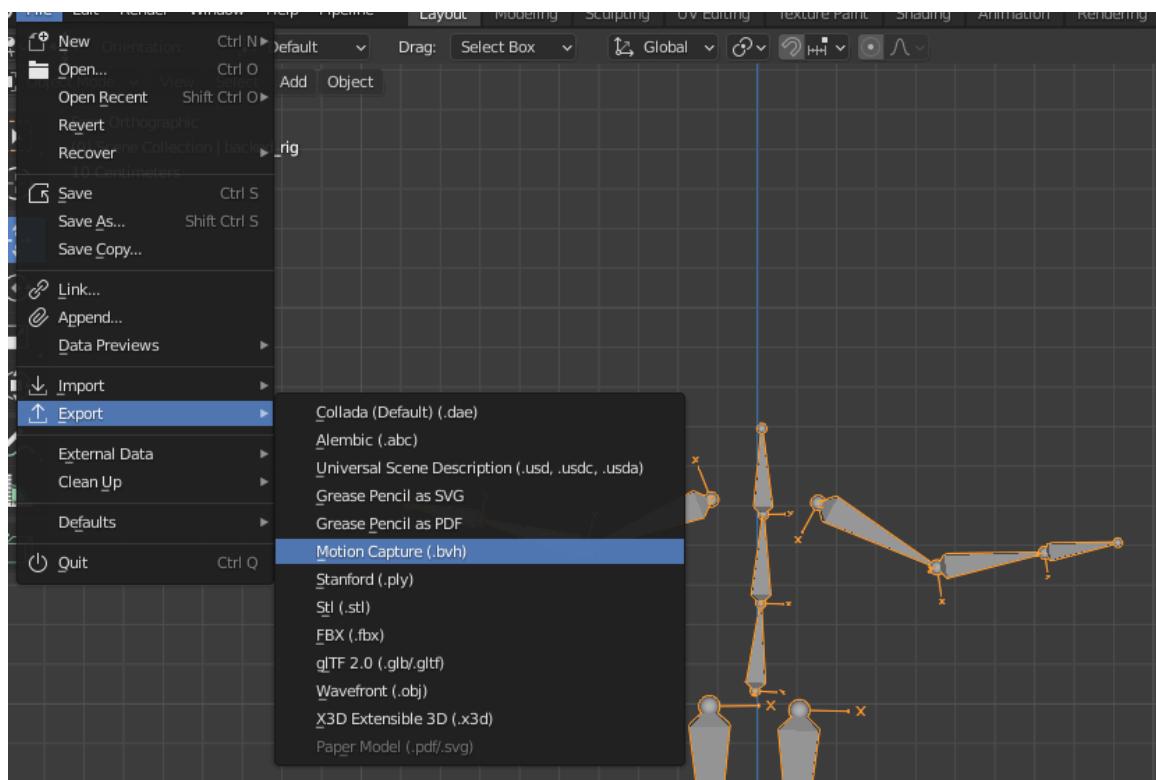


FIGURE 33 – Exportation en BVH

3 Retargeting manuel

Le retargeting consiste à transférer l'animation d'une armature sur une autre (qui est en règle général similaire, mais pas forcément identique). La solution de retargeting présentée dans ce document consiste à ajouter le delta de rotation (la rotation effectuée par un os entre deux images) d'un os source sur la rotation initiale d'un os cible. Ce qui permet de modifier la pose initiale de l'armature cible, en effet, lorsque celle-ci a une morphologie différente, cela permet de conserver son volume lors du transfert de l'animation. Un exemple de ce phénomène est visible en *Figure 34*.



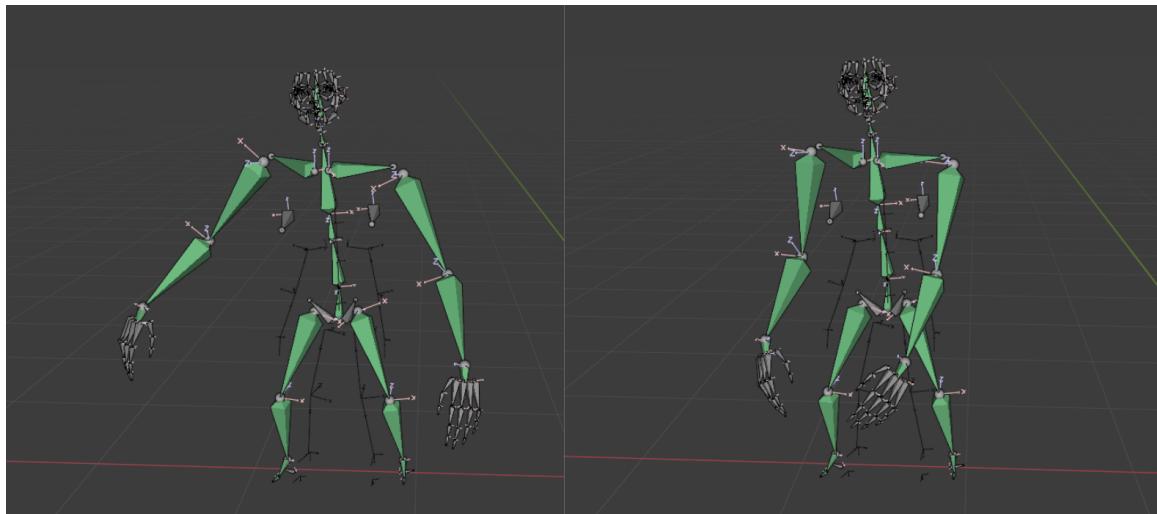


FIGURE 34 – Ajout du delta de rotation à la pose initiale à gauche (correct). Copie directe de la rotation image par image à droite (incorrect).

Le fichier [Mixamo_manual_retargeting_test_2.blend](#) contient un exemple de retargeting effectué avec cette méthode manuelle.

3.1 Contraintes à appliquer

Sur la Figure [Figure 35](#), l'os sélectionné à droite ajoute à sa rotation actuelle le delta de rotation de l'os similaire sélectionné à gauche. La contrainte à appliquer sur l'os sélectionné à droite est présente sur la [Figure 35](#).

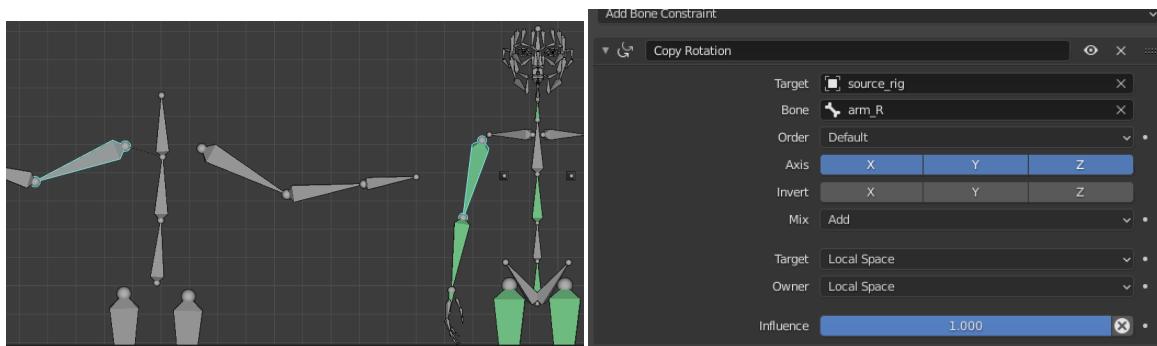


FIGURE 35 – Paramètres de la contrainte

On peut remarquer l'utilisation de l'option *Mix* en mode *Add* et l'utilisation des repères locaux aux 2 os, c'est ce qui permet d'ajouter le delta de rotation au lieu de copier la rotation à un instant précis. Cette contrainte est à appliquer sur tous les os de l'armature cible.



3.1.1 Spécificité de l'os racine

L'os racine (ou *root* en anglais) est l'os représentant la racine de la hiérarchie des os, dans l'exemple du fichier *Mixamo_manual_retargeting_test_2.blend*, l'os racine est la première vertèbre, c'est-à-dire l'os nommé *spine*. C'est le seul os à pouvoir se déplacer en plus d'être orienté. Ainsi, il faut copier les déplacements de l'os racine de l'armature source sur l'os racine de l'armature cible comme indiqué sur la *Figure 36*.

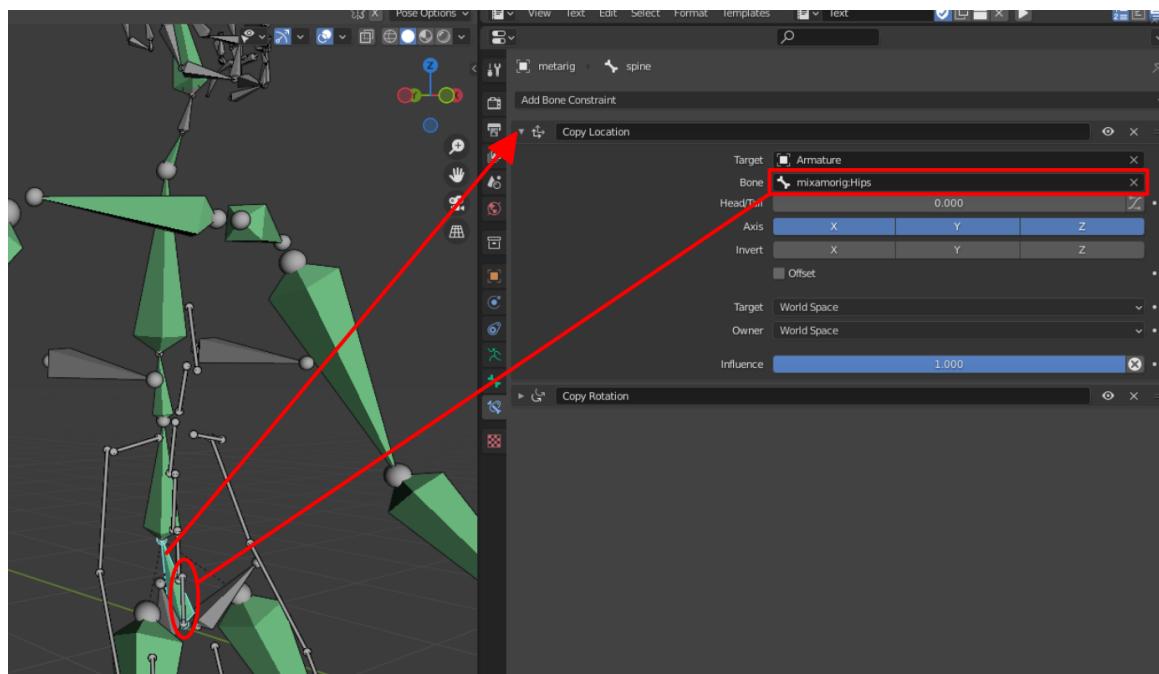


FIGURE 36 – Contrainte de position

4 Retargeting assisté

Le lecteur aura remarqué, après lecture de la section précédente, du temps nécessaire pour appliquer ces contraintes de rotations sur tous les os de l'armature cible. Cette tache est fort heureusement automatisable grâce un add-on développé pour ce workflow : *SimpleRetargeting* (voir la section [1.3 Installation des add-ons](#)).

4.1 Présentation de l'add-on

L'add-on se trouve dans l'onglet *SimpleRetargeting*, initialement seul le panneau *Main settings* possède des éléments. Comme indiqué sur la *Figure 37*, l'add-on attend une armature source (celle contenant l'animation à transférer) et une armature cible.



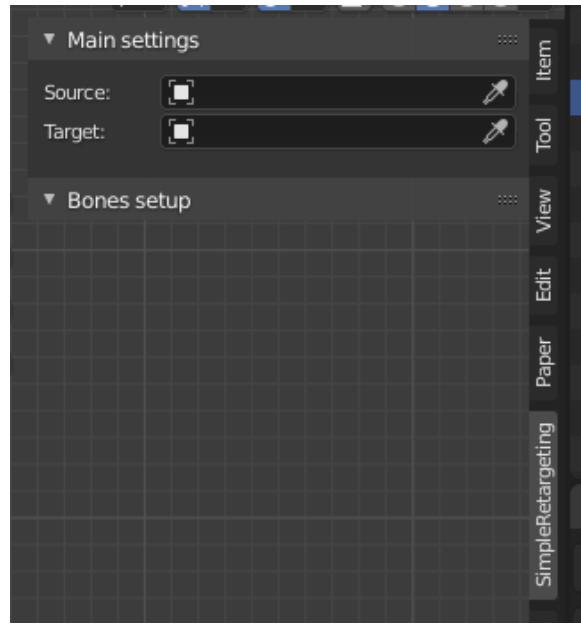


FIGURE 37 – Les deux champs requis par l'add-on

Après avoir renseigné ces 2 champs, plusieurs éléments apparaissent pour configurer le retargeting comme visible sur la *Figure 39*, l'option *Target root* sert à renseigner l'os racine de la cible, c'est celui-ci qui aura la contrainte de position. Les boutons *Import preset* et *Export preset* permettent de créer un système de préréglage en sauvegardant la configuration de ces nouveaux éléments.

Le panneau *Bones setup* permet de lier les os de la cible et de la source entre eux. Ce panneau se découpe en 3 parties :

1. des contrôles sous la forme de boutons pour assister l'utilisateur de façon graphique
2. une barre de recherche avec un filtre (os renseignés, non renseignés ou tout)
3. la liste des os de la cible avec un champ pour l'os de la source qui lui est associé et des boutons pour pouvoir inverser les rotations copiées

Voir la *Figure 38* pour plus d'informations.

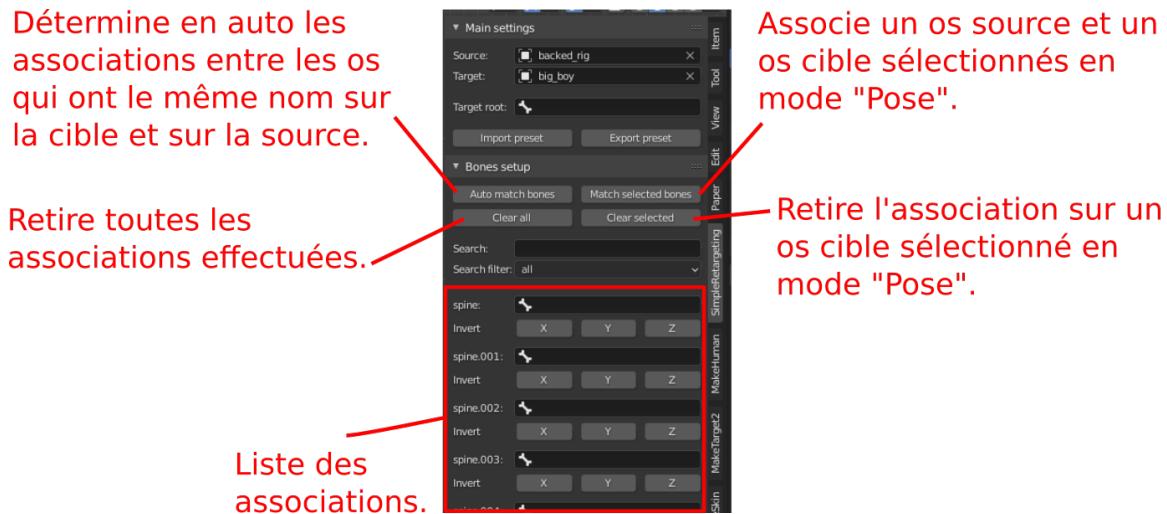


FIGURE 38 – Informations additionnelles sur l'add-on

4.2 Utilisation

L'utilisation de l'add-on sera démontré par l'exemple, dans celui-ci, il faudra transférer l'animation d'une armature BVH créée avec la méthode de la section *2 Transformer les données de capture en une armature* sur une armature cible différente. Le lecteur pourra suivre cette section en appliquant les explications fournies sur le fichier de test : `addonTest.blend`.

Commencez par renseigner l'armature source et l'armature cible comme indiqué sur la *Figure 39*.

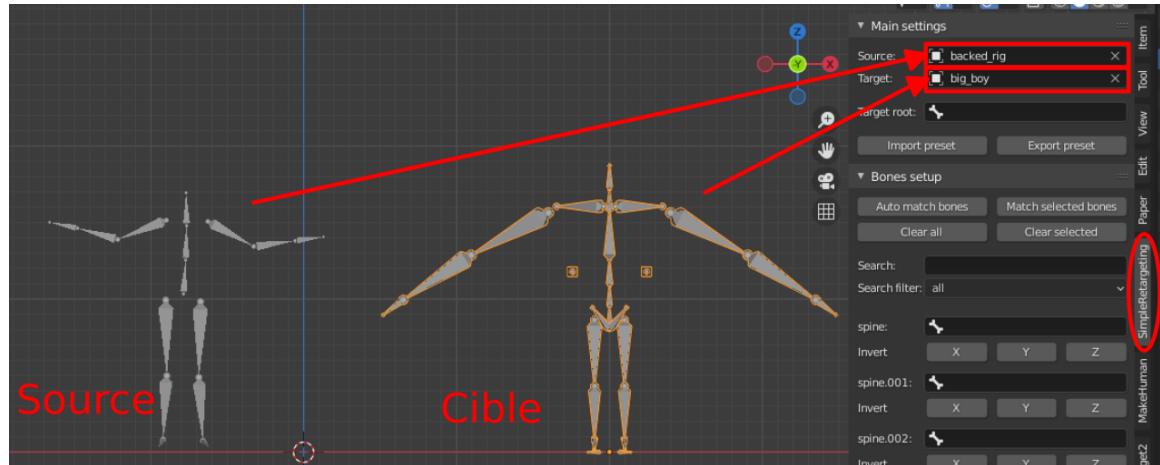


FIGURE 39 – Sélection de la source et de la cible

4.2.1 Configurer l'association des os

Nous allons maintenant procéder à l'association des os. Passez les 2 armatures en mode *Pose* pour plus de lisibilité et essayez de faire correspondre les os en automatique grâce au bouton *Auto match bones*. Le résultat de cette opération est visible sur la [Figure 40](#).

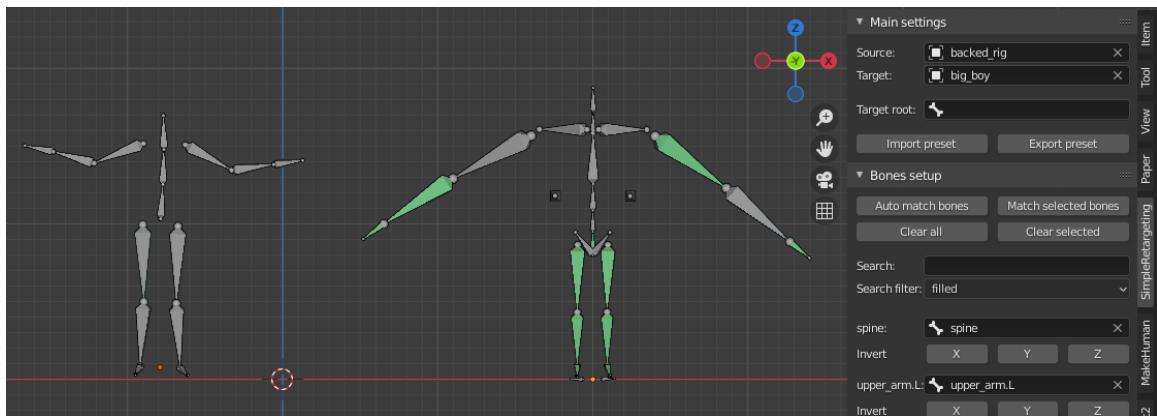


FIGURE 40 – Résultat de la correspondance en automatique

Comme vous pouvez le voir, certains os pouvant être associés n'ont pas été pris en compte, les associations manquantes sont renseignées sur la [Table 1](#) et sont visibles en [Figure 41](#).

Os de la cible	Os de la source
<i>upperarm.R</i>	<i>upper_arm.R</i>
<i>forearmv.L</i>	<i>forearm.L</i>
<i>spine.003</i>	<i>spine.1</i>
<i>spine.006</i>	<i>head</i>

TABLE 1 – Associations manquantes

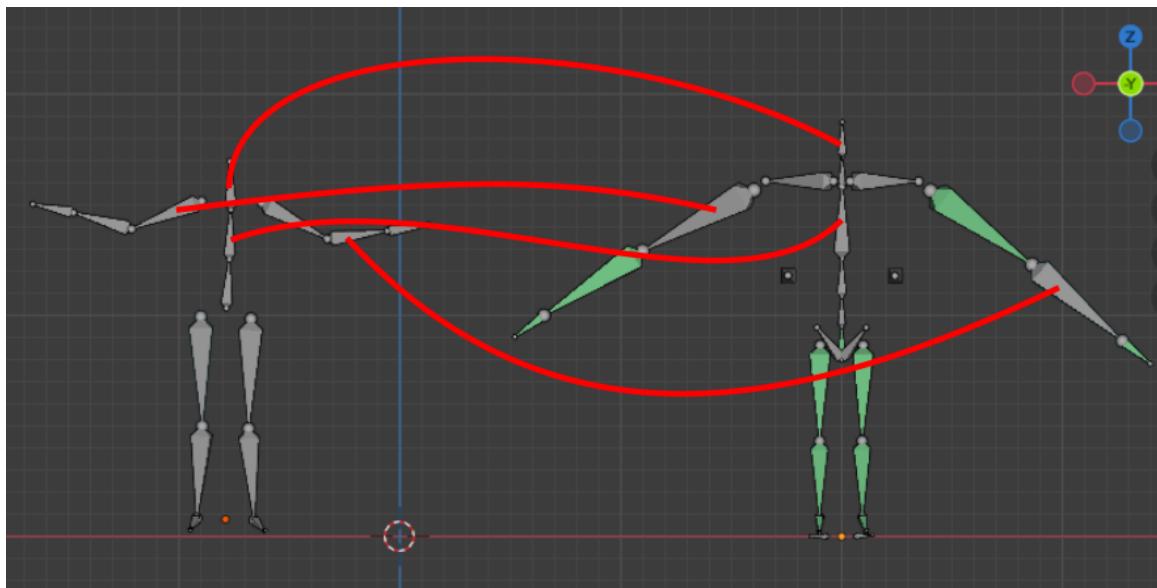
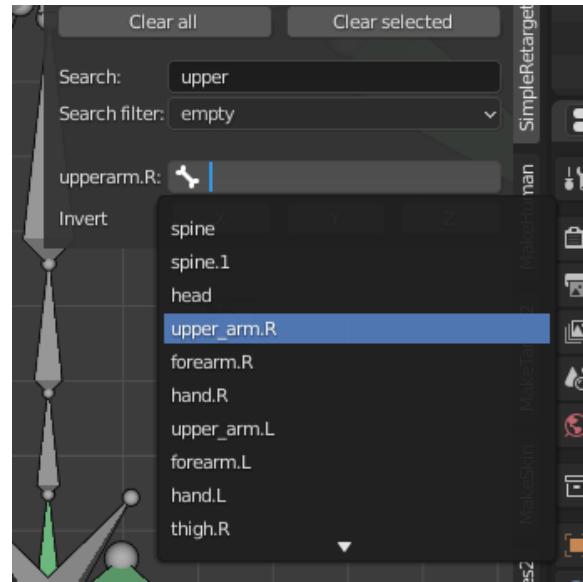
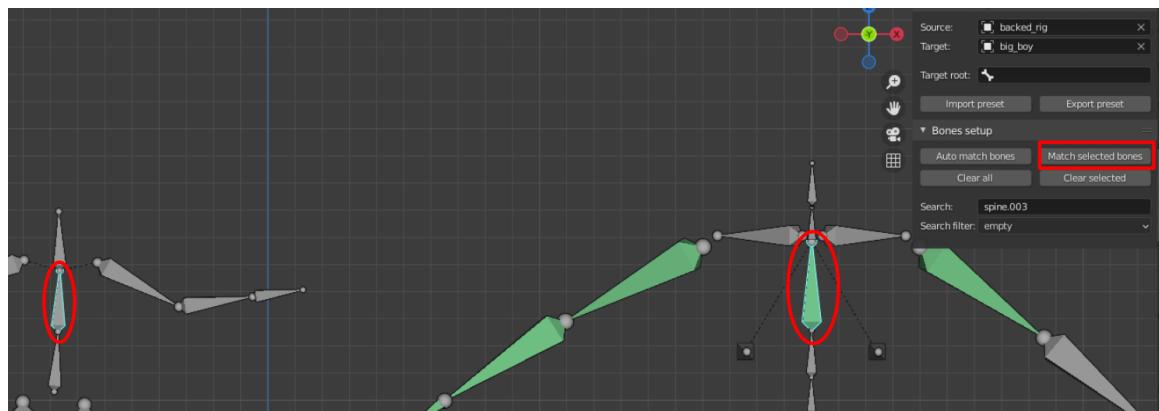


FIGURE 41 – Associations manquantes

Les autres os n'ont pas d'os analogue sur l'armature cible, on peut donc les ignorer.

Pour corriger cela, il faut soit renseigner pour chacun des os concernés de la cible son os analogue sur la source via les champs de la liste des os du panneau *Bones setup* (*Figure 42*), ou en sélectionnant l'os source et l'os cible concernés puis en utilisant l'opération *Match selected bones* (*Figure 43*). Effectuez maintenant les assignations manquantes.

FIGURE 42 – Association *upperarm.R* \Leftrightarrow *upper_arm.R*FIGURE 43 – Association *spine.003* \Leftrightarrow *spine.1*

Pour l'os racine, renseigner dans le champ *Target root* l'os *spine*, la [Figure 44](#) montre le résultat final de la configuration.



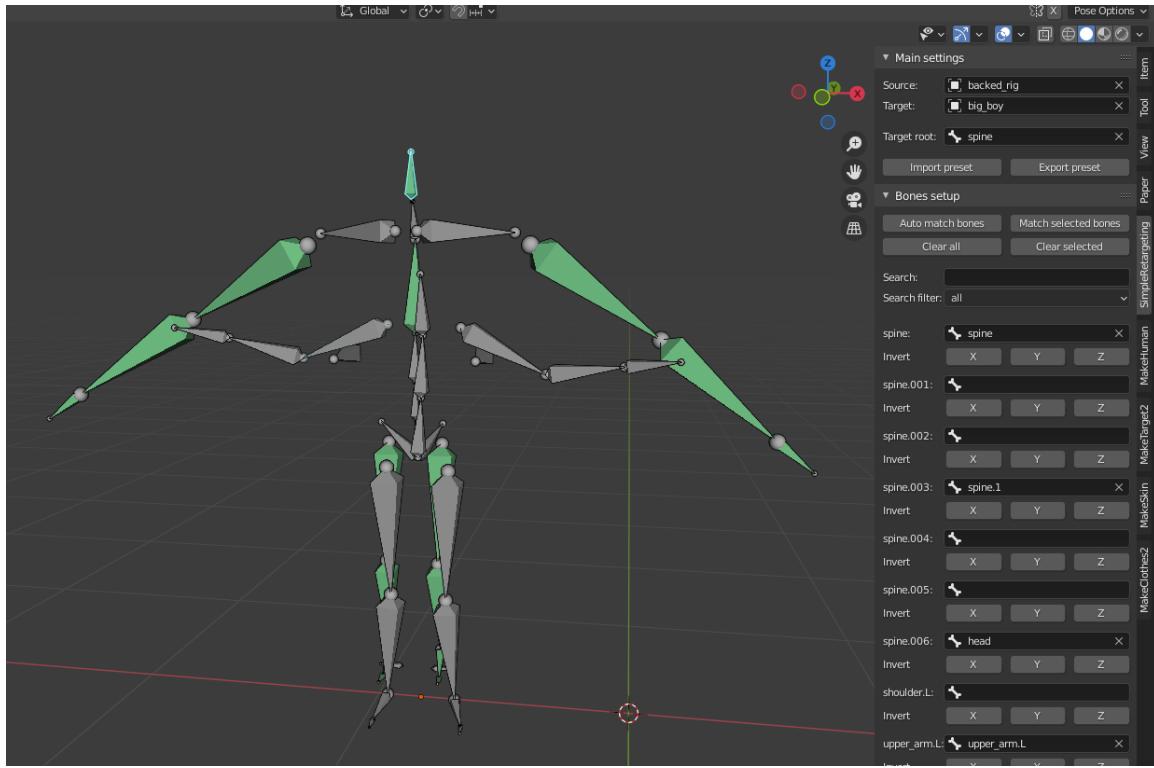


FIGURE 44 – Fin de la configuration

4.2.2 Sauvegarder la configuration

Avant d'ajuster le retargeting (corriger le mouvement des bras et la position des pieds par exemple), il est conseillé de sauvegarder la configuration actuelle grâce à l'opération *Export preset*. La configuration est exportée en format CSV textuel (donc humainement lisible). Par la suite, s'il est souhaité de transférer une animation stockée dans une autre armature, mais que celle-ci possède les mêmes noms d'os que précédemment, il suffira d'importer le fichier CSV via *Import preset* et les associations seront chargées automatiquement.

4.2.3 Ajustements du retargeting

Pour le moment, l'animation transférée n'est pas correcte, par exemple, les bras ne décrivent pas les mêmes mouvements et les pieds sont au-dessus du sol. Pour corriger la position des pieds, il suffit de rétrécir l'armature source en mode *Object* jusqu'à ce que l'armature cible ait les pieds au niveau du sol (*Figure 45*).

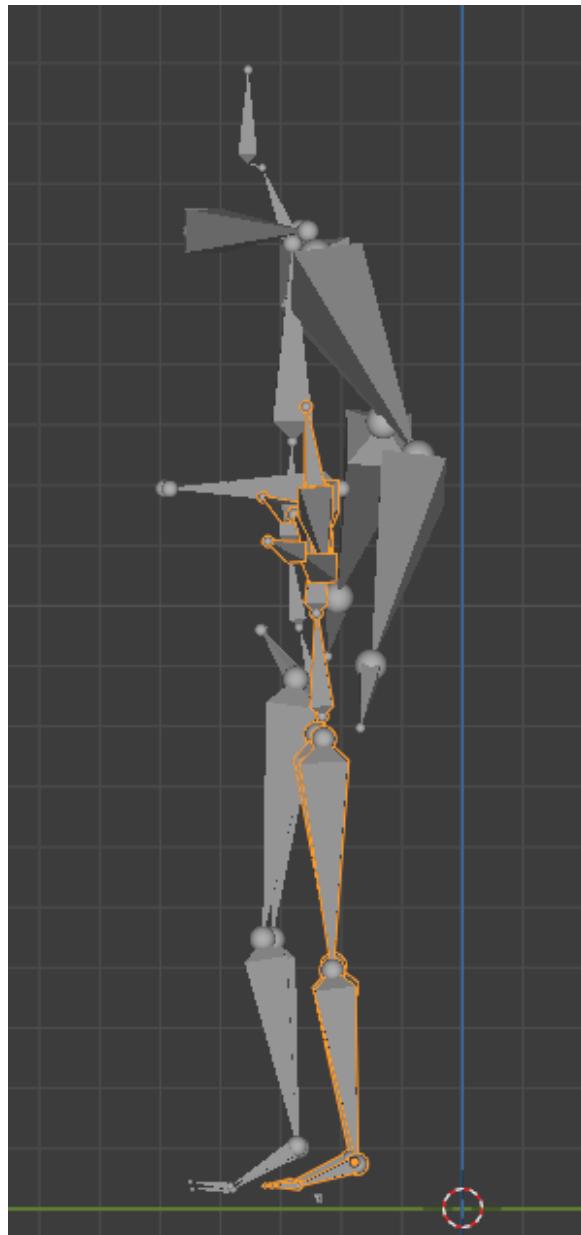


FIGURE 45 – Redimensionnement de l'armature source pour aligner les pieds

Concernant les bras, ceci est dû à la méthode de retargeting de ce workflow qui se base sur un delta de rotation, bien que cela permette de conserver le volume de l'armature cible, il faut manuellement faire correspondre au mieux la pose de celle-ci avec la pose initiale de l'armature source comme visible en *Figure 46*.



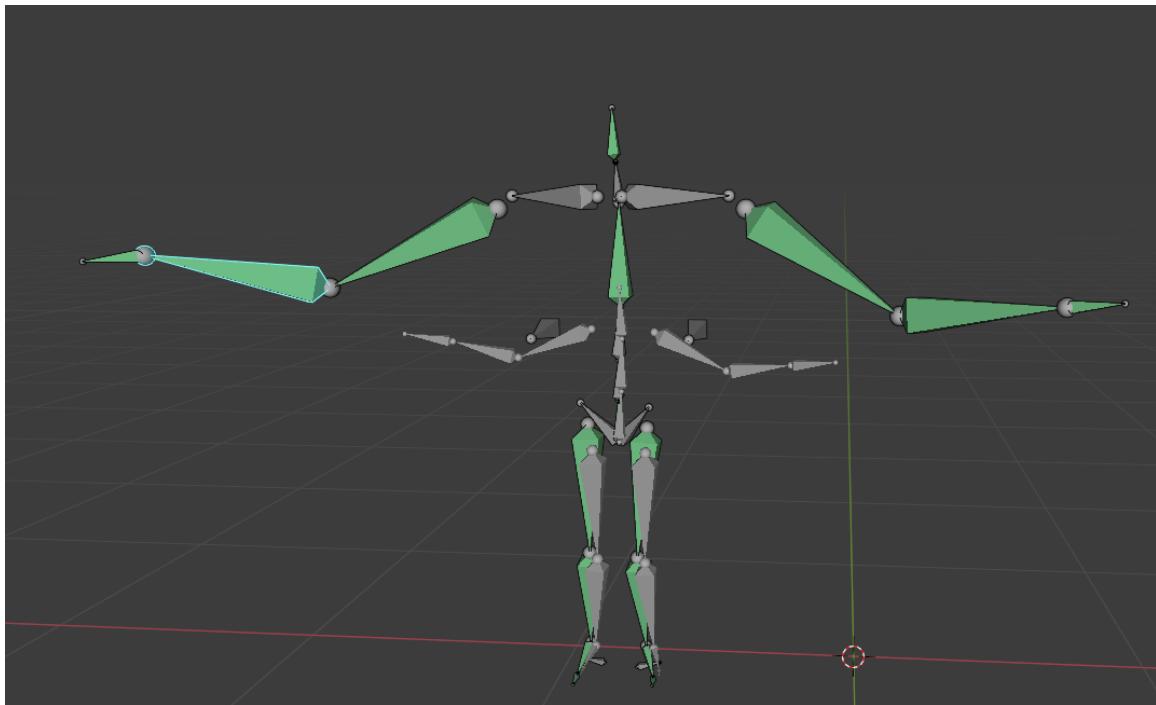


FIGURE 46 – Correspondance avec la pose initiale de la source

4.3 Pré-calculer l'animation

Il est maintenant possible de pré-calculer l'animation comme expliqué en [2.5.1 Pré-calculer l'animation](#) de la section [2 Transformer les données de capture en une armature](#). Cette fois, il n'est pas nécessaire de dupliquer l'armature, car la configuration de l'add-on a été sauvegardée, il suffira de la réimporter pour transférer une autre animation sur cette armature.

5 Démonstration du workflow

Pour démontrer le fonctionnement de ce workflow, un jeu 3D sur l'île de La Réunion a été réalisé sous [Unreal Engine 4](#) en une semaine : [GrandRaid](#). Cette démo est disponible à l'adresse suivante : <https://smallcluster.itch.io/grandraid>. Pour information, toutes les animations des personnages 3D ont été réalisées en utilisant l'add-on de ce workflow, sur des animations de la bibliothèque de capture de mouvements [Mixamo](#) (Voir un exemple en [Figure 47](#)).

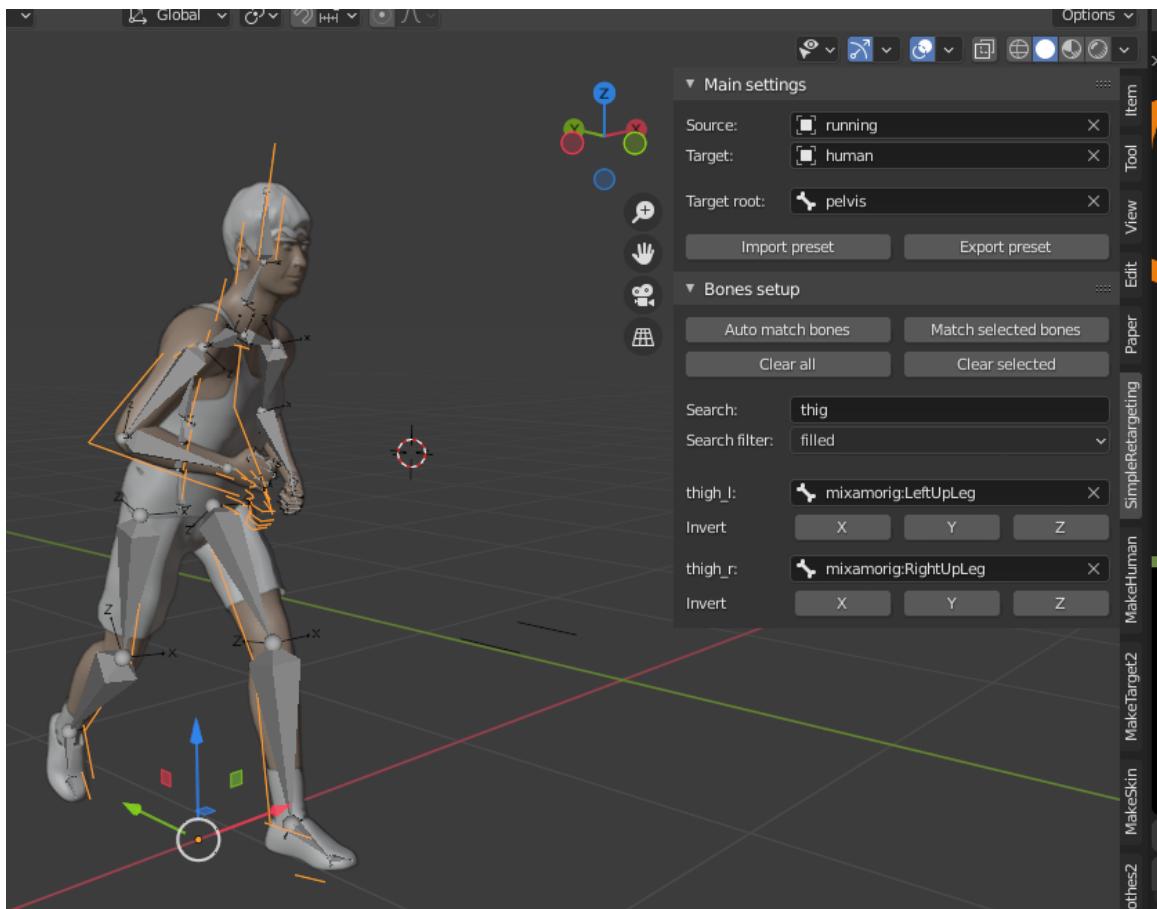


FIGURE 47 – animation de course

Voici quelques images du jeu :



