EMBRACe Expressive Movement Body fRAmework and Computing

Rapport de stage Encadrants : Fréderic BEVILACQUA et Sarah FDILI ALAOUI

Pierre MAHE

26 août 2016







Table des matières

1	\mathbf{Intr}	oducti	ion	5					
	1.1	Cadre	du Stage	5					
	1.2	Objec	tifs	5					
2	Éta	État de l'art							
	2.1	Forma	diser le mouvement dansé	7					
	2.2	Descri	pteurs du mouvement	7					
	2.3	Analy	se de l'expressivité	8					
	2.4	Intera	ction avec le geste	8					
3	Pré	sentati	ion de la théorie de Laban	9					
4	Pré	sentati	ion des bases de données	11					
	4.1	Préser	ntation d'un geste de Knocking	12					
	4.2	Segme	entation du geste en plusieurs Frappes	13					
5	Rec	herche	e d'une métrique pour comparer les différents Efforts	15					
	5.1		ipteurs de classification des Efforts	15					
	5.2		de la phase d'amorçage	16					
	5.3		de la phase de Frappes	17					
	5.4		ssion	17					
	5.5		ation des descripteurs	18					
	5.6		fication des Effort Indulging et Condensing	20					
6	Tra	Transformation Expressive : Du geste Neutre au geste Expressif 22							
_	6.1		thme basé sur la DTW	22					
	0	6.1.1	Fonctionnement de la DTW	23					
		6.1.2	Alignement Global entre deux gestes	23					
		6.1.3	Alignement par frappe	24					
		6.1.4	Transformation d'un geste Neutre à la variation	26					
		6.1.5	Résultats	27					
		6.1.6	Limitations	28					
	6.2	Aligor	rithme basé sur les HMM	30					
		6.2.1	Principe de l'algorithme	30					
		6.2.2	Fonctionnement des HMM pour le suivi de geste	31					
		6.2.3	Reconnaissance des Efforts par HMM	34					
		624	Transformation du neutre vers l'expressif par HMM	36					

		6.2.5 Résultats	37			
		6.2.6 Discussion	38			
		6.2.7 Validation des gestes générés par HMM	39			
7	Not	velle Expérience et Création d'un nouvel ensemble de données	42			
	7.1	Contenu du nouvel ensemble de données	42			
	7.2	Protocole expérimental	43			
8	Perspective et travaux futurs					
	8.1	Résumé du travail effectué	44			
	8.2	Étude de la composition des Efforts	45			
	8.3	Validation par synthèse de mouvement	45			
Appendices						
A	Annexe A Estimation de paramètres 4					

Résumé

Dans ce mémoire, nous présentons une approche pour étudier l'expressivité d'un mouvement à travers le formalisme de Rudolf Laban, analyste est théoricien de la danse. Laban analyse le mouvement selon huit Efforts pour décrire sa dynamique et ses intentions.

Dans une première partie, Nous présentons des descripteurs pour comparer un geste de Knocking effectué avec différents Efforts. En étudiant ces Efforts, dans une seconde partie, Nous présentons deux méthodes pour transformer un geste Neutre en un geste Expressif. La première méthode est basée sur l'utilisation de l'algorithme de Dynamic Time Warpping pour extraire une fonction de transformation entre deux gestes. La seconde est elle, basée sur l'utilisation de Modèles de Markov Cachés Multimodales. Dans une troisième partie, nous discuterons des résultats obtenus avec la base de données enregistrées pendant cette étude.

Mots clés:

Laban, qualité de mouvement, extraction d'expressivité, descripteurs du mouvement, geste, HMM, DTW

Abstract

This dissertation presents a new approach to study the movement expressiveness through Rubolf Laban Theory. Rudolf Laban was a dance theorist, who analyses movement along eight Efforts. The Efforts specially describe the movement's dynamic and qualities.

At first, we present a metric to compare Knocking gesture performed with Efforts. Furthermore, we present two methods to transform a neutral movement into an expressive movement. The first one is based on Dynamic Time Warping algorithm. It aims to extract transformation function between the two gestures. The second one is based on the Multimodal Hidden Markov Model. At last, we discuss about the results obtained with the new database recorded during this study.

Keywords:

Laban, movement qualities, expressivness analysis, movement descriptors, gesture, HMM, DTW

Chapitre 1

Introduction

Depuis quelques années, nous assistons à une émergence de dispositifs interactifs dans la création artistique notamment dans le domaine de la Danse. D'autre part, la communauté Interface Homme Machine (IHM) s'intéresse de plus en plus à caractériser les gestes et les mouvements expressifs.

L'analyse de l'expressivité fait l'objet de nombreuses recherches scientifiques afin de la comprendre et de la formaliser, dans le but de proposer de nouveaux types d'expression aux artistes.

Dans ce stage, nous nous sommes attachés à étudier d'une part, comment quantifier l'expressivité d'un geste et trouver des descripteurs significatifs pour classifier les différents Efforts, et d'autre part, de concevoir une méthode qui permette d'extraire cette expressivité et de l'appliquer sur un autre geste.

1.1 Cadre du Stage

Ce stage a été effectué dans le cadre du Master Acoustique, Traitement du signal et Informatique Appliqués à la Musique de l'Université de Pierre et Marie Curie, en partenariat avec l'IRCAM.

Il se déroule dans deux équipes de recherche : l'équipe ISMM (Interaction Son Musique Mouvement) à l'IRCAM et dans l'équipe HCC (human Centered Computing) au LRI à l'Université Paris Sud sous la direction de Frédéric Bevilacqua et Sarah Fdilli Alaoui. L'équipe ISMM mène des recherches et développements sur des systèmes interactifs musicaux. L'équipe HCC vise à développer de nouvelles techniques d'interaction Homme Machine. Une de ces branches mène des recherches spécifiques à l'interaction avec les danseurs.

1.2 Objectifs

Ce stage vise à étudier plusieurs approches comptutationnelles des Efforts de Laban pour caractériser l'expressivité dans le mouvement. Le but à terme est de créer un outil d'analyse de l'expressivité pour permettre de créer un système offrant une interaction temps-réel riche avec un danseur ou un musicien sur scène.

Les objectifs à réaliser au cours de ce stage sont :

L'élaboration d'une métrique permettant de catégoriser un geste selon son expressivité, en nous basant sur l'hypothèse considérant un geste expressif comme un ensemble décomposable en un geste neutre, plus une variation expressive. Cette variation expressive représentant la qualité du geste, au sens de Laban. Notre objectif est également de créer des méthodes permettant d'extraire la variation expressive d'un geste, dans le but de pouvoir appliquer cette expressivité sur un geste Neutre pour produire un geste expressif de synthèse.

Enfin dans une dernière partie, nous présentons des réflexions sur les travaux futurs sur le sujet. Le premier serait d'étendre les méthodes développées dans les sections précédentes pour d'autre type de gestes. La seconde serait de trouver une approche d'évaluation fiable pour valider les résultats obtenus dans les sections précédentes.

Chapitre 2

État de l'art

Dans cette section, nous examinons différents travaux dans le domaine de l'analyse du mouvement et de l'interaction. Ces travaux ont inspiré le travail réalisé durant ce stage.

2.1 Formaliser le mouvement dansé

Les avancées dans la reconnaissance des gestes ont permis des interactions entre ordinateur et danseurs, ouvrant de nouvelles perspectives. La nouvelle étape est d'offrir une interaction plus fine aux chorégraphes en développant des approches computationnelles permettant de capter l'expressivité dans le mouvement.

De nombreuses études ont été menées pour trouver un formalisme pour décrire l'expressivité d'un mouvement. Camurri et al dans sa première version de EyesWeb[3] a analysé l'expressivité du danseur en utilisant les formalismes développés par Kansei [4], formalismes utilisés à l'origine pour décrive les réponses émotionnelles d'un individu.

Dans son mémoire Y.Zeitoun [44] fournit une bonne vue d'ensemble des différents formalismes existants, notamment celui de Francois Delsarte et Hubert Godard, pour analyser le mouvement dansé. Dans son étude Y.Zeitoun arrive à la conclusion que le formalisme le plus abouti pour la captation de l'expressivité dans la danse est celui de Rudolf Laban. Ce formalisme a été repris dans de nombreuses études dans divers domaines touchant au mouvement, dans le domaine médical pour la rééducation [6] ou dans la classification d'activités humaines [23].

Dans le domaine de la danse des publications étudient le mouvement en utilisant également Laban. Comme celle de Khoshhal et al [38]qui utilise plusieurs notions de R.Laban pour classifier l'expressivité du mouvement.

Laban dans sa théorie définit huit Efforts. Ces efforts permettent de décrire l'ensemble des nuances de l'expressivité d'un mouvement.

Ces Efforts ont été étudiés dans plusieurs travaux notamment ceux de Chi et al [10]. Comme dans leur projet Emote, où ils tentent de trouver pour chaque Effort un descripteur.

2.2 Descripteurs du mouvement

Contrairement au domaine de la musique où il existe un ensemble de descripteurs admis permettant de décrire le son, le mouvement ne possède pas de tels descripteurs et chaque projet utilise ses descripteurs propres pour étudier le Mouvement selon les besoins et le contexte d'application.

Dans la seconde version de EyesWeb, Camurri et al [5] étudient le mouvement en analysant les relations entre les différents membres et la place du corps du danseur dans l'espace. Pour cela, ils ont défini des sous-régions du corps (bras, torse, jambe...) et ont analysé la forme de ces sous-régions et leurs liens entre elles. Dans l'étude de Françoise et al [19], des capteurs de contractions musculaires (MEG) ont été utilisés pour capter l'expressivité du danseur. En plaçant des capteurs sur différentes parties du corps (bras, jambes du danseur), ils ont pu analyser le "poids" et l'énergie mis dans le mouvement. Rett et al [37] ainsi que Chi et al [10] ont effectué un travail important sur l'analyse des angles des différentes articulations et sur la trajectoire des membres au cours du temps. Cette même approche est utilisée par Ran et al [36] qui va définir un descripteur dans le but d'effectuer une classification automatique de la dynamique des mouvements.

2.3 Analyse de l'expressivité

L'analyse computationnelle de l'expressivité est un domaine de recherche très actif encouragé par les résultats obtenus dans le domaine de l'extraction de l'émotion dans la voix. Les travaux de Dai et al [12], utilisent des descripteurs sonores pour reconnaître les émotions dans la voix et les classifier à l'aide de réseaux de neurones.

Des études dans le même but ont utilisé diverses méthodes de machine learning : des Hidden Markov Models (HMM) dans[39], Classification naïve bayésienne [41]. Fourati et al [14] dans leur article utilisent une idée similaire à Dai et al pour classifier des styles de marches. D'autre part des travaux ont essayé de décrire et de comprendre ce qu'est l'expressivité.

Zappi et al [43] dans leur étude ont analysé l'expressivité que pouvaient produire des musiciens confrontés à des instruments simples.

2.4 Interaction avec le geste

Dans le domaine de l'interaction, l'analyse de l'expressivité est au coeur de plusieurs travaux.

E.Ghomi dans sa thèse [22] s'est intéressé à établir une méthode pour concevoir des techniques d'interaction expressive pour la pratique musicale. Baptiste Caramiaux et al [8] ont développé une technique pour utiliser la variation expressive comme outil d'interaction. Fdilli Alaoui et al [1] ont développé une approche pour extraire l'expressivité en utilisant des notions de la théorie de Laban dans le but de créer un dispositif interactif artistique. Bevilacqua et al [2] ont créé un framework basé sur l'utilisation des HMM pour faire du suivi temporel de geste ainsi que de la synchronisation du geste avec du son dans le but de la sonification.

Francoise [15] dans sa thèse a effectué un raffinement de ce framework, il propose une approche appelée mapping par démonstration qui permet à un utilisateur de créer un couplage entre mouvement et son.

Chapitre 3

Présentation de la théorie de Laban

Dans le domaine de la danse, il existe plusieurs formalismes permettant de décrire un mouvement. L'un des plus populaires est celui de Rudolf Laban, théoricien de la danse du début du XXème siècle. En parallèle à la création d'un système de notation de la danse qui caractérise principalement les parties du corps en mouvement et leurs déplacements dans l'espace général et l'espace personnel, il a élaboré une manière d'analyser le mouvement.

Ce système d'analyse est utilisé pour décrire plus précisément les qualités et les intentions qui doivent être mises en oeuvre dans le mouvement. C'est ainsi qu'est née l'analyse du mouvement Laban, qui au départ était appelée l'Effort-Shape.

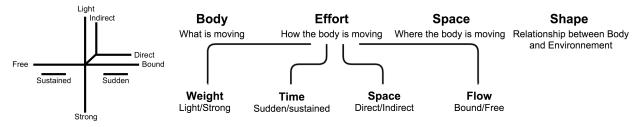
Pour cela, R.Laban caractérise le mouvement selon 4 catégories :

- Le Corps : ce qui bouge
- L' Effort : la dynamique et la manière dont le mouvement est exécuté
- L'Espace : où s'effectue le mouvement
- La Forme : La forme dynamique empruntée par le mouvement.

L' Effort est divisé en 4 facteurs qualitatifs. Chaque facteur est constitué d'un couple de deux éléments qui représente les deux extrêmes d'un continuum. Un extrême est qualifié d'indulging et l'autre condensing.

Weight Effort

Le facteur du Poids correspond au poids mis dans un mouvement. Les deux éléments



(a) Représentation des Efforts par Rudolf Laban (b) Système d'analyse des mouvements

de la composante du Poids sont : Strong (Condensing aspect) et Light (Indulging Aspect)

Time Effort

Le facteur de Temps correspond à l'évolution dans le temps du mouvement. Il peut être en accélération (Indulging aspect) ou en décélération (Condensing aspect). Les deux éléments de la composante de Temps sont : Quick (Condensing aspect) et Sustained (Indulging Aspect)

Space Effort

Le facteur d' Espace correspond à l'attitude qu'à une personne vis-à-vis de son environnement. La personne peut prêter attention à son environnement Indulging ou être concentrée sur quelque chose de précis Condensing . Les deux éléments de la composante d' Espace sont : Direct (Condensing aspect) et Indirect (Indulging Aspect)

Flow Effort

Le facteur de Flux correspond à la fluidité du mouvement ; Y a t-il une résistance au déplacement du mouvement ?

Les deux éléments de la composante du Flux sont :

Bound (Condensing aspect) et Free (Indulging Aspect)

En addition aux Efforts, Rudolf Laban créé les États et les Impulsions. Il définit un État comme une association de deux facteurs. Chaque État, possède donc quatre configurations d' Efforts. Par exemple, pour l' État Rhythm composé du facteur Weight et Time, quatre configurations sont possible : Strong-Quick, Strong-Sustained, Light-Quick, Light-Sustained. De manière homologue, Rudolf Laban définit l' Impulsion comme l'association de trois facteurs, donnant à chaque Impulsion huit configurations possible.

Chapitre 4

Présentation des bases de données

Dans cette étude, nous avons utilisé des gestes de "Knocking" provenant de deux bases de données, la base de données MoDa et une nouvelle base de données enregistrée durant ce stage.

La base MoDa

La base MoDa a été enregistrée dans le cadre du projet Movement + Meaning Middleware Project à Simon Fraser University.

La base de données est constituée d'enregistrements de motion capture Vicon de gestes de « Frappe à la porte » (Knocking) effectués avec plusieurs variations expressives. La performance est faite par une danseuse mimant le geste de Knocking.

Les gestes de Knocking sont effectués de manière :

- Neutre
- Avec une variation d' Effort
- Avec des variations de forme dynamique (Shape Qualities)
- Avec des variations de localisation spatiale
- Avec des émotions particulières.

La base MoDa est intéressante par sa diversité de variations expressives, mais ne possède que peu de répétitions de chaque variation.

La nouvelle base de données

Cette base a été enregistrée durant ce stage. Elle est également constituée de gestes de Knocking effectués avec plusieurs variations expressives. Les enregistrements ont été faits grâce à une Kinect V2. Plus de détails sur l'expérience et le protocole expérimental sont disponibles dans la section 7.

Dans cette base, les gestes de Knocking sont effectués de manière :

— Neutre

- Avec un Effort appliqué au geste
- Avec deux Efforts appliqués au geste
- Avec trois Efforts appliqués au geste

Cette nouvelle base a permis d'avoir un plus grand nombre de répétitions de chaque geste, ce qui a permis l'utilisation d'algorithmes de machine learning ainsi que des traitements statistiques des résultats.

4.1 Présentation d'un geste de Knocking

Les enregistrements Vicon fournissent la position au cours du temps de marqueurs posés sur le danseur. Le taux d'échantillonnage de 100 Hz et les signaux peu bruités permettent d'avoir un bon suivi temporel du mouvement et de percevoir les variations propres aux Efforts.

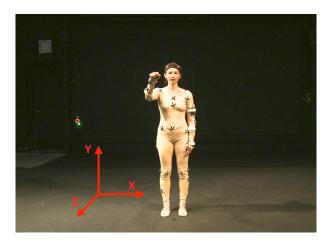


FIGURE 4.1 – Une participante équipée de marqueurs de Motion Capture. En rouge, la direction du repère utilisé par la Vicon pour situer les marqueurs. L'origine du repère est placée sur le marqueur correspondant à la vertèbre thoracique T10 (Base du dos).

Le mouvement de Knocking n'engageant pas le corps entier, certains marqueurs n'apportent pas d'information significative. Seuls les marqueurs du bras effectuant l'action, le bras droit, ont été utilisés. Les marqueurs utilisés sont : le Marqueur du coude, du poignet et du dos de la main.

Sur la figure 4.2 est représentée l'évolution au cours du temps de la position du marqueur placé sur le poignet du danseur selon les trois axes X, Y, Z. L'origine du repère utilisé correspond au marqueur sur la vertèbre thoracique T10. Par rapport au corps les trois axes du repère correspondent au plan Frontal pour l'axe X, au plan Sagittal pour l'axe Y et au plan Transverse pour l'axe Z. Le geste de Knocking est un geste où le mouvement est fait majoritairement selon le plan Sagittal et Transverse (Y et Z).

Sur les courbes sur l'axe Y et Z, trois portions se découpent : La portion de la montée du bras, qui correspond au passage de la position "bras le long du corps" à la position de Frappe. La position de Frappe correspondant à la position débout, le poignet au niveau de

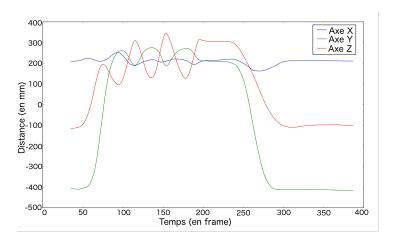


FIGURE 4.2 – Évolution au cours du temps de la position du marqueur placé sur le poignet du participant sur les trois axes X, Y et Z.

l'épaule, le coude plié.

Le seconde portion correspond aux multiples frappes. Sur la figure, elle correspond aux ondulations où chaque ondulation est une frappe. Le poignet décrit une trajectoire parabolique jusqu'au point de frappe puis revient à la position de Frappe.

La dernière portion est le mouvement antagoniste de la portion de la monté du bras. Cette portion correspond au passage du bras de position de frappe à la position le bras le long du corps.

La vitesse, l'accélération et le jerk ont été extraits à partir des données de position des marqueurs, par l'utilisation de l'algorithme de Savitzky-Golay. Cet algorithme est basé sur l'utilisation d'un polynôme pour approximer la courbe dans une fenêtre de temps donnée. Cela permet de lisser le signal et d'en extraire les dérivées successives en limitant la sensibilité aux erreurs de calculs et aux bruits de mesures.

Pour l'algorithme de Savitzky-Golay, le polynôme est de forme cubique et la largeur de fenêtre d'analyse est de 9.

4.2 Segmentation du geste en plusieurs Frappes

Kendon [31] propose un formalisme pour décrire les phases d'un geste : la phase de préparation, la partie centrale du mouvement et la dernière phase de relâchement. J.Francoise dans sa thèse reprend une idée similaire pour le découpage du mouvement.

Dans le geste de Knocking, les trois phases sont découpées comme suivant, voir figure 4.3 : La phase de préparation correspond au départ du bras, jusqu'à la position de frappe. La phase centrale correspond aux multiples frappes et la phase de relâchement correspond au retour de la position de frappe à la position initiale.

Pour délimiter précisément ces trois phases, nous avons calculé la dérivée spatiale de la vitesse (vitesse en fonction de la variation de la position).

Quand la variation de la position est nulle et que la vitesse ne l'est pas, alors une discontinuité se forme dans la courbe de la dérivée spatiale. Ces moments correspondent à l'instant de la cassure dans le geste (un changement de direction, un arrêt ...). En recherchant ces points de discontinuité, il est possible de déterminer les segments. La méthode a le défaut d'être trop sensible et de sur-segmenter le geste. Une correction manuelle est donc obligatoire pour ne garder que les trois phases trois du geste.

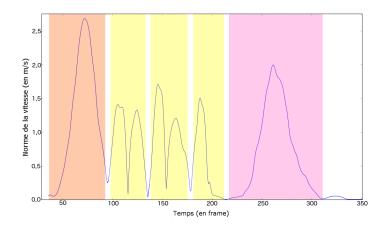


FIGURE 4.3 – Représente le geste de Knocking segmenté en phase : l'amorçage en orange, les trois frappes en jaune et le geste de retour en violet.

La figure 4.3, représente la norme vitesse du poignet en fonction du temps. Les différentes phases du geste sont représentées. La phase de frappe, en jaune, est constituée de trois frappes. Chaque frappe est constituée de deux pics, le premier correspond à l'avancée du bras vers le point de frappe et le second correspond au retour du bras vers sa position de frappe initiale.

Il n'y a qu'un pic pour la dernière frappe car le participant est passé directement du point de frappe à la position bras le long du corps sans revenir à la position de frappe initiale, supprimant le pic correspondant au retour du bras vers sa position de frappe initiale.

Chapitre 5

Recherche d'une métrique pour comparer les différents Efforts

Dans cette étude, nous avons effectué une série de tests sur un ensemble des descripteurs permettant de caractériser les Efforts. Ces descripteurs ont été directement inspirés par la théorie de Laban, malgré le fait que nous nous limitons à un geste unique dans notre étude. L'idée est d'appliquer la métrique résultante à des mouvements complexes du corps entier.

5.1 Descripteurs de classification des Efforts

Dans une première approche, nous évaluons une série de descripteurs pouvant caractériser chaque effort.

Chi et al [10] dans son article cherche à trouver un descripteur unique pour chaque Effort. Trouver un unique descripteur pour rendre compte de chaque Effort est quelque chose de difficile, dû à la complexité de chaque Effort.

Laban décrit chaque Effort comme un composante à plusieurs aspects, c'est pour cela que l'utilisation de plusieurs descripteurs pour rendre compte de chaque Effort parait plus adaptée.

Pour catégoriser les différents Knocking de la base de données, notre approche a été d'étudier un ensemble de descripteurs dynamiques pour approximer chaque Effort.

Liste des descripteurs significatifs :

Pour chaque geste, chaque axe a été analysé séparément (X,Y,Z) et les calculs des descripteurs ont été faits sur chaque composante de la position, la vitesse, l'accélération et le jerk.

Le Jerk est la dérivée du vecteur accélération, il correspond aux à-coups ou saccades dans le mouvement.

$$\vec{j} = \frac{\mathrm{d}\vec{a}}{\mathrm{d}t} = \frac{\mathrm{d}^2 \vec{v}}{\mathrm{d}t^2} = \frac{\mathrm{d}^3 \vec{r}}{\mathrm{d}t^3} \tag{5.1}$$

Avec a l'accélération, v la vitesse et t le temps.

Les descripteurs utilisés sont : l'énergie du signal, la durée du geste, mean-crossing, l'indice de cumul de 25%, 50%, 75% de l'énergie totale du signal, le nombre de pics dans le signal, le centroïde spectral (spectral centroid) ainsi que la platitude spectrale (spectral flatness).

L'énergie du signal Es est défini comme :

$$E_s = \int_{-\infty}^{\infty} |x(t)|^2 dt \tag{5.2}$$

Avec Es l'énergie du signal x(t) où x est un signal variant au cours du temps.

L'indice de cumul de l'énergie est défini comme l'indice de frame F où l'égalité suivante n'est plus vérifiée :

$$P * E_s \ge \sum_{n=0}^{F} |x[n]|^2 \tag{5.3}$$

Avec Es l'énergie du signal x[n] où x est un signal variant au cours du temps. Et P comme le pourcentage de l'énergie du signal désiré.

Dans cette étude nous appellerons Énergie de la vitesse, l'énergie du signal calculée pour la composante de la vitesse. Il sera fait de même pour l'accélération et le jerk.

Les descripteurs étant nombreux, nous avons choisi pour chaque figure d'utiliser les descripteurs illustrant le mieux la propriété que nous avons voulu mettre en lumière, bien que la propriété soit visible sur plusieurs autres descripteurs.

Du point de vue de la vitesse, les trois phases du mouvement ne sont pas du même ordre de grandeur. Par exemple pour un Knocking Neutre, la phase d'amorçage a une énergie approximativement 8 fois plus importante que la phase de Frappe. Pour cette raison, les trois phases ont été analysées de manière séparée.

5.2 Étude de la phase d'amorçage

Il est remarquable que certains Efforts se distinguent de la phase d'amorçage du geste, alors que les différentes phases d'amorçage sont très proches entre elles, du point de vue de la vitesse et de l'accélération. Voir exemple de la figure 5.1

La classification des gestes en fonction des différents Effort est difficile. Pourtant comme le montre la figure, il est possible de classifier les Efforts : Strong, Quick et Bound. Pour les autres Efforts, il n'y a pas de groupe distinct. Les gestes sont trop similaires pour permettre un quelconque regroupement.

Il n'est pas possible de faire ressortir des groupes de plusieurs Efforts, comme par exemple, les Efforts Indulging et Condensing de la théorie de Laban.

Pour rappel le groupe Indulging regroupe les Efforts : Light, Quick, Free, Indirect et le groupe Condensing regroupe les Efforts : Strong, Susteined, Bound, Direct.

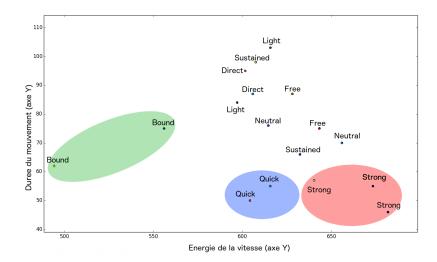


FIGURE 5.1 – Projection des phases d'amorçage de Knocking effectués avec différents Efforts. Les deux descripteurs utilisés sont l'énergie de la vitesse sur l'axe Y et la durée de la phase.

5.3 Étude de la phase de Frappes

En utilisant les descripteurs énoncés précédemment, il est possible de classifier les phases de Frappes des gestes de Knocking.

Sur la figure 5.2 Nous avons représenté la phase de frappe de Knocking effectuée avec différents Efforts en fonction de leur vitesse (somme de la vitesse au cours du temps) et la durée de la phase de frappe.

Avec ces deux descripteurs, il est possible de regrouper tous les Efforts. Seuls les Efforts Free et Sustained ne peuvent être regroupés.

5.4 Discussion

Les deux analyses montrent que l'information pour décrire et classifier les Efforts est principalement contenue dans la phase de Frappe. Sur cette phase, il est possible avec peu de descripteurs d'avoir une première classification des différents Efforts.

Dahl Sofia dans son étude [11] sur le geste d'un joueur de Batterie avait fait l'observation qu'il était possible de manière computationnelle de prédire la force de frappe du musicien à partir de sa phase d'amorçage.

D'après notre analyse de la phase d'amorçage, cette propriété est également vraie pour certains Efforts; Strong, Quick, Bound.

Sur l'analyse de la phase de Frappe, deux Efforts Sustained et Free ne sont pas regroupés. Dans la théorie de Laban un geste Sustained est décrit comme un mouvement en décélération.

L'utilisation de descripteurs analysant l'ensemble du geste n'est donc pas une solution

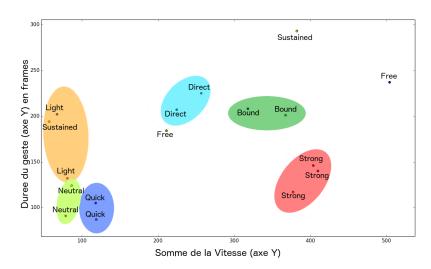


FIGURE 5.2 – Phase de frappes de différents Knocking effectués avec différents Efforts, en fonction de la vitesse et de la durée de la phase de frappe.

optimale. L'ajout de descripteurs rendant compte de l'augmentation ou de la diminution de l'amplitude d'une valeur (l'accélération par exemple) permettrait une meilleure classification les Efforts Sustained.

Malgré cela, la classification n'est pas parfaite car pour un mouvement de Knocking, il est possible d'appliquer l'Effort Sustained de deux manières différentes : la première est de voir le geste comme un seul mouvement et donc ralentir le mouvement de manière continue sur l'ensemble du geste. La seconde est de voir le geste comme une suite de mouvement de frappes, où chaque frappe commencerait avec la même vitesse puis ralentirait. Le choix entre les deux possibilités est laissé à l'interprète. Cela explique également la difficulté à classifier les gestes Sustained.

Le mouvement de Free est lui, lié à la contraction musculaire. Les études précédentes [19] utilisaient des capteurs MEG (electroneuromyogramme). Ne disposant pas de telles mesures, nous avons décidé d'utiliser la variation de l'angle du poignet pour distinguer les Efforts Free et Bound. Comme expliqué dans la section sur la théorie de Laban [ref section], les gestes Bound sont des mouvements où tous les muscles sont contractés ce qui verrouille l'articulation du poignet. Au cours du temps, la variation d'angle formée par l'articulation, va être minimale.

Au contraire un geste Free va être un geste souple avec une grande variation de l'angle formé par l'articulation. En utilisant cette variation d'angle, il est possible de discriminer les deux Efforts.

5.5 Validation des descripteurs

Pour valider la pertinence des descripteurs trouvés dans la section précédente, nous avons utilisé un algorithme d'apprentissage automatique : les K-nearest neighbors (KNN).

L'idée est de faire classer au KNN les différents gestes selon leurs Efforts en utilisant les descripteurs trouvés manuellement. De plus, une estimation de paramètres permet de déceler des descripteurs peu pertinents ou redondants.

Fonctionnement des Knn

L'algorithme des K-nearest neighbors fonctionne en deux phases, la première consiste en une phase d'entraînement où l'algorithme prend en entrée des couples de : données (à D dimensions) et un label de classe. Dans notre cas, les données sont les D descripteurs d'un geste et le label est l'Effort associé. L'algorithme va associer à chaque classe une portion de l'espace (à D dimensions). Durant la phase de classification, l'algorithme prend un geste test dont l'effort est inconnu, puis place ce geste dans l'espace des descripteurs. En étudiant la portion de l'espace où le geste est placé, il est possible de déterminer sa classe.

Principe général

Pour l'algorithme des Knn, nous avons utilisé l'implémentation de la bibliothèque scikit-learn (scikit-learn.org).

Pour étudier si certains descripteurs ne sont pas redondants ou peu pertinents, nous avons fait une évaluation de paramètres. Notre méthode consiste à entraîner le Knn avec un sous ensemble de descripteurs puis de lui faire classer des gestes. À la fin de cette classification, la proportion de bonne classification par rapport au nombre total de gestes à classer est calculée. Ce résultat nous donne le score d'accuracy.

En comparant l'accuracy obtenue pour tous les sous-ensembles de paramètres, il est possible de connaître les descripteurs permettant une reconnaissance maximale.

Pour plus de robustesse, notre méthode calcule l'accuracy moyenne en itérant 500 fois le calcul de l'accuracy pour chaque sous-ensemble de paramètres.

Calcul de l'Accuracy

Le sous-ensemble de paramètre est sélectionné dans l'ensemble des descripteurs. L'ensemble des descripteurs sont : durée, énergie de la vitesse, mean crossing, spectral centroid, spectral flatness, duration, l'indice de cumule de 25%, 50% et 75% de l'énergie. Ces descripteurs sont calculés sur les 3 axes (X, Y, Z) ce qui nous donne au total un ensemble de 27 descripteurs.

Pour chaque sous-ensemble de paramètres (configuration de paramètres), 90% des gestes de chaque Effort sont utilisés pour entraîner le Knn. Les 10% restant servent pour la phase de classification. À chaque itération, les gestes sont séparés aléatoirement entre les deux groupes. Les descripteurs optimaux déterminés par notre méthode d'évaluation de paramètres correspondent à l'ensemble des descripteurs privés de : la spectral flatness sur les trois axes, l'énergie sur X et Z ainsi que l'indice de cumul de 25% de l'énergie sur Y. Ce qui nous donne 21 descripteurs.

Il est toutefois à noter que d'autres configurations arrivent à une accuracy moyenne proche de l'accuracy optimale. Il y a donc de la ressemblance entre les descripteurs.

Le tableau 5.3 dresse un comparatif entre différentes configurations. La première est une configuration naïve avec comme uniques descripteurs l'énergie selon les trois axes et la durée du geste. Malgré ce côté simpliste, il est possible de réussir à classer un geste test correctement entre les 8 efforts possibles dans 56,7% des cas. La seconde configuration est celle des descripteurs déterminés manuellement. La troisième correspond à la configuration déterminée par l'estimation de paramètres. Avec ces descripteurs le Knn réussit à classifier un geste dans 84,0% des cas.

Descripteurs	Accuracy Moyenne	
Energie et Durée	0,567	
Tout les descripteurs	0,789	
Descripteurs determiné par l'algorithme	0,840	

FIGURE 5.3 – Accuracy moyenne obtenue avec avec trois configurations différentes : avec deux descripteurs Énergie de la vitesse et Durée, avec tous les descripteurs et le sous-ensemble optimal déterminé par notre algorithme.

5.6 Classification des Effort Indulging et Condensing

Dans cette partie, nous présentons une observation inattendues découverte durant les recherches sur les descripteurs. Cette observation met en lumière un lien entre les Efforts et d'autres notions de Rudolf Laban.

Cette section en fait une première analyse mais cette observation mériterait une étude plus approfondie.

Pour rappel, Laban découpe les Efforts en deux groupes qu'il nomme : Indulging et Condensing.

Le groupe Indulging correspond aux Efforts : Free, Quick, Indirect et Free. Le groupe Condensing correspond aux autres Efforts : Strong, Sustained, Direct et Bound.

Durant la recherche de la métrique, une caractéristique intéressante semble ressortir. Sur de multiples représentations, comme sur celle de la figure 5.4 les Efforts forment deux groupes distincts. Cette limite entre les deux groupes est représentée sur la figure par la droite rouge. Un groupe est composé de Light, Quick et un autre de Strong, Bound et Direct.

Les difficultés de classification des Efforts Free et Sustained, mentionnées dans la section 5.1 ne permettent pas de savoir dans quel groupe ils se situent.

La composition des deux groupes formés est proche des groupes Indulging et Condensing décrits par Laban. Même si cela n'est pas mentionné dans la théorie de Laban, cette observation laisse penser que la séparation en deux groupes faits par R.Laban correspond à une réalité physique.

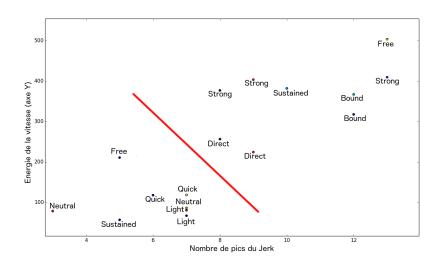


FIGURE 5.4 – Les Efforts sont divisés en deux groupes qui correspondent aux groupes Indulging et Condensing formalisés par Laban. La droite rouge matérialise la séparation entre deux groupes.

Chapitre 6

Transformation Expressive : Du geste Neutre au geste Expressif

Grâce à la Métrique élaborée dans la section précédente, pour comparer les différents Efforts, il est maintenant possible d'étudier comment passer d'un geste Neutre au même geste avec un autre Effort.

Nous sommes partis de la même hypothèse que formule Cammurri [5]. Dans sa publication, il décrit un geste expressif comme un geste neutre auquel serait appliqué une transformation expressive. Dans notre cas, la transformation expressive serait un Effort de R.Laban.

L'idée principale est de calculer une fonction de transformation entre le geste Neutre et un geste Expressif, puis dans un second temps, de l'appliquer à un geste cible pour générer un geste cible auquel un Effort aura été appliqué.

À l'aide des descripteurs développés dans la section précédente, il sera possible de comparer le geste cible avec le geste de synthèse.

Pour étudier cette transformation expressive, deux méthodes ont été implémentées, la première utilisant l'algorithme de Dynamic Time Warping (DTW) pour calculer la différence entre les deux gestes, la seconde utilisant les Hidden Markov Model développés durant la thèse de J.Françoise [15].

6.1 Algorithme basé sur la DTW

La première méthode consiste à soustraire la courbe du geste Neutre au geste Expressif pour récupérer la transformation entre les deux puis approximer par une fonction cette transformation, et appliquer cette fonction sur un autre Knocking.

Les gestes n'ayant pas tous la même durée et donc pas le même nombre d'échantillons, il n'est pas possible de comparer deux mouvements échantillon à échantillon. Pour rendre cette opération de comparaison possible, nous avons utilisé un algorithme d'alignement le Dynamic Time Warpping.

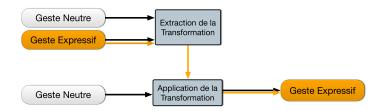


FIGURE 6.1 – Principe de l'algorithme

6.1.1 Fonctionnement de la DTW

La déformation temporelle dynamique ou Dynamic Time Warpping (DTW) est un algorithme permettant d'aligner deux séquences de longueur différente en minimisant la distance entre ces deux séquences.

L'algorithme de DTW fait partie des algorithmes de programmation dynamique, c'est à dire que l'algorithme va tenter de trouver la solution optimale pour un sous-problème puis progressivement va déduire la solution pour l'ensemble du problème.

L'algorithme fonctionne en deux phases. Dans la première, l'algorithme va calculer pour chaque couple d'échantillons i et j, respectivement échantillons de la séquence 1 et de la séquence 2, la distance entre les deux valeurs. À cette distance va être additionnée la plus petite distance entre les valeurs calculées pour les couples : {i-1, j}, {i-1, j-1}, {i, j-1}.

Une fois tous les couples {i, j} calculés, l'algorithme va partir de la fin des deux séquences. À chaque pas d'avancement, l'algorithme choisit le couple de coût minimum entre les couples : {i-1, j}, {i-1, j-1}, {i,j-1}. L'opération est réitérée jusqu'à arriver au début des deux séquences. L'ensemble de couples {i, j} traversés correspond à l'alignement optimal des deux séquences.

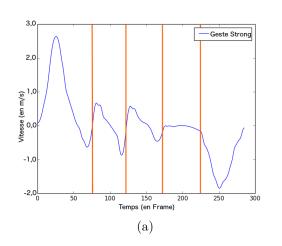
Pour de plus amples informations sur la DTW, je vous renvoie à l'article de Young-Seon Jeong et au [ref vers le papier Weighted dynamic time warping for time series classification].

6.1.2 Alignement Global entre deux gestes

Dans l'optique de pouvoir déterminer une fonction de transformation entre deux gestes en calculant la variation entre les deux, Nous avons appliqué l'algorithme de la DTW sur l'ensemble de la longueur des courbes.

Sur la figure 6.3 Nous constatons que l'algorithme de DTW a aligné les deux gestes de manière optimale du point de vue de la distance entre les deux courbes. Par contre, d'un point de vue des différentes phases, l'alignement n'est pas convaincant car il ne fait pas correspondre les bonnes phases des deux gestes.

Les phases des gestes de Knocking Neutre et Strong ont été représentées sur la figure 6.2. Sur la figure 6.3, la DTW a regroupé le segment d'amorçage et la première frappe du Neutre pour l'aligner avec la phase d'amorçage du Strong, puis va aligner les segments de manière décalée (le second segment avec le premier segment du Strong...). Avec cet



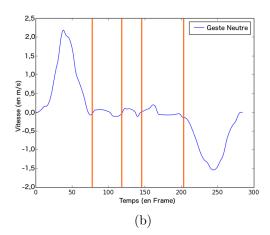


FIGURE 6.2 – À gauche, la vitesse sur l'axe Y d'un geste de Knocking effectué de manière Neutre. À droite, la vitesse sur l'axe Y d'un geste de Knocking effectué avec l'Effort Strong. Les barres verticales oranges délimitent les différentes phases du Knocking : phase d'amorçage, 1ère frappe, 2ème frappe, 3ème frappe et retour.

alignement le découpage par phases est complètement perdu.

La principale cause de ces erreurs est due à la différence d'ordre de grandeur de la vitesse selon les phases du mouvement. L'algorithme aligne en priorité les pics importants (pic de l'amorçage et du retour par exemple) car se sont les éléments les plus significatifs pour le calcul de la distance. Ce comportement est inhérent à l'algorithme de DTW. Pour améliorer cet alignement, notre solution a été d'ajouter un raffinement à l'algorithme, permettant de faire un alignement par phase.

6.1.3 Alignement par frappe

Pour ne pas perdre d' information sur les phases, l'alignement n'est pas fait sur le geste entier, il est effectué sur chaque phase de manière séparée.

Pour cela, il a fallu découper le geste en segment correspondant à chaque phase du geste. La phase de Frappe a été découpée en plusieurs segments chacun représentant une frappe. Un geste se décompose en cinq segments : amorçage, 1ère frappe, 2 ème frappe, 3 ème frappe et retour.

La transformation entre segment Neutre et segment Expressif correspond à une déformation d'amplitude et un déformation temporelle. Ces deux aspects ont été analysés séparément.

Transformation d'amplitude

Chaque segment d'un geste est aligné avec le segment correspondant d'un autre geste à l'aide de la DTW.

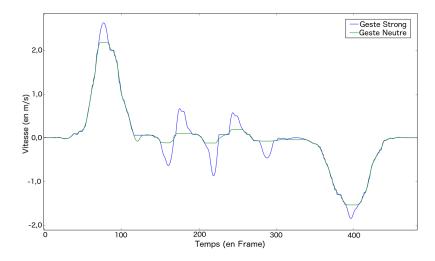
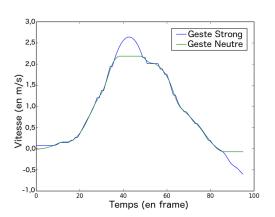
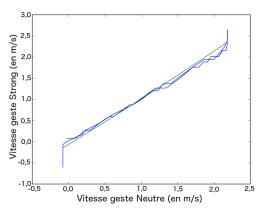


FIGURE 6.3 – Vitesse d'un Knocking Neutre avec un Knocking Strong après alignement.



(a) Deux phases d'amorçage respectivement d'un geste Neutre et Strong alignés.



(b) La vitesse du geste Neutre en fonction de la vitesse du geste Strong (en Bleu) et la régression polynomiale de degré 2 associé (en Vert).

La courbe de la figure 6.4b, représente la vitesse au cours du temps du geste Neutre en fonction de la vitesse du geste Strong. Chaque valeur du geste Neutre va être placée sur l'axe X et chaque valeur du geste Strong sur l'axe Y. Grâce à cette courbe, il est possible de connaître pour chaque valeur du geste Neutre, la valeur associée pour le geste Expressif.

Pour le segment d'amorçage et la phase de retour, la courbe formée est très proche d'une fonction linéaire, comme cela est montré sur la figure 6.4b.

Pour les segments de frappe, la fonction est plus proche d'un polynôme de degré 3. Pour approximer la courbe de transformation, une régression polynomiale a été appliquée afin d'extraire une fonction. Grâce à cette fonction polynomiale, il est possible d'appliquer un Effort à un segment de geste.

Transformation temporelle

Le même geste effectué avec des efforts différents n'ayant pas la même durée, il paraissait essentiel de prendre en compte la durée de chaque effort en calculant un facteur d'élongation entre la section neutre et la section expressive.

Pour appliquer la transformation temporelle sur un geste cible, l'algorithme effectue un ré-échantillonnage sur le geste avec le facteur d'élongation calculé.

$$Facteur_{\'{e}longation} = \frac{Geste_{Expressif}}{Geste_{Neutre}}$$
 (6.1)

Avec $Geste_{Expressif}$ et $Geste_{Neutre}$ respectivement la longueur du geste Expressif et du geste Neutre.

6.1.4 Transformation d'un geste Neutre à la variation

Pour extraire la transformation sur un geste entier, l'algorithme découpe le geste en plusieurs segments que nous avons analysé isolément avec la méthode de la section précédente 6.1.3. Cela permet d'avoir un alignement correct des différentes phases ainsi qu'une fonction régression polynomiale cohérente sur le segment. Ensuite, l'algorithme utilise les fonctions calculées pour générer pour chaque portion du geste un nouveau segment et concaténer tous les segments générés pour reconstruire un geste entier.

Tous les Knocking de la base de données n'ont pas le même nombre de frappes (certains des Knocking ont trois frappes, d'autres quatre). Le nombre variable de frappes pose problème pour le calcul des fonctions de transformation. Il faut déterminer à quelle frappe correspond la frappe supplémentaire.

Cette question renvoie à la problématique de savoir comment déterminer automatiquement les deux segments à associer pour calculer la fonction de transformation.

Associer les segments des deux gestes

Un algorithme a été crée pour déterminer quels segments associer entre deux gestes. L'algorithme représente chaque geste comme un automate, avec un état pour chaque segment. Un des deux gestes est considéré comme le geste de référence et l'autre comme le geste observé. Dans notre cas, le geste de référence correspond au geste Neutre et le geste observé correspond lui au geste Expressif. Pour l'automate du geste observé, chaque état a une seule transition, celle permettant de passer au segment temporel suivant (de l'amorçage vers la 1ère frappe, de la 1ère frappe vers la seconde frappe ...). Cette topologie garantit que tous les segments du geste observé seront associés à un segment du geste de référence.

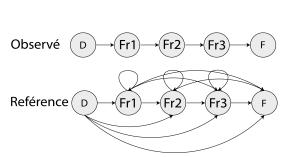
Pour l'automate du geste de référence, chaque état a une transition avec les états représentant un segment suivant. L'état d'amorçage est lié à tous les autres états, l'état de la première frappe est lié à tous les autres états sauf celui d'amorçage. Pour chaque automate, l'état de départ correspond au segment d'amorçage et l'état final correspond au segment retour.

Pour trouver la solution optimale, l'algorithme va utiliser un paradigme de programmation dynamique similaire à celui de la DTW.

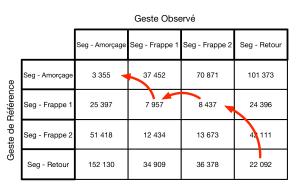
Respectivement i et j, les états de l'automate de référence et de l'automate observé. L'algorithme va calculer pour chaque combinaison d'états {i, j} la distance entre les deux segments associés à chacun des deux états. Il va additionner la plus petite distance entre tous les couples d'états permettant l'accès au couple d'états {i, j}.

Dû à la topologie des automates, le couple d'état sera de la forme $\{i-1, j-x\}$ avec $x \in [0, i]$. Une fois cette opération faite, le programme va déterminer le chemin de poids minimum permettant de parcourir l'intégralité de la séquence observé.

Sur la figure 6.4b est représenté un tableau des coûts (distance entre segment + coût des prédécesseurs). En rouge est représenté, le chemin de coût minimum pour aller de la fin au début, de la séquence d'observation. Ce chemin permet de connaître les segments à associer. Dans l'exemple, l'association est celle-ci : Amorçage du geste de référence (ref) avec Amorçage du geste de référence (obs), Frappe1 (ref) avec Frappe1 (obs), Frappe1 (ref) avec Frappe2 (obs), Retour (ref) avec Retour (obs).



(a) Automates représentant le geste de réference et observé.



(b) Tableau représente le coût de chaque segment (distance entre segment + score prédécesseur). En rouge le chemin de coût minimum, donnant les segments à associer.

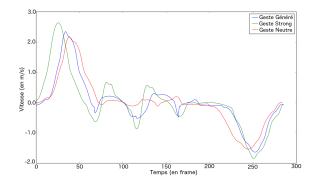
6.1.5 Résultats

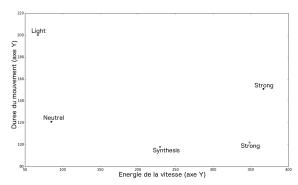
Nous avons donné à l'algorithme deux Knocking : un Neutre et un Strong dans le but d'extraire la fonction de transformation. En appliquant sur un autre geste Neutre, il est possible de le transformer en un geste Strong.

Pour comparer le résultat de l'algorithme, la fonction est appliquée au geste de Knocking Neutre ayant servi à l'extraction de la fonction de transformation. En comparant le Knocking Strong généré et le Knocking Strong d'origine, il est possible d'évaluer les résultats de la méthode.

Les deux figures 6.4a et 6.4b, présentent les similarités entre le geste de synthèse et le geste Strong.

En regardant plus particulièrement la figure 6.4b, il est possible de noter que le geste de





(a) le geste neutre sur lequel a été appliqué la transformation Strong, un geste Strong servant de témoin et le geste synthétique, généré par l'algorithme.

(b) La figure représente les différents Knocking ainsi que le Knocking de synthèse en fonction de la Somme de la vitesse et de la Durée du geste.

FIGURE 6.4

synthèse possède la même durée que le geste Strong. De plus, son energie de la vitesse est plus proche d'un geste Strong que d'un geste Neutre.

Le geste de synthèse est obtenu en appliquant la fonction de transformation sur chaque segment d'un geste source. La précision de ces fonctions canalise en grande partie la vraisemblance de la génération.

Pour améliorer les fonctions de transformation, nous avons fait varier le degré du polynôme pour étudier l'incidence que cela peut avoir sur le résultat.

Avec les mêmes données d'entrée que décrites précédemment, nous avons généré plusieurs gestes en changeant à chaque fois le degré du polynôme. Les degrés de polynôme sont : 1, 2, 3, 5, 7.

Comme le montre la figure 6.5, des polynômes de faibles degrés génèrent des gestes plus proches du geste cible (Strong dans notre exemple) que des polynômes de degrés élevés. Une régression polynomiale essaie d'optimiser une fonction d'un certain degré, dans le but que la fonction soit aussi proche que possible d'une courbe. Une régression de faible degré va produire une fonction ne rendant compte que des grandes variations dans la courbe analysée. A contrario, une régression de degré élevé va produire une fonction rendant compte de toutes les petites variations de la courbe analysée. Dans notre cas, les courbes analysées sont des courbes bruitées (dues à l'acquisition et au calcul). Une régression de degré élevé va donc prendre en compte ce bruit. Ce bruit va avoir un impact sur la pertinence de la fonction polynomiale.

Ceci explique en grande partie les résultats inférieurs des polynômes de degré élevé par rapport aux polynômes de plus faible degré.

6.1.6 Limitations

L'algorithme concaténant les segments de gestes, cela peut provoquer des discontinuités dans le geste synthétisé, comme l'illustre la figure 6.6, où sont représentés le geste Neutre et le geste Strong ayant servi à calculer la fonction de transformation et le geste Strong

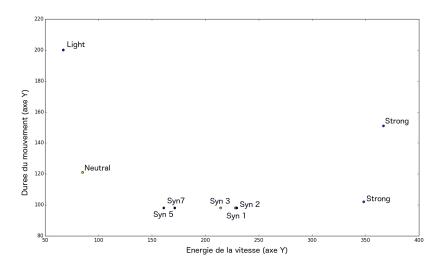


FIGURE 6.5 – Différents Knocking générés par des polynômes de degrés : 1, 2, 3, 5, 7 comparés à des Knocking Strong, Neutre et Light, représentés en fonction de la somme de la vitesse et de la durée du geste.

généré par l'application de cette fonction.

Les bandes jaunes montrent les zones de discontinuité du geste synthèse (courbe bleue). Ce problème est dû au fait que la fonction de transformation n'est pas la même entre les segments voisins, ce qui pose problème pour les échantillons proches de la frontière entre deux segments.

Une valeur en entrée de part et d'autre de cette frontière n'aura pas la même valeur en sortie et c'est ce phénomène qui entraîne des discontinuités dans le signal. Bien que ce problème soit intrinsèque à l'algorithme une solution est possible pour atténuer ce défaut. La solution est de calculer pour chaque valeur d'entrée, la valeur de sortie obtenue par la fonction de transformation du segment courant et la fonction du segment voisin le plus proche (soit le précédent pour la première moitié du segment, soit la seconde moitié), puis de faire la moyenne pondérée entre ces deux valeurs.

Si
$$t > 0.5$$
:

$$y_{sortie} = (1 - t) * F_{seq-1}(x) + (t - 1) * F_{seq}(x)$$
(6.2)

Si t < 0.5:

$$y_{sortie} = (1 - t) * F_{seg-1}(x) + (t - 1) * F_{seg}(x)$$
(6.3)

La taille du segment est normalisée entre 0 et 1. Avec t la position dans le segment, F_{seg} la fonction de transformation du segment, \mathbf{x} la valeur d'entrée et \mathbf{y} la valeur de sortie.

Une seconde limitation est que même si du point de vue des descripteurs, le geste de synthèse est proche du geste expressif, voir figure [numéro du Scatter], les deux courbes n'ont pas la même forme. La courbe générée semble comme lissée par rapport à la courbe Strong. Ce problème est lié à l'utilisation de la régression polynomiale qui va gommer la déformation temporelle.

Ce problème met en lumière le besoin d'avoir un descripteur pour rendre compte de la

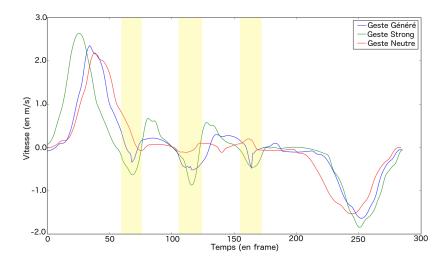


FIGURE 6.6 – Geste Neutre et Strong ayant servie au calcul de la fonction de transformation et le geste Strong généré. En jaune les zones de discontinuité

forme et la courbure du geste.

La dernière limitation de la méthode est la nécessité de segmenter les gestes manuellement. L'algorithme est donc exclusivement utilisable pour des traitements a posteriori.

6.2 Aligorithme basé sur les HMM

La seconde méthode pour transformer un geste neutre en un geste expressif se base sur les Hidden Markov Model (HMM).

Dans sa thèse [15], J.Françoise a développé un Framework développant une approche appelée mapping par démonstration. Le mapping par démonstration permet à l'utilisateur la création d'interactions sonores à partir de démonstrations d'association entre mouvement et son. Pour permettre la construction du mapping entre geste et son, J.Françoise développe un algorithme basé sur les HMM.

Dans cette section, nous explorons comment utiliser le mapping de ce framework pour pouvoir effectuer un mapping entre deux gestes effectués avec différents Efforts.

Dans un premier temps, nous avons évalué la capacité du Framework implémenté par J.Françoise pour reconnaître les Efforts, dans un second temps, nous avons utilisé ce framework pour transformer un geste Neutre en un geste Expressif.

6.2.1 Principe de l'algorithme

Un HMM est un modèle statistique dans lequel le système modélisé est une chaîne de Markov où les paramètres sont inconnus. Un HMM est composé de deux systèmes à états

corrélés. L'un des systèmes sont les états observés, l'autre, sont les états cachés.

Il est possible d'utiliser de deux manières différentes les HMM. La première est d'utiliser le HMM pour classifier des éléments. L'utilisateur donne plusieurs séquences en entrée (input) où chaque séquence est associée à un label au HMM. Cette phase s'appelle la phase d'apprentissage. Puis l'utilisateur donne une nouvelle séquence d'entrée qu'il veut classifier. Le HMM va calculer la ressemblance entre les séquences apprises durant la phase d'apprentissage et donner un score de similarité entre la séquence testée et la séquence apprise.

En entraînant un HMM avec des séquences représentant des gestes de Knocking, effectuées avec différents Efforts, le HMM pourra être en mesure de classifier des gestes en fonction de leurs Efforts. Dans ce cas, les états observés seront les gestes de Knocking et les états cachés seront les noms des Efforts.

La seconde utilisation du HMM permet de générer des séquences similaires aux séquences apprises. L'utilisateur donne des couples de séquences, l'une considérée comme la valeur d'entrée (input) et l'autre considérée comme la valeur de sortie (output). Durant cette phase d'apprentissage, le HMM étudie la correspondance entre les inputs et les outputs. Quand l'utilisateur donne une nouvelle séquence en entrée le HMM va synthétiser une séquence avec la même correspondance que pour les gestes appris.

En entraînant un HMM avec des couples de séquence représentant un geste Neutre et un geste Expressif, le HMM pourra à partir d'un nouveau geste générer le geste Expressif associé.

Pour pouvoir utiliser plusieurs HMM dans des espaces continus, J.Francoise utilise des modèles de mélange Gaussien (GMM) pour calculer ses probabilités d'observation. Les GMM permettent également de pouvoir utiliser tous types de données multidimensionnelles.

6.2.2 Fonctionnement des HMM pour le suivi de geste

Dans sa thèse, J.Francoise utilise les modèles de Markov cachés pour faire de la reconnaissance. Dans cette partie nous présentons en détail comment l'algorithme fonctionne.

L'idée générale de l'algorithme est de faire correspondre un mouvement avec un HMM. Pour chaque état du HMM, l'algorithme va faire correspondre une portion du mouvement. A l'intérieur de chaque état, J.Francoise utilise des modèles de mélange gaussien pour calculer les probabilités d'observation. Sur la figure 6.7, le mapping entre le geste et le modèle de Markov caché associé.

Pour rendre compte de l'avancement temporel du geste, le modèle de Markov caché a une topologie gauche-droite. Cette topologie, force le modèle à avancer temporellement ce qui permet de garder une cohérence lors de la reconnaissance de geste.

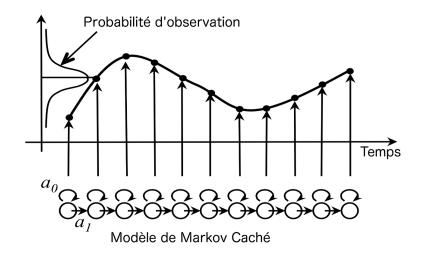


FIGURE 6.7 – Phase d'apprentissage : Mapping entre le geste et le HMM

Modèle de Mixture Gaussien

Un modèle de mixture gaussien (GMM) est un modèle statistique, il permet de modéliser par une somme de gaussienne un phénomène physique. Pour déterminer les paramètres, la variance, la moyenne et l'amplitude de chaque gaussienne, l'algorithme espérance-maximisation (EM) est utilisé.

Considérons un set $\{x_i\}_{i=1:T}$ d'observation correspondant à des enregistrements de mouvements où chaque mouvement est représenté par une séquence de frame $x_i \in \mathbb{R}^{\mathbb{D}}$, avec des données à D-dimensions pour representer le mouvement. Cet ensemble de données va être représenté dans le GMM par un ensemble de K composantes Gaussiennes, définies par une fonction de densité de probabilité :

$$p(x_i|\theta) = \sum_{k=1}^{K} \omega_k \mathcal{N}(x_i; \mu_k; \Sigma_k)$$
(6.4)

Le modèle est descrit par un set de parametres $\theta = \{\omega_{1...K}, \mu_{1...K}, \Sigma_{1...K}\}$ où :

 ω_k est la pondération de la Kème composante, $\omega_k \geq 0, \sum_{k=1}^K \omega_k = 1$

 $\mu_k \in \mathbb{R}^{\mathbb{D}}$ est la moyenne du Kème vecteur

 Σ_k st la covariance de la matrice de la Kème composante.

Pour rappel, la moyenne μ et la covariance Σ dans une fonction de densité de probabilité est définie comme :

$$\mathcal{N}(x;\mu;\Sigma) = \frac{1}{(2\pi)^{D/2} |\Sigma|^{1/2}} exp\left[-\frac{1}{2} (x-\mu)^T \Sigma^{-1} (X-\mu) \right]$$
 (6.5)

Sur la figure 6.8 est représentée la modélisation d'une portion de mouvement par un modèle de mélange Gaussien.

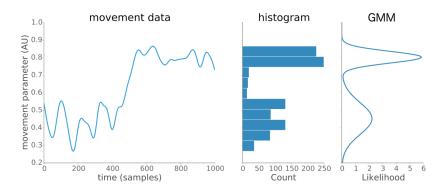


FIGURE 6.8 – Mouvement modélisé par GMM. Le GMM a été entraîné avec les paramètres K=2 $[\sigma]=[1e^{-1},1e^{-3}]$

Phase d'apprentissage

Contrairement aux autres domaines du Machine Learning, dans le domaine de l'interaction gestuelle, le nombre de répétition de gestes est limité. C'est pour cette raison que les Méthodes classiques d'entraînement statistique ont besoin d'être adaptées.

Durant la phase d'apprentissage, l'utilisateur fournit à l'algorithme plusieurs séquences de données (la trajectoire, des descripteurs ...) et un nombre total d'états. Avec ces données, l'algorithme cherche à optimiser l'emplacement des états et le mélange de gaussiennes associé à chaque états.

Pour trouver cet optimal, l'algorithme va utiliser une variante de l'algorithme de Baum-Welch (un algorithme d'Expectation-Maximisation), lui permettant d'entraîner le HMM avec peu de séquences de gestes et des probabilités d'observation continues.

Plusieurs autres variantes de la phase d'apprentissage sont détaillées dans la thèse de J.Francoise [15]

Phase de reconnaissance

Pour la phase de reconnaissance, le HMM utilise la méthode standard de Forward [35]. Considérons, une séquence observée $O_1, O_2, ...O_t$. La méthode de Forward nécessite le calcul de $\alpha_i(t)$ qui correspond à la distribution de probabilité de la séquence d'observation jusqu'au temps t et l'état i.

Elle est calculée inductivement de la manière suivante :

Initialisation

$$\alpha_1(i) = \pi_i b_i(O1) \qquad 1 \le i \le N \tag{6.6}$$

Où π est la distribution initiale de l'état et b la distribution de probabilités d'observation.

Induction

$$\alpha_{t+1}(i) = \left[\sum_{i=1}^{N} \alpha(i) a_{ij}\right] b_i(O_t) \qquad 1 \le t \le T - 1, 1 \le j \le N$$
 (6.7)

Où a_{ij} est la distribution de probabilité de transition de l'état

: À partir de la variable $\alpha_i(t)$, il est possible de calculer deux informations :

La progression temporelle de la séquence, par rapport à la séquence de référence

$$timeprogressionindex(t) = argmax[\alpha_t(i))]$$
 (6.8)

Vraisemblance (likelihood) de la sequence

$$likelihood(t) = \sum_{i=1}^{N} \alpha_t(i)$$
(6.9)

6.2.3 Reconnaissance des Efforts par HMM

Dans l'article de Caramiaux [8], l'auteur explique la difficulté d'étudier l'expressivité avec du Machine Learning du fait que les petites variations d'un geste peuvent être assimilées à la variance. Cela peut provoquer un perte d'information et donc perte de l'expressivité.

Dans un premier temps, une étude a été faite pour vérifier que le framework développé par J.Francoise, était capable de percevoir les variations d'Effort d'un geste.

Objectif principal

L'objectif de cette partie est de vérifier la capacité du framework à reconnaître l'Effort d'un geste à partir de l'évolution de sa vitesse au cours du temps.

Durant la phase d'apprentissage, le HMM est entraîné avec deux gestes de Knocking Strong et deux gestes de Knocking Quick. La vitesse au cours du temps est utilisée comme donnée d'entrée pour le HMM.

Une fois cette phase d'apprentissage terminée, un autre geste de Knocking est donné au HMM pour qu'il le catégorise. Le geste donné au HMM est un geste de Knocking Strong. Le HMM va calculer échantillon par échantillon, le degré de ressemblance entre les deux types d'Efforts (Strong et Quick). Cela permet de connaître l'Effort associé au geste de manière instantanée, avant que le geste ne soit terminé.

La détection d'un état est considérée comme bonne si le geste d'apprentissage ayant la likelihood la plus élevée est celui qui a le même Effort que le geste testé.

Par exemple si le geste test est un geste Strong, la détection pour l'état X est bonne si le geste Strong a un score de likelihood plus important que le geste Quick. En étudiant la proportion de bonnes détections par rapport au nombre de mauvaises, il est possible de connaître l'efficacité de reconnaissance du HMM.

Dans la thèse de J.Francoise, le choix du nombre d'états ainsi que la régularisation statistique des gaussiennes sont choisis empiriquement. Pour déterminer les paramètres optimaux, une estimation de paramètres a été faite. Plus de détails sur ce sujet en annexe A.

D'après cette estimation, le nombre d'états optimal de l'automate dépend de la longueur du geste fourni durant la phase d'apprentissage. Ce nombre d'état est compris entre 35% et 45% du nombre d'échantillons total du geste. La régularisation quant à elle doit être inférieure à 0.01.

Résultats

Un HMM a été entraîné avec deux gestes Knocking Strong et deux gestes de Knocking Quick. Pour la reconnaissance un autre geste de Knocking Strong lui a été fourni. Le nombre d'état du HMM est de 120 (40% du nombre total d'échantillons), le nombre de gaussienne par état est de un et la régularisation est égale à 0,001.

Pour chaque état, le HMM va retourner le degré de similarité (likelihood) entre le geste testé et les gestes de référence. Sur la figure 6.9, la vitesse du geste testé a été tracée ainsi qu'un geste de référence de chaque Effort. En dessous, la log-likelihood du geste test par rapport aux deux types d'Effort a été représentée. La couleur de fond correspond à la couleur de la courbe ayant la log-lokelihood la plus importante. La proportion de bonnes détections est plus importante que les mauvaises. Sur l'exemple de la figure 6.9 la détection est bonne sur 90% des états.

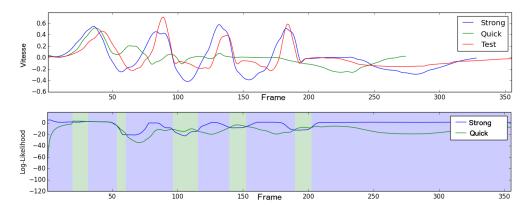


FIGURE 6.9 – Au dessus, représentation de la vitesse au cours du temps de deux gestes de références, Strong, Quick et le geste test Strong. En dessous, la log-likelihood entre le geste test et les deux gestes de réferences retournée par le HMM. La couleur de fond correspond à la couleur de la courbe ayant la likelihood la plus importante.

Discussion

Les résultats de la section précédente montrent que les HMM comme développés dans la thèse de J.Francoise sont capable de classifier un geste entre deux efforts en temps-réel. Cela valide l'hypothèse que sous certaines conditions il est possible de capter l'expressivité du mouvement.

D'après des premiers tests, la classification d'un geste entre les huit Efforts semble également possible. Malheureusement le manque d'exemple d' Efforts ne nous permet pas d'évaluer les performances du HMM.

6.2.4 Transformation du neutre vers l'expressif par HMM

Dans la section précédente, nous avons vérifié que le framework développé par J.Francoise était capable de percevoir les Efforts dans le geste.

Dans cette section, nous allons étudier comment l'utiliser pour transformer un geste Neutre en un geste Expressif. Plus précisément, le HMM va être entraîné avec deux gestes, un neutre et un expressif, dans le but d'être capable de générer un geste Expressif à partir d'un autre exemple de geste neutre.

Principe algorithmique

Le fonctionnement des modèles de Markov cachés pour de la génération est assez proche du fonctionnement de la reconnaissance.

La différence réside dans le fait que durant la phase d'apprentissage, le HMM n'apprend pas la correspondance entre un mouvement et un label "Effort" mais apprend la corrélation entre un mouvement et un autre mouvement.

Le HMM prend en entrée plusieurs couples séquences, une considérée comme l'entrée et l'autre la sortie. Pour chaque état le HMM calcule un modèle de mixture gaussiennes représentant une portion temporelle des séquences d'entrée, puis le HMM fait de même pour les séquences de sortie. Pour chaque état, le HMM calcule la correspondance entre le GMM d'entrée et le GMM de sortie. Voir figure 6.10.

Pour la phase de génération, le HMM analyse un geste d'entrée échantillon par échantillon. Pour chaque l'échantillon, le HMM va comparer sa valeur avec les GMM d'entrée pour connaître l'état le plus probable. Avec cet état le HMM peut générer la valeur de sortie en utilisant les GMM de sortie

Application de l'algorithme

Durant la phase d'apprentissage, le HMM a été entraîné avec quatre gestes, deux couples de geste Knocking Neutre-Strong. Pour représenter les gestes dans le HMM, la vitesse au cours du temps est utilisée. Pour la phase de génération, le HMM a synthétisé un geste Expressif à partir d'un geste d'entrée Neutre.

Dans un premier temps, le même geste Neutre est utilisé pour la phase d'apprentissage et pour la phase de génération, ce qui permet de comparer le geste Expressif généré et le geste Expressif d'origine. Dans un second temps un autre Knocking Neutre est utilisé pour la phase de génération.

Comme pour la reconnaissance, une estimation de paramètres a été faite pour déterminer le nombre optimal d'états ainsi que la régularisation statistique des gaussiennes. Le

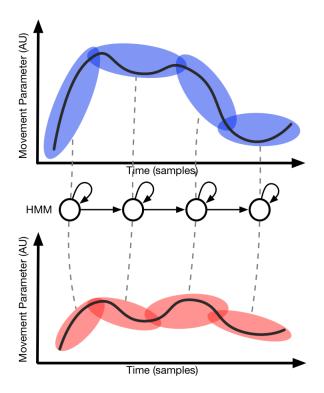


FIGURE 6.10 – Mapping fait par le HMM entre le geste Neutre (en bas) et le geste expressif (expressif). Les GMM sont représentés par les ellipses rouges et bleues.

nombre d'états optimal du HMM dépend de la longueur du geste fourni durant la phase d'apprentissage. Le nombre d'état est compris entre 70-80% du nombre d'échantillon total du geste d'entrée et la régularisation doit être inférieure à 0,01.

6.2.5 Résultats

Pour de la génération, un HMM a été entraîné avec un couple de gestes : un Neutre et un Strong. Pour la phase de génération, le même geste Neutre va être donné en entrée pour le HMM dans le but de pouvoir comparer le geste Strong généré avec le geste Strong d'origine (celui ayant servi à l'apprentissage).

Le nombre d'états du HMM est 250 soit 74% de la longueur des séquences d'apprentissage et la régularisation est de 0,001.

Sur la figure 6.11, est représentée la vitesse des deux gestes Neutre et Strong, ayant servis pour l'apprentissage ainsi que le geste Strong généré par le HMM. Le geste généré par le HMM est très proche du geste Strong d'origine.

Un second test a été réalisé en utilisant le même HMM. Les deux gestes pour la phase d'apprentissage sont les mêmes que précédemment mais en utilisant un geste Neutre diffèrent de celui d'apprentissage.

Le geste généré par le HMM est visible sur la figure 6.12. Le geste généré possède des caractéristiques communes avec le geste Strong d'origine. Ces résultats montrent que les

HMM développés par J.Francoise sont utilisables pour passer d'un geste Neutre à un geste Expressif.

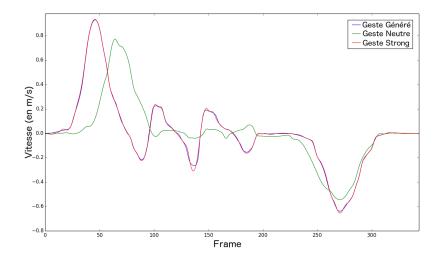


FIGURE 6.11 – Comparatif entre les deux gestes utilisés pour l'apprentissage du HMM et le geste Strong généré à partir du geste Neutre représenté.

6.2.6 Discussion

Les transformations faites entre un geste Neutre et geste Expressif sont très concluantes. L'utilisation des HMM dispense de segmenter le geste manuellement de plus la méthode par HMM permet de générer le geste en temps réel ce qui rend cette méthode plus adaptée pour l'interaction.

Le premier défaut de la méthode basée sur les HMM est que les paramètres optimaux pour la génération sont liés à la taille des séquences d'apprentissage, cela oblige d'ajuster les paramètres pour chaque cas. De plus, la manière dont est fait l'algorithme d'estimation de paramètres ne permet pas d'estimer les paramètres optimaux si la phase d'apprentissage est faite avec plusieurs séquences d'entrée. Dans l'algorithme implémenté, le critère d'optimisation est la distance échantillon à échantillon entre la séquence générée par le HMM et la séquence de sortie de l'apprentissage. Pour un HMM entraîné avec plusieurs séquences d'apprentissage, il est donc nécessaire de trouver une autre manière de calculer ce critère.

Dans la méthode par DTW, pour passer d'un geste Neutre à un geste Expressif, la fonction de transformation est explicite. Il est possible de comparer la fonction de transformation calculée sur des couples de gestes (Neutre et Strong, Neutre et Sustained). Dans le cas de la méthode par HMM, la fonction de transformation est contenue dans la structure même du HMM. L'information de la transformation est contenue dans la position et les paramètres des GMM.

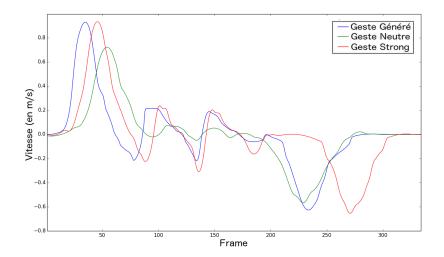


FIGURE 6.12 – Comparatif entre le geste Strong d'origine et le geste Strong généré. Le geste Strong généré a été synthétisé à partir du geste Neutre représenté et d'un HMM entraîné avec un geste Neutre (représenté sur la figure 6.11) et le Strong d'origine.

Le dernier défaut de la méthode par HMM est lié au fait qu'un HMM ne peut générer que ce pourquoi il a été entraîné. Durant la phase d'apprentissage le HMM a calculé les GMM pour représenter le geste d'entrée et la correspondance avec le geste de sortie. Le HMM ne peut donc produire des résultats cohérents uniquement si le geste d'entrée de la phase de génération est proche du geste d'entrée de la phase d'apprentissage. Si le geste d'entrée utilisé pour la génération est trop éloigné du geste utilisé par l'apprentissage, les valeurs des échantillons ne correspondront à aucun GMM et donc le HMM générera une séquence de sortie aberrante. Déterminer si un geste utilisé pour la génération est assez proche du geste d'apprentissage pour permettre au HMM de générer un geste cohérent n'est possible qu'empiriquement.

De plus, déterminer si un geste est cohérent est une chose complexe car le fait que le geste généré soit proche du geste de référence ne garantit pas qu'il possède exactement la même Expressivité. Même si l'utilisation de la métrique permet de vérifier si les deux gestes partagent des caractéristiques communes, une nouvelle méthode de validation à besoin d'être étudiée. Voir section perceptive 8.

6.2.7 Validation des gestes générés par HMM

Nous avons voulu évaluer la ressemblance entre les gestes synthétisés par la méthode des HMMs et les gestes réels issus du Dataset enregistré pendant le stage.

L'idée est de faire classer au KNN les différents gestes de synthèse selon leurs efforts et d'étudier si l'effort attribué par le Knn est le même que celui du geste réel utilisé pendant l'entraînement du HMM qui a servi à produire le geste de synthèse.

Principe de la méthode

Pour étudier la ressemblance entres les gestes de synthèse et les gestes réels. La méthode consiste à entraı̂ner un Knn avec les gestes réels. Chaque geste est constitué d'un couple : données, label. Les données correspondant aux différentes valeurs des descripteurs associés au geste.

Une fois le Knn entraîné, il va classifier les gestes de synthèse. À la fin de la classification, la proportion de gestes bien classés par rapport au nombre total de gestes de synthèse va être comptabilisée pour établir un score Accuracy.

Un geste est considéré comme bien classé si la classe attribuée au geste par le Knn est la même que celle du geste ayant servi de geste d'entraînement pour le HMM ayant généré le geste de synthèse. Par un exemple, un geste généré à partir d'un HMM entraîné avec un geste Neutre en entrée et un geste Strong en sortie, sera considéré bien classé si la classe Strong lui est attribuée par le Knn.

Génération des gestes de synthèse

Pour générer les gestes de synthèse, nous avons procédé de la manière suivante : Pour chaque Effort E, un HMM est entraîné avec tous les gestes Neutre sauf un comme donnée d'entrée et tous les gestes effectués avec l'Effort E. Le geste Neutre non utilisé pour la phase d'apprentissage est tiré aléatoirement. Puis nous avons utilisé le geste Neutre, n'ayant pas été utilisé pour l'entraînement, pour servir comme donnée d'entrée pour la phase de génération.

L'opération est itérée 50 fois pour chaque Effort afin de constituer un ensemble conséquent de gestes. Soit un total de 350 gestes regroupés en 7 Efforts.

Classification par Knn

Pour la phase d'entraînement du Knn, les gestes utilisés sont ceux du Dataset et labellisés par leurs Efforts respectifs. Les descripteurs utilisés sont ceux déterminés dans la section 5.1 par notre méthode d'évaluation de paramètres.

Efforts	Accuracy
Strong	1,0
Light	0,29
Quick	1,0
Sustaiened	0,90
Free	0,53
Bound	0,87
Direct	0,74

FIGURE 6.13 – Accuracy moyenne obtenue avec avec trois configurations différentes : avec deux descripteurs Énergie de la vitesse et Durée, avec tous les descripteurs et le sous-ensemble optimal déterminé par notre algorithme.

Le tableau 6.13 dresse un récapitulatif des scores d'accuracy pour les différents Efforts.

La reconnaissance moyenne est supérieure à 75% ce qui est un résultat encourageant. Il faut également noter que pour 2 des Efforts le taux est de 100% ce qui veut dire que nos descripteurs sont adéquats pour reconnaître ces Efforts. La différence de résultats laisse penser que d'autres descripteurs devraient être ajoutés pour permettre d'augmenter la reconnaissance de certains Efforts notamment Light et Free.

Chapitre 7

Nouvelle Expérience et Création d'un nouvel ensemble de données

Dans cette section, nous présentons le nouvel ensemble de données enregistrées durant ce stage.

Cet ensemble est constitué de gestes de Knocking faits avec un Effort et de gestes de Knocking effectués avec plusieurs Efforts en même temps.

Ces données ont été enregistrées dans deux buts : Le premier est de combiner ces nouveaux enregistrements avec ceux de la base MoDa pour avoir plus de répétitions de chaque Efforts, pour permettre une analyse statistique des résultats. De plus, cela devrait permettre de faire des premiers tests de classification automatique des Efforts (K-means, KNN...). Le deuxième but va permettre d'analyser des gestes effectués avec des Efforts multiples.

Pour pouvoir combiner les données venant des enregistrements et de la base MoDa, nous avons enregistré le même type de données. Pour cela nous avons utilisé un Kinect2 qui a permis l'enregistrement des mouvements du squelette du participant au cours du temps.

7.1 Contenu du nouvel ensemble de données

La base est constituée de gestes de Knocking effectués avec chaque Effort. Le nombre d'États et d'Impulsions étant très important, seule une combinaison a été enregistrée pour chaque État et chaque Impulsion. Pour rappel, un État au sens de Laban correspond à l'association de deux Efforts, et une Impulsion est elle, l'association de trois Efforts.

7.2 Protocole expérimental

Cet ensemble de données a été enregistré le 20 juillet 2016 à l'Ircam. Nous avons eu la participation d'une élève du cursus Notation De Mouvement Laban, au CNSMD (Conservatoire Nationnal Supérieur de Musique et de Danse) de Paris. L'acquisition du mouvement du danseur a été faite à l'aide d'une Kinect2 puis traité par un patch Max.

Déroulement de l'expérience

Le participant est face au capteur Kinect2. Pour chaque Effort, il va effectuer une T-pose puis une répétition de 10 gestes de Knocking et pour finir va faire de nouveau un T-pose. Comme décrit dans la section 4.1, un geste de Knocking est composé d'une phase d'amorçage où le participant va passer de la position de repos (les bras le long du corps) à la position de frappe (le poignet au niveau de l'épaule), d'une phase Frappe où il va effectuer une série de trois frappes puis une phase de Retour où le participant va revenir à sa position de repos.

Chapitre 8

Perspective et travaux futurs

Dans cette section nous exposons les perspectives et les travaux futures envisageables pour le projet.

8.1 Résumé du travail effectué

Le travail effectué durant ce stage s'est orienté en trois grand axes. Le premier est une recherche théorique sur les différentes manières de capter et d'analyser l'expressivité dans le mouvement. Nous avons établi les problèmes que soulèvent l'analyse de l'expressivité en dressent l'état de l'art des recherches effectué dans le domaine. En s'appuyant sur ces recherches bibliographiques, nous avons choisi d'étudier en profondeur la théorie de Laban. Cette théorie semblant être actuellement la théorie la plus complète pour analyser le mouvement.

Le second grand axe est l'analyse des mouvements en utilisant la théorie de Laban, et l'élaboration d'un ensemble de descripteurs permettant de comparer les différents Efforts. La pertinence de ces descripteurs a été vérifiée en utilisant des algorithmes de K-nearest neighbors. De plus, cette évaluation a montré que la reconnaissance, en temps différé, des efforts d'un geste est possible. D'autre part, nous avons étudié la possibilité de pouvoir reconnaissance, en temps réel, des efforts grâce au framework de J.François.

Le troisième grand axe est l'élaboration de deux systèmes pour transformer un geste neutre en un geste expressif. Le premier système permet d'extraire un fonction explicite de transformation entre le geste neutre et le geste expressif. En utilisant cette fonction, il est possible d'appliquer une transformation expressive à n'importe quelle geste neutre. Le défaut majeur de ce système est le besoin de devoir segmenter le geste manuellement, ce qui limite son utilisation à grande échelle.

Le second système utilise le framework de J.Françoise, l'utilisation de ce framework, permet de créer un mapping entre des gestes neutres et des gestes expressifs, dans le but de pouvoir synthétiser des gestes expressifs. Avec ce système, il est possible de générer facilement un très grands nombres de gestes. Pour évaluer la vraisemblance des gestes générés, nous avons utilisé des algorithmes de classification pour étudier si l'algorithme faisait la distinction entre gestes réels et les gestes de synthèses.

8.2 Étude de la composition des Efforts

Durant ce projet nous n'avons étudié uniquement la transformation entre un geste Neutre et un geste effectué avec un seul Effort. Il serait maintenant intéressant d'élargir l'étude à des gestes effectués avec plusieurs Efforts en même temps.

Cette étude est maintenant réalisée grâce à la nouvelle base de données enregistrée durant ce projet.

L'intérêt d'étudier ces gestes serait de valider l'hypothèse considérant les Efforts de Laban comme une base, au sens mathématique. Grâce à un telle base, il serait possible de décomposer n'importe quelle l'expressivité.

De plus, il serait également intéressant d'étudier les fonctions de transformation des gestes effectués avec plusieurs Efforts, dans le but de vérifier si la fonction de transformation d'un geste effectué avec plusieurs Efforts correspond à la somme des fonctions de transformation de gestes effectués avec un seul Effort.

8.3 Validation par synthèse de mouvement

Au travers de ce projet, la plus grosse difficulté a été de trouver une méthode pour valider les résultats obtenus par les différentes méthodes.

Une première approximation peut être faite par l'utilisation d'une métrique pour comparer les gestes générés avec des gestes de références. Cette métrique n'est pas suffisante car les descripteurs ne rendent pas forcement compte de tous les aspects du geste, et ne peut pas garantir que l'expressivité soit présentée dans le geste de synthèse.

Après de nombreuses discussions avec des chercheurs en interaction ainsi qu'avec des danseurs utilisant Laban, il nous est apparu que la seule validation possible serait de générer un avatar et de l'animer à partir des courbes générées par nos algorithmes puis de faire analyser les mouvements de l'avatar par un expert de Laban (un Laban Movement Analysis) pour valider nos résultats.

Annexe A

Estimation de paramètres

Pour déterminer les paramètres optimaux pour maximiser la reconnaissance des Efforts par HMM, une estimation de paramètre a été faite.

Les paramètres à déterminer sont le nombre d'états et la valeur de la régularisation des GMM. Chaque paramètre peut prendre une valeur comprise entre deux bornes (inférieure et supérieure). La méthode d'estimation consiste à entraîner un HMM avec toutes les combinaisons de paramètres possible, puis déterminer la configuration qui permet d'avoir la reconnaissance maximale.

Plus précisément, un HMM est entraîné avec deux gestes effectués avec deux Effort différents, puis un geste test est donné au HMM pour être classifié. Ce geste de Knocking est du même Effort que l'un des deux gestes d'apprentissage. Le nombre de bonnes détections sur l'ensemble des états du HMM est calculé pour la configuration.

Pour rappel, la détection d'un état est considérée comme bonne si le geste d'apprentissage ayant la likelihood la plus élevée est celui qui a le même Effort que le geste testé.

Une fois toutes les valeurs de paramètres balayées, la configuration ayant eu la meilleure détection donnera la valeur optimale des paramètres.

Pour ne pas biaiser l'estimation, nous avons itéré le test avec tous les couples d'Efforts opposés (Strong/Light, Free/Bound ...) ainsi que des couples d'Efforts tirés aléatoirement (Susteined/Light, Direct / Quick ...).

Sur la figure A.1 est représenté le nombre de bonnes détections reconnues par le HMM selon le nombre d'états et la régularisation. Le HMM a été entraîné avec un Knocking Strong et un Knocking Light, le geste test est lui un Knocking Strong.

Le nombre d'échantillons des trois gestes sont 336 et 328 pour les gestes Strong et Light utilisés pour l'apprentissage et 342 pour le geste Strong de test. Sur cette figure, deux pics offrent une reconnaissance optimale : un autour de 120 et l'autre après 320 états.

Le second pic correspond au moment où le nombre d'états correspond au nombre d'échantillons des gestes d'apprentissage. À l'intérieur du HMM, chaque GMM représente un échantillon du geste ce qui permet d'avoir une bonne reconnaissance.

Le premier pic quant à lui est beaucoup plus dur à expliquer. Selon les gestes, la position du pic n'est pas toujours centrée sur la même valeur, de plus le pic peut être plus ou moins large.

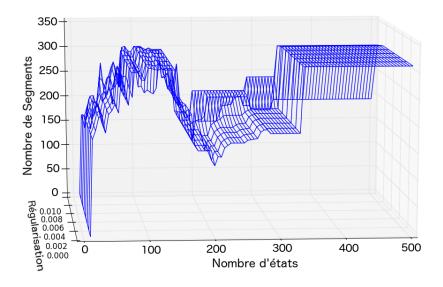


FIGURE A.1 – Nombre de segment reconnues selon le nombre d'états du HMM et la régularisations des GMM. Le HMM est entraîné avec deux gestes Strong et Light, et le geste test est Strong.

De manière empirique, nous avons réussi à déterminer un lien entre la taille de la séquence d'apprentissage et la position du pic. La position du pic est toujours comprise entre 30 et 40 % de la taille de la séquence.

Les deux pics ayant la même proportion de bonne détection, nous avons choisi arbitrairement de prendre les paramètres correspondant au premier comme paramètres optimaux. Il y a plusieurs raisons à ce choix. La première est que dans les HMM utilisés, la chaîne d'états ne permet que deux types de transition : self-transition et left-right. Avec cette topologie durant l'analyse du geste, soit on reste dans le même état, on reste au même endroit dans le geste référence, soit on passe à l'état suivant et donc on avance dans le geste de référence. Il n'existe pas de transition pour sauter des états, il est donc impossible pour le geste test d'aller plus vite que le geste utilisé pour l'apprentissage.

Un second argument en faveur d'un HMM avec peu d'états est la complexité algorithmique des HMM. L'algorithme de Viterbi permet au HMM de reconnaître une séquence avec un complexité de $O(N^2T)$ avec N le nombre d'états et T la taille de la séquence observée. Comme le souligne Fréderic Bevilacqua et al [2], il est donc préférable, dans le cadre d'un dispositif interactif, d'avoir un HMM avec peu d'états.

Pour la régularisation des GMM, nous avons balayé l'intervalle [$1*10^{-5}$, 5], un palier de reconnaissance optimal apparaît en dessous de la valeur 0,01. Au de là, le nombre de bonnes détections chute rapidement.

Bibliographie

- [1] Sarah Fdili Alaoui, Baptiste Caramiaux, Marcos Serrano, and Frédéric Bevilacqua. Movement qualities as interaction modality. In *Proceedings of the Designing Interactive Systems Conference*, pages 761–769. ACM, 2012.
- [2] Frédéric Bevilacqua, Bruno Zamborlin, Anthony Sypniewski, Norbert Schnell, Fabrice Guédy, and Nicolas Rasamimanana. Continuous realtime gesture following and recognition. In *International Gesture Workshop*, pages 73–84. Springer, 2009.
- [3] A. Camurri, M. Ricchetti, and R. Trocca. Eyesweb-toward gesture and affect recognition in dance/music interactive systems. In *Multimedia Computing and Systems*, 1999. *IEEE International Conference on*, volume 1, pages 643–648 vol.1, Jul 1999.
- [4] Antonio Camurri and Marc Leman. Gestalt-based composition and performance in multimodal environments. In *Music*, *Gestalt*, and *Computing*, pages 495–508. Springer Berlin Heidelberg, 1997.
- [5] Antonio Camurri, Barbara Mazzarino, and Gualtiero Volpe. Analysis of expressive gesture: The eyesweb expressive gesture processing library. In *International Gesture Workshop*, pages 460–467. Springer, 2003.
- [6] Antonio Camurri, Barbara Mazzarino, Gualtiero Volpe, Pietro Morasso, Federica Priano, and Cristina Re. Application of multimedia techniques in the physical rehabilitation of parkinson's patients. *The Journal of Visualization and Computer Animation*, 14(5):269–278, 2003.
- [7] Baptiste Caramiaux. Motion modeling for expressive interaction: A design proposal using bayesian adaptive systems. In *Proceedings of the 2014 International Workshop on Movement and Computing*, page 76. ACM, 2014.
- [8] Baptiste Caramiaux, Frederic Bevilacqua, and Atau Tanaka. Beyond recognition: using gesture variation for continuous interaction. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*, pages 2109–2118. ACM, 2013.
- [9] Baptiste Caramiaux and Atau Tanaka. Machine learning of musical gestures. In *NIME*, pages 513–518, 2013.
- [10] Diane Chi, Monica Costa, Liwei Zhao, and Norman Badler. The emote model for effort and shape. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 173–182. ACM Press/Addison-Wesley Publishing Co., 2000.
- [11] Sofia Dahl. Striking movements: A survey of motion analysis of percussionists. *Acoustical science and technology*, 32(5):168–173, 2011.
- [12] Keshi Dai, Harriet J Fell, and Joel MacAuslan. Recognizing emotion in speech using neural networks. *Telehealth and Assistive Technologies*, 31:38, 2008.

- [13] Frank Dellaert, Thomas Polzin, and Alex Waibel. Recognizing emotion in speech. In Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on, volume 3, pages 1970–1973. IEEE, 1996.
- [14] Nesrine Fourati and Catherine Pelachaud. Emilya: Emotional body expression in daily actions database. In *LREC*, pages 3486–3493, 2014.
- [15] Jules Françoise. Motion-Sound Mapping by Demonstration. PhD thesis, UPMC, 2015.
- [16] Jules Françoise, Frédéric Bevilacqua, and Thecla Schiphorst. GaussBox: Prototyping Movement Interaction with Interactive Visualizations of Machine Learning. In Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems, CHI EA '16, pages 3667–3670, San Jose, CA, 2016. ACM.
- [17] Jules Françoise, Baptiste Caramiaux, and Frédéric Bevilacqua. Realtime Segmentation and Recognition of Gestures using Hierarchical Markov Models. PhD thesis, Master ATIAM - Ircam, UPMC, 2011.
- [18] Jules Françoise, Baptiste Caramiaux, and Frédéric Bevilacqua. A hierarchical approach for the design of gesture-to-sound mappings. In 9th Sound and Music Computing Conference, pages 233–240, 2012.
- [19] Jules Françoise, Sarah Fdili Alaoui, Thecla Schiphorst, and Frédéric Bevilacqua. Vocalizing dance movement for interactive sonification of laban effort factors. In *Proceedings of the 2014 conference on Designing interactive systems*, pages 1079–1082. ACM, 2014.
- [20] Jules Françoise, Agnès Roby-Brami, Natasha Riboud, and Frédéric Bevilacqua. Movement sequence analysis using hidden markov models: a case study in tai chi performance. In *Proceedings of the 2nd International Workshop on Movement and Computing*, pages 29–36. ACM, 2015.
- [21] Jules Françoise, Norbert Schnell, and Frédéric Bevilacqua. A multimodal probabilistic model for gesture—based control of sound synthesis. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 705–708. ACM, 2013.
- [22] Emilien Ghomi. Designing expressive interaction techniques for novices inspired by expert activities: the case of musical practice. PhD thesis, Université Paris Sud-Paris XI, 2012.
- [23] Consuelo Granata, Aurelien Ibanez, and Philippe Bidaud. Human activity-understanding: a multilayer approach combining body movements and contextual descriptors analysis. *International Journal of Advanced Robotic Systems*, 12, 2015.
- [24] Kozaburo Hachimura, Katsumi Takashina, and Mitsu Yoshimura. Analysis and evaluation of dancing movement based on lma. In *ROMAN 2005. IEEE International Workshop on Robot and Human Interactive Communication*, 2005., pages 294–299. IEEE, 2005.
- [25] Young-Seon Jeong, Myong K Jeong, and Olufemi A Omitaomu. Weighted dynamic time warping for time series classification. *Pattern Recognition*, 44(9):2231–2240, 2011.
- [26] Kamrad Khoshhal, Hadi Aliakbarpour, Joao Quintas, Paulo Drews, and Jorge Dias. Probabilistic lma-based classification of human behaviour understanding using power

- spectrum technique. In *Information Fusion (FUSION)*, 2010 13th Conference on, pages 1–7. IEEE, 2010.
- [27] Rudolf Laban and Lisa Ullmann. The mastery of movement. 1971.
- [28] Longin Jan Latecki, Qiang Wang, Suzan Koknar-Tezel, and Vasileios Megalooikonomou. Optimal subsequence bijection. In *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, pages 565–570. IEEE, 2007.
- [29] E Moons. Laban movement analysis using a bayesian model and perspective projections. *Machine Learning*, 4(2):227–243.
- [30] Meinard Müller and Tido Röder. Motion templates for automatic classification and retrieval of motion capture data. In *Proceedings of the 2006 ACM SIGGRA-PH/Eurographics symposium on Computer animation*, pages 137–146. Eurographics Association, 2006.
- [31] Jean-Luc Nespoulous, Paul Perron, and André Roch Lecours. The biological foundations of gesture: Motor and semiotic aspects. Psychology Press, 2014.
- [32] Nuria Oliver and Eric Horvitz. A comparison of hmms and dynamic bayesian networks for recognizing office activities. In *International conference on user modeling*, pages 199–209. Springer, 2005.
- [33] Frank E Pollick. The features people use to recognize human movement style. In *International Gesture Workshop*, pages 10–19. Springer, 2003.
- [34] Frank E Pollick, Vaia Lestou, Jungwon Ryu, and Sung-Bae Cho. Estimating the efficiency of recognizing gender and affect from biological motion. *Vision research*, 42(20):2345–2355, 2002.
- [35] Lawrence R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.
- [36] Bernstein Ran, Shafir Tal, Tsachor Rachelle, Studd Karen, and Schuster Assaf. Multitask learning for laban movement analysis. In *Proceedings of the 2nd International Workshop on Movement and Computing*, pages 37–44. ACM, 2015.
- [37] Joerg Rett, Jorge Dias, and Juan-Manuel Ahuactzin. Laban Movement Analysis using a Bayesian model and perspective projections. INTECH Open Access Publisher, 2008.
- [38] Luís Santos, Kamrad Khoshhal, and Jorge Dias. Trajectory-based human action segmentation. *Pattern Recognition*, 48(2):568–579, 2015.
- [39] Björn Schuller, Gerhard Rigoll, and Manfred Lang. Hidden markov model-based speech emotion recognition. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, volume 2, pages II–1. IEEE, 2003.
- [40] Diego Silang Maranan, Sarah Fdili Alaoui, Thecla Schiphorst, Philippe Pasquier, Pattarawut Subyen, and Lyn Bartram. Designing for movement: evaluating computational models using lma effort qualities. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*, pages 991–1000. ACM, 2014.
- [41] Dilip Swaminathan, Harvey Thornburg, Jessica Mumford, Stjepan Rajko, Jodi James, Todd Ingalls, Ellen Campana, Gang Qian, Pavithra Sampath, and Bo Peng. A dynamic bayesian approach to computational laban shape quality analysis. *Advances in Human-Computer Interaction*, 2009:2, 2009.

- [42] Mihai Zanfir, Marius Leordeanu, and Cristian Sminchisescu. The moving pose: An efficient 3d kinematics descriptor for low-latency action recognition and detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2752–2759, 2013.
- [43] Victor Zappi and Andrew McPherson. Dimensionality and appropriation in digital musical instrument design. In *NIME*, pages 455–460, 2014.
- [44] Yohan Zeitoun. Danse et nouvelles technologies-inscrire le corps dans les dispositifs interactifs.