

SkateboardXXX3000

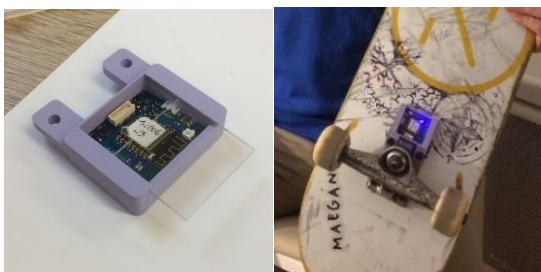
Faire de l'analyse de données de skate

Le projet

SkateboardXXX3000 est un capteur (**Movuino**) d'**accélération** et **gyroscopique** placé sous le skate permettant d'analyser les **mouvements simples**¹ qu'un skateur peut faire. J'ai démarré ce projet l'année dernière durant ma césure, et je l'ai continué en tant que projet de fin d'étude cette année avec un camarade. Nous avons orienté notre travail sur la **reconnaissance des figures**. Nous sommes très fiers des premiers résultats que nous avons obtenus, mais il reste encore beaucoup à faire et à améliorer.

Les motivations

Quand j'ai parlé de ce projet à mes encadrants de stage, Joël et Adrien, je voyais en lui l'opportunité de challenger mes capacités de futur ingénieur sur un sujet qui me passionne depuis maintenant 2 ans et demi. Très intéressé par la programmation et l'analyse de données, avec l'aide de nombreuses personnes, nous avons monté brique par brique les premières étapes du projet. Ces étapes allant de la confection d'un **boîtier** se fixant sur le skate à la **reconnaissance de figure** en passant par la **visualisation 3D** des figures.



Le movuino placé dans le boîtier et sur le skate

Les enjeux

Sportifs

Prévention de la blessure, étude de la performance, apprentissage de la discipline

Pédagogiques

Etude et compréhension de la discipline avec une approche scientifique

Nos objectifs

La reconnaissance de figure

Détection de figure, classification...

Extraction des données pertinentes pour la pratique du skate

Hauteur, évaluation de la progression...

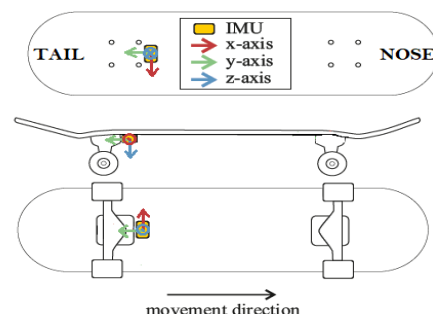


Schéma du movuino et de ses axes sur le skate

L'approche générale

Simple mais efficace

Dans l'état de l'art, nous pouvons trouver beaucoup d'articles scientifiques ayant traité de problématiques similaires. Cependant nous avons remarqué que peu font le lien entre la discipline et leurs données. Les données sont pourtant très visuelles et compréhensibles.

¹ Simplicité : Action simple voir instinctive pour la personne qui l'exécute mais en réalité très complexe

C'est pourquoi nous avons abordé une démarche simple, voire naïve afin d'exploiter au maximum les données extraites.

La reconnaissance de figure

Pour effectuer la reconnaissance de figure, nous avons dissocié la tâche en deux : la détection d'événement, et la classification des événements détectés. Cela permet de dissocier la performance de détection et de classification. De plus, cela permet de réduire la quantité de données à traiter par notre algorithme.

Les figures évaluées :

- Ollie
- Kickflip
- Heelflip
- Pop shovit
- Frontside shovit
- 360 flip

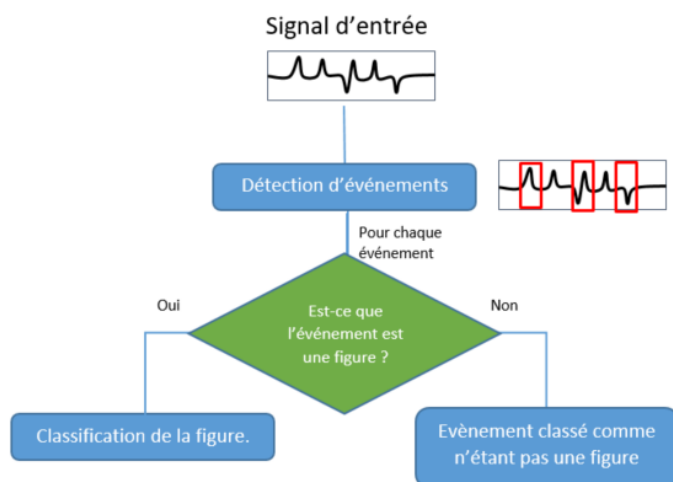


Schéma de l'algorithme

La détection d'évènement

Pour effectuer cette tâche, nous avons utilisé **l'énergie** présente dans les données d'accélération et gyroscopiques. En effet, des pics sont présents quand le skateur effectue une figure, car il apporte de l'énergie au skate. Cependant, le skateur peut aussi apporter de l'énergie au skate sans pour autant effectuer une figure (virage, pick-up...etc).

Tri des événements et classifications

D'une pierre 2 coups

Comme évoqués précédemment, beaucoup d'événements détectés n'étaient pas des figures. Cependant, notre méthode de classification permettait d'effectuer un tri. En effet, afin de savoir quelle figure correspondait à l'événement détecté, nous avons comparé ce dernier à **des figures de références**. La figure correspondant à l'événement était celle qui « ressemblait le plus à l'événement ». Dans notre cas, nous avons choisi la figure de référence qui minimisait la distance euclidienne avec l'événement. La figure était attribuée si la distance était en dessous d'un certain **seuil**, sinon elle était considérée comme un mouvement de transition.

Résultats

Détection d'événements

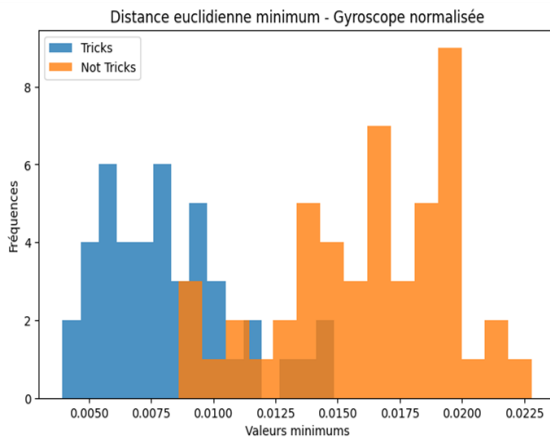
Pour nos différents enregistrements, nous avons détecté 162 événements. Sur ces événements, 113 correspondaient à des figures et 49 à des mouvements de transitions parasites.

N = 162	Positive	Négative
Positive	113 (TP)	0(FN)
Négative	49 (FP)	N/A

Nous avons donc une **sensibilité de 1** car nous détectons toutes les figures, mais une **précision de seulement 70%**.

Avec le tri effectué durant la classification, nous avons réussi à **améliorer les résultats de la détection en appliquant un seuil**.

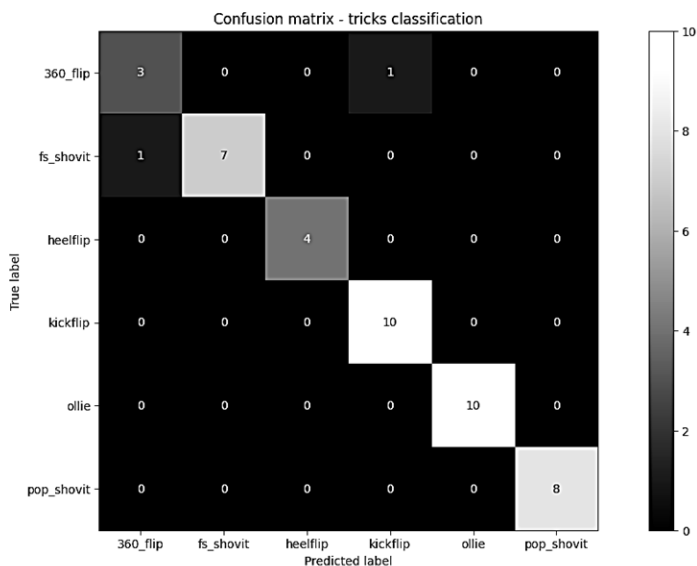
En effet, le seuil que nous avons choisi maximise le f1 score et permet d'obtenir en plus une **précision de 90.6%** et un **rappel de 87.7%**.



Histogramme des distances minimales des figures et des non-figures

Classification

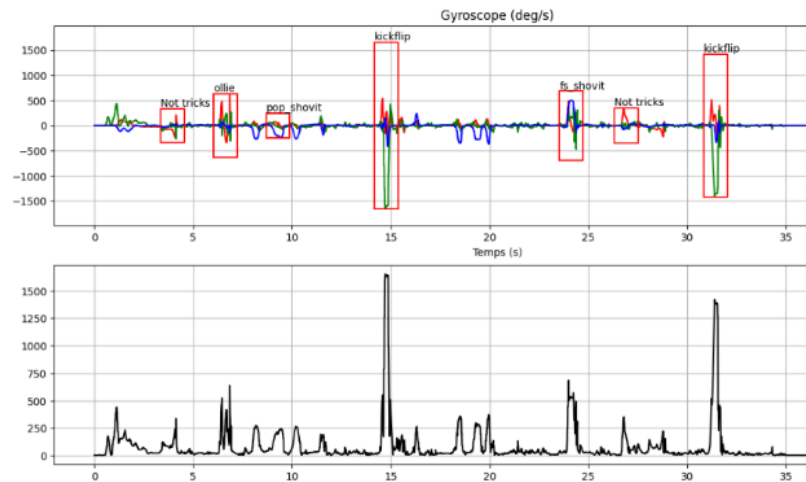
Sur les 6 figures choisies, notre algorithme de classification par minimisation d'une distance était très performant. Sur 44 figures testées, seulement 2 ont donné le mauvais résultat :



Matrice de confusion de la classification

La sortie de l'algorithme

Les performances de nos sous-algorithmes étant satisfaisantes, nous avons choisi de les tester sur une séquence de figures complètes, et nous avons obtenu en sortit :



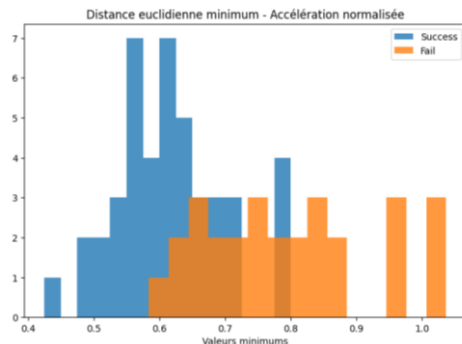
Sortie de l'algorithme de classification

On peut voir sur cette courbe les rectangles qui correspondent aux événements détectés. Nous avons mis la mention not_tricks sur les figures situées au-dessus du seuil. Nous pouvons remarquer la présence d'un pop-shovit, qui n'en est pas un. En effet, l'événement correspond à un virage, et possède une courbe similaire à celui du pop-shovit, ce qui fait que l'on a détecté comme tel. C'est une des limites de notes algorithme.

Succès d'une figure

Avec la même méthode du seuil, nous avons essayé de différencier les figures ratées et non ratées.

Voici l'histogramme représentant la distance minimale avec l'accélération pour les figures ratées et non ratées :



Histogramme des distances minimales des figures ratées et non ratée

On remarque que les deux groupes se séparent un peu. Un seuil peut donc être appliqué pour estimer si une figure est ratée ou non. En maximisant le f1 score, nous obtenons une précision de 80% et un rappel de 80% environ aussi. Ces résultats sont bons, mais à creuser, car nous ne prenons pas en compte le fait que la figure soit ratée connaissant la figure.

Limites de notre approche

Méthode trop « simple »

Perte d'informations en « fabricant » les figures de références, on évalue seulement la globalité de la figure

Beaucoup d'événements parasites

Malgré les améliorations dues au seuil, nous avons encore la présence d'événements parasites. Cependant, nous connaissons et pouvons classifier la plupart des événements, Penser à classifier les événements parasites seraient une première idée ?

Idées pour la suite

Tester l'algorithme avec d'autres figures

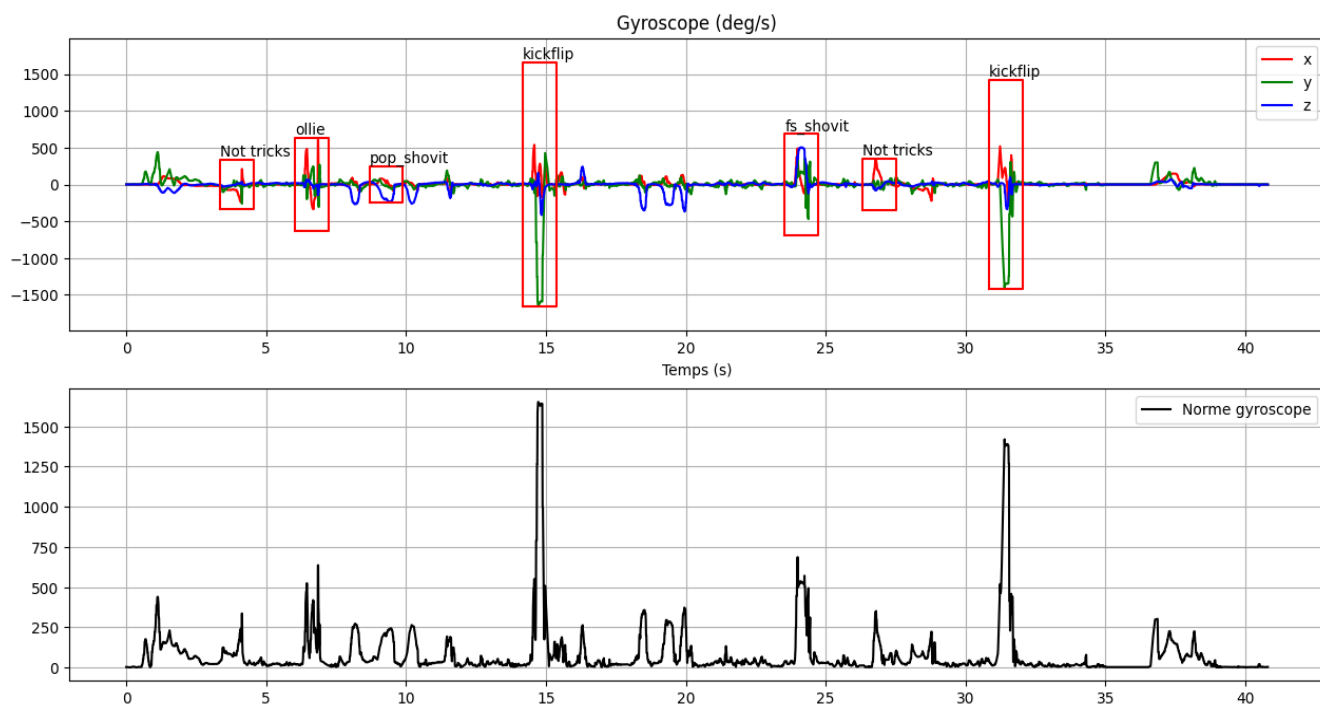
Tester des algorithmes plus performants

Etudier des figures impliquant la rotation du corps

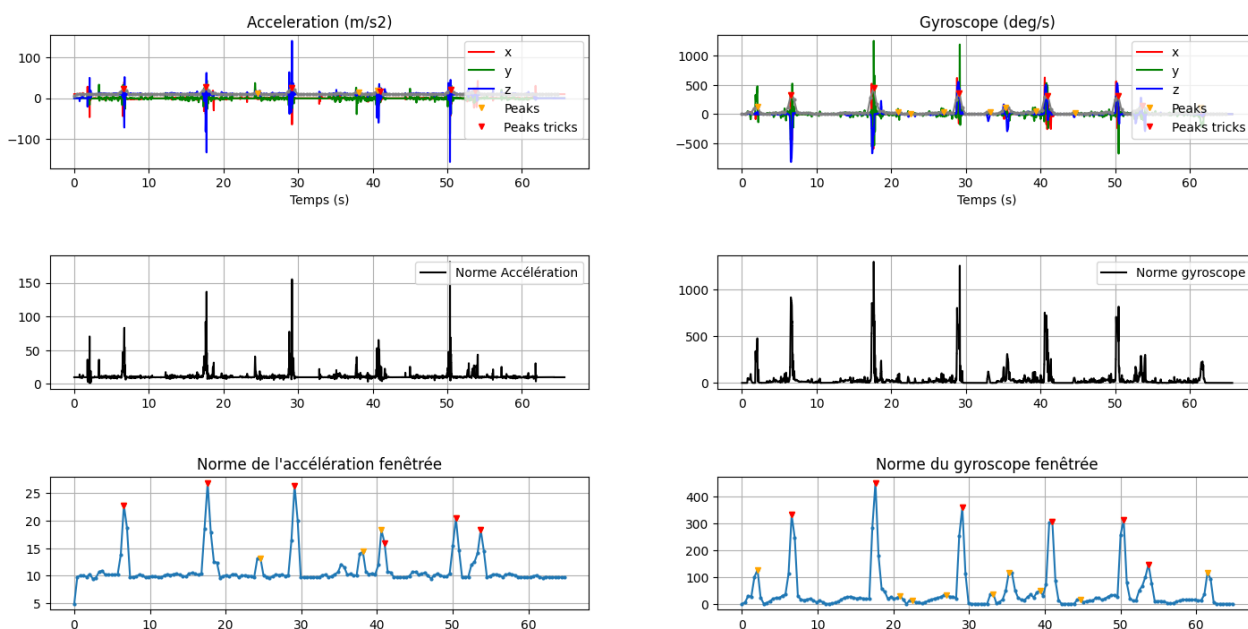
Etudier plus en profondeur le lien skate/données

Trouver des critères permettant d'évaluer des figures (timing, hauteur...)

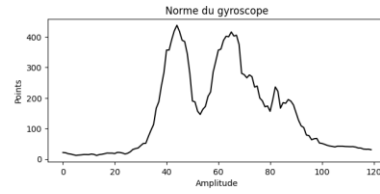
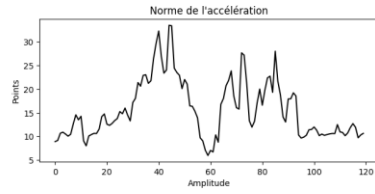
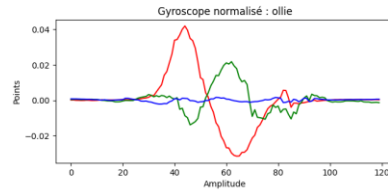
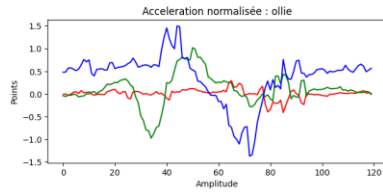
Annexes



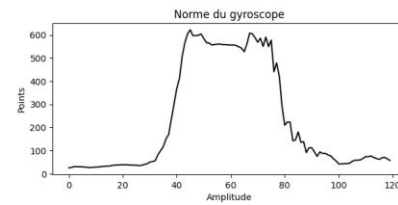
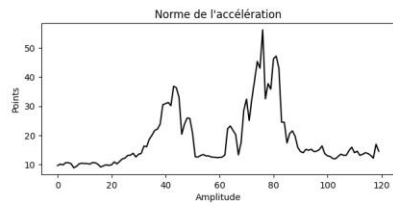
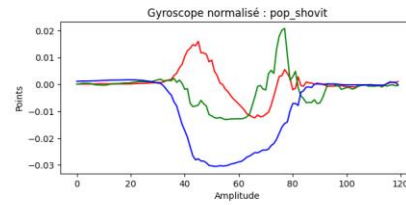
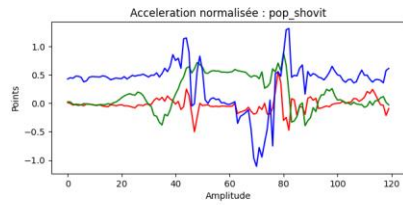
Sortie de l'algorithme - Zoom



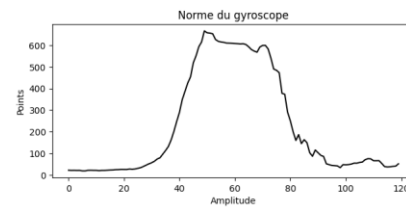
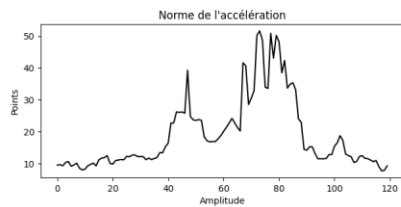
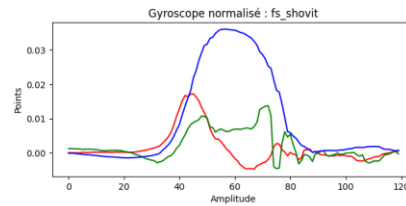
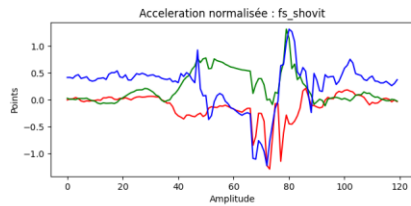
Sortie de l'algorithme de détection d'événements



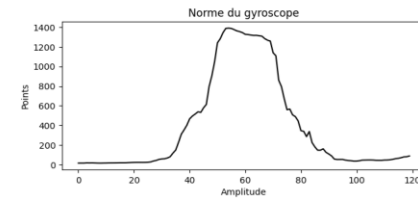
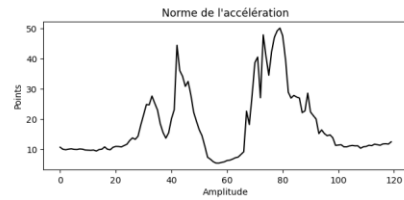
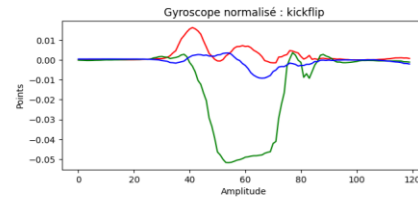
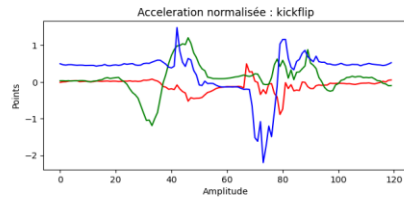
Ollie – Figure de référence



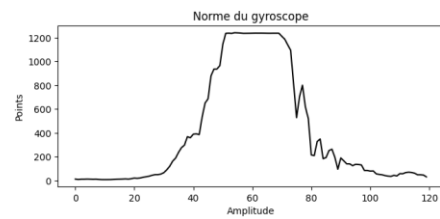
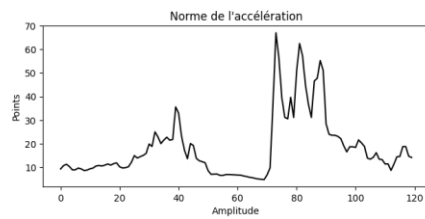
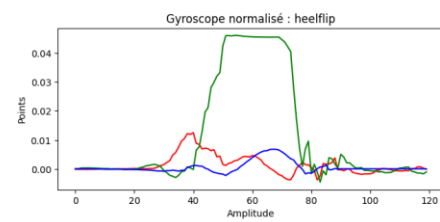
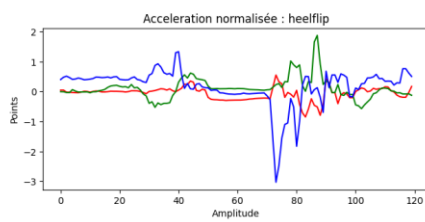
Pop shovit – Figure de référence



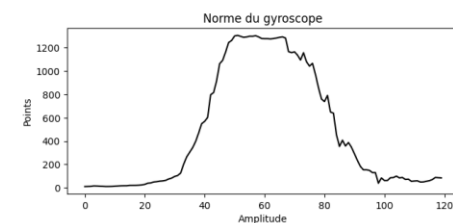
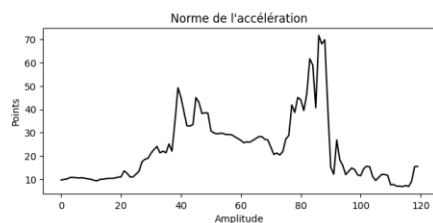
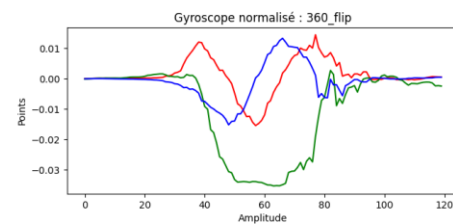
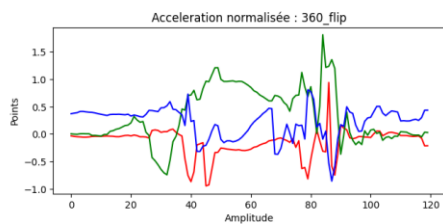
Fs shovit – Figure de référence



Kickflip- Figure de référence



Heelflip - Figure de référence



360 flip - Figure de référence