Détection précoce d'actions squelettiques 3D dans un flot non segmenté à base de modèles curvilignes

S.Y. Boulahia[1] É. Anquetil[1] F. Multon[2] R. Kulpa[2]

[1] Univ Rennes, INSA de Rennes, IRISA. France
[2] Univ Rennes, Inria, M2S. France

{said-yacine.boulahia, eric.anquetil, franck.multon, richard.kulpa}@irisa.fr

Résumé

La détection précoce d'actions vise à déterminer au plus tôt la classe d'une action, si elle a lieu, dans un flot non segmenté et en utilisant le moins d'observations possibles. Cette tâche est d'autant plus complexe qu'elle doit considérer à la fois des contraintes de détection, de reconnaissance et celles de précocité. De ce fait, et en dépit de son intérêt pratique certain, cette problématique est rarement abordée dans la littérature. Dans ce papier, nous proposons une approche combinant trois modèles basés sur le concept de segmentation curviligne. Ces modèles permettent de traiter le flot d'entrée à court, moyen et long terme de façon à réduire à la fois le risque d'erreur et la latence. De plus, le fait que ces modèles soient basés sur la notion de segmentation curviligne permet de s'affranchir de difficultés inhérentes à la détection, notamment la variabilité des vitesses de performance. En outre, l'approche proposée est adaptée au cas de la reconnaissance d'actions pré-segmentées afin de permettre une comparaison aux meilleures approches de la littérature, notamment celles à base d'apprentissage profond. Les résultats que nous présentons dans le cadre de la détection précoce et celui de la reconnaissance sont prometteurs vis-à-vis de ce sujet de recherche encore en émergence.

Mots Clef

Détection précoce, représentation basée squelette, modèles curvilignes, flot non segmenté, reconnaissance d'actions, gestes 3D.

Abstract

Early action detection aims to determine as soon as possible the class of an action, if it takes place, in an unsegmented flow and using as few observations as possible. This task is all the more complex since it must consider several constraints related to detection, recognition and precocity. As a result, and despite its obvious practical interest, this problem is rarely addressed in the literature. In this paper, we propose an approach combining three models based on the concept of curvilinear segmentation. These models make it possible to process the input stream in the short, medium and long term so as to reduce both the risk of misdetecting and latency. Moreover, the fact that these models are based on the notion of curvilinear segmentation makes it possible to overcome the inherent difficulties of detection, in particular the variability of performance speeds. In addition, the proposed approach is adapted to the case of pre-segmented action recognition in order to compare it to the best approaches of the literature, in particular those based on deep learning. The results we present in the field of early detection and recognition are promising for this topic of research that is still emerging.

Keywords

Early detection, skeleton-based representation, curvilinear models, unsegmented flow, action recognition, 3D gestures.

1 Introduction

La capacité de détecter et d'identifier au plus tôt l'occurrence d'une action (geste 3D) dans un flot non segmenté a des applications dans une multitude de domaines, allant de la robotique, les jeux vidéos à la surveillance et la rééducation. Par exemple, considérons un scénario où un robot compagnon partage l'espace de vie avec les humains. Une exigence clé pour le robot serait sa capacité à interagir physiquement avec les humains. Le robot doit être capable d'interpréter voire d'anticiper les actions de l'humain pour réagir en temps opportun ; sinon, l'interaction sera lente et peu réaliste.

La détection précoce d'actions 3D est une problématique différente mais connexe à d'autres problématiques mieux connues en apprentissage automatique et reconnaissance de formes. On retrouve notamment la **reconnaissance** d'actions et la **détection** enligne d'actions (**OAD** pour *Online Action Detection*). La reconnaissance se résume à l'étiquetage de séquences d'actions complètement achevées et pré-segmentées et la détection (OAD) vise à identifier des actions ayant lieu dans un flot non segmenté et ce en temps-réel. En comparant ces deux tâches, il est clair que la tâche de la OAD est plus complexe que la simple reconnaissance étant donné que les instants de début et de fin de l'action,

si elle a lieu, ne sont pas connus au préalable.

Sur la base de ces deux définitions, il est possible de mieux comprendre en quoi consiste la problématique considérée dans ce papier, à savoir la **détection précoce**. Il s'agit en fait de reproduire la tâche de la OAD mais en faisant cela le plus tôt possible : idéalement au début de l'action et au plus tard avant la fin cette action. Ainsi, l'objectif est d'anticiper la reconnaissance d'une action avant que celle-ci ne soit terminée et cela dans un flot continu. L'approche que nous proposons utilise des données squelettiques du corps complet. Elle est basée sur le déploiement de trois modèles traitant le flot d'entrée à court, moyen et long terme. Ceci permettrait notamment d'émettre une décision sur l'action en cours à partir du moment où celle-ci est suffisamment discernable des autres classes d'actions. Ce discernement peut être possible dés le traitement des premières frames, auquel cas le modèle à court terme permet d'identifier cette action. Si au contraire plusieurs classes d'actions partagent un même début, c'est alors le modèle à moyen voire à long terme qui devrait se prononcer.

L'approche proposée est d'autant plus intéressante qu'elle se base sur le concept de segmentation curviligne. A l'opposée des approches de détection précédentes [1, 2, 3], la segmentation curviligne consiste à indexer la taille du segment d'action à considérer sur le flot spatial et non pas temporel. C'est notamment au moyen de ce concept de segmentation que nous permettons de mieux adresser les variabilités des vitesses d'exécution lors de la performance d'une action. En plus de l'évaluation de ce concept dans le cadre de la détection précoce, nous proposons une approche alternative de reconnaissance d'actions présegmentées afin de permettre notamment une comparaison aux approches à base d'apprentissage profond.

Le reste de cet article est organisé comme suit. La Section 2 est consacrée à la présentation des travaux de littérature relative à la modélisation et la détection précoce d'actions à base de données squelettiques. Nous présentons ensuite dans la Section 3 notre approche de détection précoce, où est notamment introduit le concept de segmentation curviligne. Dans cette Section, nous expliquons comment l'approche proposée est adaptée pour le cas de la reconnaissance d'actions pré-segmentées et en quoi elle est sensée mieux modéliser une action que les approches à base d'apprentissage profond. Dans la Section 4, nous présentons et discutons les résultats expérimentaux obtenus sur deux bases d'actions squelettiques, dont HDM05 [4] et MSRC-12 [5]. La Section 5 conclut ce papier et présente des propositions d'amélioration futures.

2 État de l'art

Très peu d'approches ont par le passé considéré la problématique de détection précoce d'actions squelettiques. La plupart des travaux, notamment ceux à bases d'apprentissage profond, se sont en effet focalisés sur le problème beaucoup plus simple de la reconnaissance d'actions présegmentées. Nous décrivons donc dans un premier temps les principales architectures à base d'apprentissage profond traitant de la reconnaissance. Dans un deuxième temps, nous présentons les approches précédentes ayant porté sur la détection précoce d'actions squelettiques.

Approches de reconnaissance à base d'apprentissage profond. Récemment, plusieurs architectures utilisant des réseaux de neurones récurrents (RNN), en particulier des LSTM, ont été proposées pour la reconnaissance d'actions 3D à base de squelette. Du et al. [6] ont proposé une architecture composée de plusieurs réseaux RNN bidirectionnels structurés de façon hiérarchique. Pour ce faire, la structure squelettique humaine a été divisée en cinq groupes articulaires majeurs. Zhu et al. [7] ont introduit un terme de régularisation à la fonction objective du réseau LSTM pour pousser l'ensemble du cadre vers l'apprentissage des relations de cooccurrence entre les articulations pour la reconnaissance d'actions. Une technique d'abandon au sein de l'unité LSTM a également été introduite dans [7]. Shahroudy et al. [8] ont proposé de diviser la cellule de mémoire LSTM en sous-cellules pour pousser le réseau à apprendre les représentations de contexte pour chaque partie du corps séparément.

Toutes ces architectures de reconnaissance partagent le fait d'utiliser des LSTM pour modéliser la dépendance temporelle entre les frames. Cette modélisation est en fait basée sur une recherche en mode force brute, sans indications à priori pour cibler et guider cette recherche. La proposition que nous formulons dans ce papier tente justement de rechercher de manière plus transparente et explicite la dépendance temporelle entre ces frames.

Approches de détection précoce. La plupart des approches traitant du problème de précocité sont en réalité des approches de reconnaissance précoce et non pas de détection précoce. Ces approches étendent les méthodes populaires de reconnaissance d'actions en tentant de réduire le nombre de frames observées avant la reconnaissance.

Au contraire, certains auteurs ont proposé des approches considérant le cas de la détection précoce. En effet, Huang et al. [3] ont proposé une des premières approches de détection précoce consistant en un ensemble de détecteurs d'événements séquentiels appelés SMMED. Cette méthode consiste à rejeter séquentiellement les classes les plus improbables en analysant des segments de taille croissante (composés d'un nombre de frames croissant), jusqu'à ce qu'une classe puisse être identifiée. Une approche similaire basée sur le modèle bayésien naïf multinomial a été introduite dans [2]. Plus récemment, Bloom et al. [1] ont proposé une approche de détection précoce à base de données squelettiques, qui recherche des correspondances au moyen d'une déformation temporelle DTW entre des segments d'actions de test et des modèles appris. La subtilité de leur approche réside dans la procédure de construction des modèles d'actions (*action templates*) et qui permet d'assurer une plus grande indépendance par rapport au style d'exécution de chaque utilisateur lors des tests.

Ces approches ont pour élément commun la construction

sur la base d'une fenêtre glissante temporelle des segments d'actions considérés. En effet, les fenêtres déterminant les segments de mouvement à analyser sont indexées sur le flot temporel. Or, ceci ne permet pas d'adresser le problème de variabilités des vitesses d'exécution, amplifié dans le cas non segmenté. De plus, étant donné que ces approches analysent le flot d'entrée à un seul niveau, elles ne considèrent pour la prise de décision à un instant donné qu'une information à court, moyen ou long terme et pas les trois en même temps. Ceci est d'autant plus important que les actions peuvent avoir des parties communes et s'étaler sur des périodes plus ou moins longue. De ce fait, l'approche que nous présentons dans ce papier tiens compte de ces deux facteurs.

3 Approche proposée

L'approche que nous proposons pour la détection précoce est constituée de trois modèles, chacun opérant sur le flot d'entrée de frames squelettiques suivant le nouveau principe de fenêtres curvilignes. De ce fait, dans cette Section nous expliquons d'abord comment opèrent ces fenêtres curvilignes et quelles sont les représentations constituées à chaque nouvelle frame. Ensuite nous décrivons notre approche de détection précoce. Enfin, nous expliquons comment le concept de fenêtre curviligne peut être adapté pour la reconnaissance d'actions pré-segmentées et en quoi il est un complément voire même une alternative aux approches à base d'apprentissage profond.

3.1 Segmentation curviligne

Une approche qui traite un flot continu de frames doit se baser sur une procédure pour déterminer la taille des segments considérés à l'arrivée de chaque nouvelle frame. Les

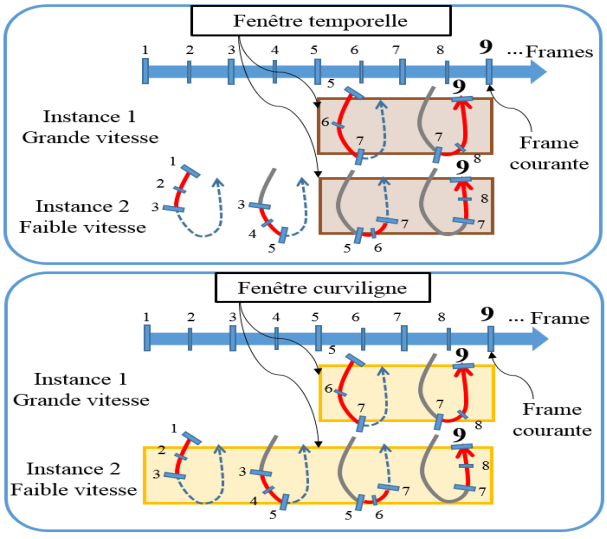

FIGURE 1 – Mise en évidence de la différence entre une fenêtre *curviligne* et une fenêtre *temporelle* standard. Les deux gestes illustrés appartiennent à la même classe mais sont produits à deux vitesses différentes.

approches de la littérature opèrent suivant des fenêtres indexées au flot temporel. Or, étant donné les variabilités des vitesses d'exécution, il est vain de vouloir trouver une taille temporelle à ces fenêtres. C'est la raison pour laquelle nous introduisons le concept de segmentation curviligne.
Ce concept consiste à définir dynamiquement des fenêtres en fonction de la quantité d'informations (c'est-à-dire de mouvement) disponible dans le flot d'entrée (Figure 1). La taille de la fenêtre est ainsi contrainte par le déplacement curviligne du squelette au lieu d'être indexée au flot temporel habituel. L'adaptation de la taille des fenêtres curvilignes permet de capturer une même quantité d'information à chaque frame. Ainsi, les représentations extraites sur les fenêtres curvilignes pendant les deux phases d'entraînement et de test sont cohérentes dans la mesure où elles représentent des (segments de) mouvements avec une quantité similaire d'informations quelle que soit la vitesse d'exécution.
À cette fin, nous utilisons comme métrique pour mesurer la quantité d'information, le déplacement curviligne des articulations. Nous introduisons donc la fonction $CuDi(F_S, F_E)$ qui permet le calcul du déplacement curviligne pour un segment de mouvement donné, en partant de la frame F_S et en s'achevant à la frame F_E, comme suit :

$$CuDi(F_S, F_E) = \sum_{i=F_S}^{i=F_E} d_i^{Moy} \tag{1}$$

Où d_i^{Moy} est le déplacement moyen instantané, calculé pour chaque frame i comme indiqué dans l'équation 2 :

$$d_i^{Moy} = \sqrt{\sum_{j=1}^{M} (d_i^j)^2} \tag{2}$$

Où M est le nombre d'articulations considérées et d_i^j le déplacement instantané de chaque articulation j, calculé comme suit :

$$\begin{aligned} d_i^j &= \sqrt{(\Delta x_i^j)^2 + (\Delta y_i^j)^2 + (\Delta z_i^j)^2}, \\ \Delta x_i^j &= x_i^j - x_{i-1}^j, \quad \Delta y_i^j = y_i^j - y_{i-1}^j, \\ \Delta z_i^j &= z_i^j - z_{i-1}^j \end{aligned} \tag{3}$$

Avec $1 \leq j \leq M$ et $x_i, y_i, z_i, x_{i-1}, y_{i-1}, z_{i-1}$ les coordonnées 3D des articulations correspondantes dans la frame courante i et la frame précédente $i-1$, respectivement.
Sur la base de la métrique définie dans l'équation 1, nous définissons une fenêtre curviligne comme une fenêtre glissante dont la taille temporelle est continuellement mise à jour de sorte qu'elle englobe, à chaque frame, un mouvement ayant un déplacement curviligne spécifique. Par exemple, à une frame donnée F_t, la fenêtre curviligne devrait englober le segment de mouvement se terminant à cette frame F_t et commençant à la frame F_S. La trame F_S est déterminée de telle sorte que :

$$S = \max_{s_i} CuDi(F_{s_i}, F_t) \geq \rho \quad ; 1 \leq s_i < t \qquad (4)$$

Avec ρ un seuil de déplacement curviligne spécifique à chaque classifieur.

Nous entrainons ensuite plusieurs classifieurs, où chacun se base sur une fenêtre curviligne de taille spécifique. En effet, due aux variabilités cette fois-ci inter-classes, il est nécessaire de considérer plusieurs classifieurs de différentes tailles curvilignes. Nous considérons alors dans ce papier le cas général où toutes les classes d'actions ont des tailles curvilignes différentes et donc nous entrainons autant de classifieurs qu'il y a de classes d'actions. En pratique, le nombre de classifieurs pourrait être réduit si certaines actions ont des déplacements curvilignes similaires, ce qui permettrait de réduire le temps de traitement de chaque frame. Chacun de ces classifieurs est entrainé pour reconnaître toutes les classes d'actions [1] et pas uniquement pour reconnaitre la classe à laquelle il est associé. Par conséquent, la sortie de chacun de ces classifieurs est une étiquette gestuelle, c'est-à-dire l'une des étiquettes de classes d'actions, et n'est pas seulement une sortie binaire.

Ainsi pour former un classifieur donné Ci, nous calculons d'abord la taille curviligne $CuDi_i$ associée à la classe d'action Gi. Il s'agit de la moyenne de tous les déplacements curvilignes des instances d'apprentissage appartenant à la classe Gi. Ensuite, au moyen de chaque fenêtre curviligne de taille $CuDi_i$, les instances de l'ensemble d'apprentissage appartenant aux différentes classes sont analysées en partant de leur début jusqu'au point d'action [2]. Au cours de ce processus, des descripteurs locaux sont extraits selon cette fenêtre curviligne pour construire l'ensemble d'apprentissage du classificateur Ci. Nous avons retenu comme descripteurs locaux les descripteurs HIF3D conçus par [10] pour modéliser efficacement des trajectoires squelettiques 3D. De plus, chacun des classifieurs est un SVM multi-classes entrainé suivant la stratégie un-contre-tous. Nous utilisons pour celà la bibliothèque LIBSVM [11].

3.2 Détection précoce d'actions dans un flot non segmenté

L'approche de détection précoce que nous proposons est constituée de trois modèles curvilignes. Ces modèles curvilignes permettent le traitement du flot d'entrée respectivement à court, moyen et long terme. Chacun de ces modèles est composé des classifieurs précédemment appris et où chacun opère suivant une fenêtre de taille curviligne spécifique. Comme expliqué dans la Section 3.1, chaque classifieur est associé à une classe d'action de façon à ce que la taille de la fenêtre qu'il utilise s'obtient en moyennant les déplacements curvilignes de toutes les instances de cette classe d'action. Ainsi, pour le premier modèle (modèle à *long terme*), les tailles curvilignes utilisées sont égales à 100% de ces valeurs moyennes. Pour le deuxième modèle (modèle à *moyen terme*), les tailles curvilignes utilisées sont égales à 50% des valeurs moyennes. Enfin, pour le dernier modèle (modèle à *court terme*), les tailles curvilignes utilisées sont égales à 10% des valeurs moyennes.

Tout au long du traitement d'un flot de frames, ces trois modèles sont lancés et opèrent en parallèle. Chacun de ces modèles est associé à son propre système de décisions. Un système de combinaison reçoit à chaque frame les décisions de chacun et permet de décider de la classe d'action finale. Ci-après, nous expliquons d'abord comment opère chacun des modèles en local. Nous présentons ensuite le système de combinaison des décisions émises par chaque modèle.

Prise de décisions locales. Avant d'émettre une décision, chaque modèle doit en local combiner les décisions émanant des classifieurs le constituant. Afin de rendre plus robuste ces décisions, nous comptabilisons les scores de confiance qu'émet chaque classifieur dans un histogramme de façon à favoriser la décision ayant le plus grand score. C'est sur la base de cet histogramme que le modèle formule sa décision.

L'histogramme en question est, noté His_Modele, est composé d'autant d'entrées qu'il y a de classifieurs (Figure 2). Chaque entrée i correspond à la classe $Predit\_i$ prédite par le classifieur Ci à la frame courante avec son score cumulé depuis le début du traitement. Ces scores sont ensuite utilisés au sein du modèle afin d'assurer un compromis entre le degrés de confiance accordé à la classe détectée et la latence de détection. Pour ce faire, nous calculons expérimentalement une matrice de seuil, notée *ThreshMat*, déterminant le seuil qu'un classifieur Ci doit atteindre pour une classe donnée Gj. *ThreshMat* est une matrice $m \times n$ où m est le nombre de classifieurs et n le nombre de classes. Nous considérons ici le cas général où il y a autant de classifieurs que de classes, c'est-à-dire $m = n$.

De plus, la façon dont les valeurs de *ThreshMat* sont déterminées est cruciale pour assurer l'équilibre entre la réduction de la latence de détection et l'augmentation de la robustesse aux faux positifs. A cette fin, nous calculons d'abord deux matrices : *PrecoMat* et *ConfMat*. *PrecoMat* est composée de seuils de confiance cumulés qu'un classifieur donné pourrait idéalement atteindre pour chaque classe. Au contraire, *ConfMat* contient des valeurs de seuil minimales qu'un classifieur doit dépasser pour éviter de confondre différentes classes. Nous expliquons plus loin dans cette Section comment les valeurs de ces deux matrices sont utilisées pour calculer celles de *ThreshMat*.

Ainsi, une classe *Gj* prédite par un classifieur *Ci* est considérée comme la classe finale si le score cumulé par ce classifieur *Ci* pour cette classe *Gj* dépasse le seuil $\theta_{i,j} = ThreshMat(i, j)$. A noter que tout classifieur entraîné avec une taille de fenêtre curviligne spécifique est capable de prédire toute classe à partir du moment où ce classifieur

1. A la condition que ces actions produisent au moins autant de déplacement curviligne que celui requis par le classifieur considéré.

2. Le point d'action introduit par [9] ne correspond pas à la fin effective d'une action mais à l'instant auquel la présence de l'action est claire et peut être distinguée des autres classes et identifiée de façon unique.

est suffisamment sûr de sa décision. Nous résumons le processus de décision de chaque modèle dans l'équation 5.

$$Output\_p = \begin{cases} \textbf{Gj} \quad , \quad \text{Si} \quad \exists \quad 1 \leq i,j \leq n \quad \& \\ \qquad\qquad His\_Modele(i) \geq \theta_{i,j} \quad \& \\ \qquad\qquad Output\_Ci = Gj \\ \textbf{?} \quad , \quad \text{sinon} \end{cases} \tag{5}$$

Où $Output\_p$ est la classe prédite par le modèle $p = 1, 2, 3$, correspondant respectivement aux approches utilisant $\alpha = 100, 50, 10\%$. $Output\_Ci$ est la classe prédite par le classifieur Ci.

L'équation 5 résume le fait que la décision émise par chaque modèle est égale à un "?" tant qu'aucun classifieur n'a dépassé le score d'aucune des classes. De plus, si le modèle hésite entre deux classes potentielles ou plus, la décision est reportée jusqu'à ce qu'il ne reste qu'une seule classe. A la fin du processus de décision, tous les scores sont réinitialisés à zéro, de même que les déplacements curvilignes cumulés pour chaque classifieur. Cette ré-initialisation est nécessaire dans un flot non segmenté afin d'éviter que plusieurs détections soient émises pour une même action (faux positifs). Une illustration du fonctionnement de l'histogramme est proposée dans la Figure 2.

Combinaison des décisions. Considérons à présent la combinaison des décisions qu'émet chacun des trois modèles. La procédure de combinaison dépend en réalité de la finalité et du sens donné à la notion de détection précoce. En effet, nous distinguons dans ce papier deux interprétations possibles de la détection précoce.

D'une part, la première interprétation consiste à procéder exactement comme pour les approches OAD mais devoir en plus le faire au plus tôt. D'autre part, suivant la deuxième interprétation, notamment utilisée dans les travaux de Bloom et al. [1], l'évaluation est menée frame

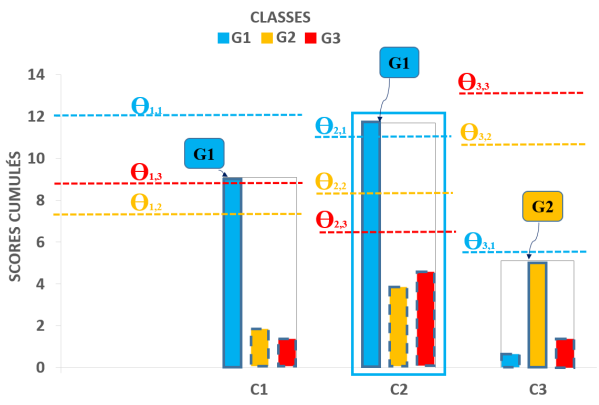

FIGURE 2 – Illustration du fonctionnement de l'histogramme d'un modèle pour trois classifieurs C1, C2 et C3 et trois classes G1, G2 et G3, à un instant de prise de décision effective.

par frame de telle sorte que les fausses détections qui surviennent souvent au début de l'action ne sont pas comptabilisées dans le calcul de la performance globale de l'approche. Ce deuxième cas représente en fait une version très simplifiée de la détection précoce (plus simple même que la OAD). En effet, la première interprétation est beaucoup plus contraignante (aussi bien par rapport à la deuxième interprétation que par rapport aux approches de la OAD) car, tout comme pour une approche de la OAD, toute mauvaise détection est comptabilisée pour le calcul de la performance globale comme étant un faux positif. Nous présentons ci-après la procédure de combinaison la plus appropriée pour chacun de ces deux cas.

Pour ce qui est du premier cas, où l'objectif est d'être le plus précoce possible en réduisant le risque de faux positifs, la procédure de combinaison se base d'abord sur la manière de déterminer les valeurs des seuils de la matrice *ThreshMat* pour chaque modèle. Comme expliqué plus haut dans cette Section, la matrice des seuils *ThreshMat* est une combinaison linéaire des matrices de confusion et de précocité définies précédemment. Cette combinaison est néanmoins différente pour chacun des modèles. En fait, pour assurer une équité entre ces trois modèles, la combinaison est inversement proportionnelle au pourcentage des tailles curvilignes défini pour chacun des modèles. Ceci correspond au fait que plus la taille curviligne d'un modèle est importante, moins le score qu'il doit cumuler pour se prononcer est grand.

Nous proposons alors que les valeurs $\theta_{i,j} = ThreshMat(i,j)$ soient déterminées comme suit :

$$\theta_{i,j} = \gamma_p \times PrecoMat(i,j) + (1 - \gamma_p) \times ConfMat(i,j) \tag{6}$$

Où γ_p = 1/4, 1/2, 3/4 pour $p = 1, 2, 3$ respectivement. Comme mentionné plus haut, les modèles d'indice $p = 1, 2, 3$, correspondent respectivement aux approches utilisant $\alpha = 100, 50, 10\%$.

Ainsi, comme pour une combinaison de classifieurs dans un seul modèle, la décision finale n'est pas prise tant qu'aucun modèle n'en a émis une. De plus, si les modèles émettent des décisions différentes, nous attendons jusqu'à ce qu'il n'en reste qu'une seule. Une fois que la décision finale est prise, tous les scores sont remis à zéro ainsi que les distances curvilignes cumulées par les modèles pour se préparer à détecter un prochaine action dans le flot continu de frames.

S'agissant à présent du second cas expérimental, celui de la détection précoce simplifiée et suivant lequel les travaux de Bloom et al. [1] ont été menés, la contrainte relative à la robustesse aux faux positifs n'a plus lieu d'être. En particulier, il n'y a plus besoin d'attendre qu'un classifieur soit suffisamment sûr de la classe qu'il identifie pour émettre une décision finale. De ce fait, les valeurs $\theta_{i,j}$ des matrices $ThreshMat$ regroupant les seuils de confiances de chacun des classifeurs Ci pour chaque classe Gj sont mises à zero et ce pour les trois modèles introduits précé-

demment.
De plus, nous avons opté pour une combinaison simple des décisions fournies par les trois modèles constituant notre approche. Cette combinaison est résumée dans l'équation ci après :

$$Output = \begin{cases} \textbf{Gi} \quad , \quad \text{Si} \quad \exists \quad 1 \leq j \neq k \leq 3 \quad \& \\ \qquad\qquad\quad Output\_j = Gi \quad \& \\ \qquad\qquad\quad Output\_k = Gi \\ \textbf{Gi} \quad , \quad \text{Sinon Si} \quad Output\_1 = Gi \\ \textbf{?} \quad , \quad \text{Sinon} \end{cases} \tag{7}$$

Le principe de combinaison résumé dans l'équation 7 traduit le fait que si au moins deux modèles prédisent la même classe d'action **Gi** alors la frame est labellisée avec cette classe. Au contraire, s'il n'y a pas de consensus entre au moins deux modèles, alors la décision finale prise est celle émise par le modèle de plus petite taille curviligne (10%). Ce choix est motivé par une volonté de précocité de façon à ce que s'il n'y a pas de consensus c'est probablement parce que l'action est à son début et c'est donc au classifieur à 10% de décider. Enfin, et contrairement au fonctionnement standard des modèles, les distances curvilignes cumulées ne sont pas remise à zero à chaque nouvelle détection vu que l'évaluation est faite pour chaque frame. Dans ce cas les faux positifs ne sont pas comptabilisés.

3.3 Adaptation pour la reconnaissance d'actions pré-segmentées

L'approche de détection peut être adaptée au cas de la **reconnaissance d'actions pré-segmentées**. Pour cela, nous utilisons un seul modèle, celui basé sur 100% des tailles curvilignes, et sans utilisation des seuils de confiance de la matrice *ThreshMat*. De plus, à chaque frame de la séquence de test, un seul classifieur est lancé, au lieu de plusieurs classifieurs en parallèle, puisque nous connaissons exactement le début de l'action en cours.
En effet, à partir du début de la séquence de test, le déplacement curviligne cumulé est calculé à chaque frame F_t et si ce déplacement dépasse la taille curviligne $CuDi_i$ d'un classifieur donné C_i alors ce classifieur est activé. Parmi les classifieurs activés C_i à la frame courante F_t, nous considérons seulement le classifieur C_j qui correspond à la plus grande fenêtre curviligne $CuDi_j$ (Équation 8).

$$CuDi_j = \max_{F_t} CuDi_i \tag{8}$$

Les descripteurs HIF3D [10] ne sont ensuite extraits que selon la fenêtre curviligne correspondante de taille $CuDi_j$. Le but d'utiliser à chaque frame un seul classifieur activé, celui avec la plus grande taille curviligne, est double. D'une part, cela nous permet de considérer à chaque frame le plus grand segment de mouvement afin de prendre en compte autant d'informations que possible. D'autre part, cela nous permet de ne considérer qu'un segment de mouvement approprié (et pas nécessairement toute la séquence) ce qui garantit une cohérence entre ce segment de mouvement et ceux utilisés pour l'apprentissage des classifieurs curvilignes. Ceci correspond à une recherche ciblée des relations temporelles dans une séquence et peut être plus performante qu'une recherche non guidée comme effectuée par les approches d'apprentissage profond.
En outre, au fur et à mesure que la séquence de test est traitée, un histogramme est mis à jour à chaque frame avec la décision de sortie et le score de confiance émis par le classifieur. En fait, pour la reconnaissance d'actions pré-segmentées, un seul histogramme est utilisé. Cet histogramme est différent de celui utilisé en détection précoce car chaque entrée correspond à une classe et pas à un classifieur. Ainsi, à chaque fois qu'une classe Ci est détectée par le classifieur utilisé Cj, la valeur de l'histogramme correspondant à la i^{me} entrée est incrémentée avec le score de la classe prédite, i.e. *Ci*. A la fin de la séquence traitée, la classe finale est celle associée au score le plus élevé de l'histogramme.

4 Résultats expérimentaux

Nous présentons dans cette Section les résultats obtenus d'abord dans le cadre de la reconnaissance d'actions pré-segmentées, et ensuite celui de la détection précoce d'actions dans un flot non segmenté.

4.1 Résultats de reconnaissance pré-segmentées

Bien que le principal objectif de notre papier est d'adresser le problème de détection précoce dans un flot non segmenté, cette expérience vise à se faire une idée des performances de notre approche par rapport aux approches de l'état de l'art, en particulier les méthodes basées sur l'apprentissage profond, pour modéliser et reconnaître des actions squelettiques pré-segmentées. Cette expérience est réalisée en utilisant la base de données HDM05 [4]. Il s'agit d'une base de données squelettiques qui a été collectée via un marqueur optique. Cette base contient une centaine de classes de mouvement, dont divers mouvements de marche et de course, des mouvements de rotation et de saisis, etc.
Pour cette expérimentation, nous adoptons le protocole d'évaluation proposé par [12] où 65 classes ont été considérées et une validation croisée de 10 folds a été menée.
La Table 1 rapporte les résultats en terme de précision moyenne. Dans l'ensemble, notre approche fait mieux que les approches basées sur un apprentissage profond, et obtient un score de **99.4%**. Tout d'abord, l'approche proposée obtient de meilleurs résultats que l'approche proposée dans [12] à base d'un seul perceptron multicouche. Cette première constation suggère que l'utilisation de plusieurs classifieurs spécialisés assure une meilleure modélisation du mouvement que l'utilisation d'un seul modèle. Deuxièmement, par rapport aux réseaux récurrents hiérarchiques

Approche	Taux de reco. (%)
HIF3D + SVM + Level = 2 [10]	91.0
Multi-layer Perceptron [12]	95.6
Hierarchical RNN [6]	96.9
Deep LSTM [7]	96.8
Deep LSTM + Co-occurrence [7]	97.0
Deep LSTM + Simple Dropout [7]	97.2
Deep LSTM + In-depth Dropout [7]	97.3
Deep LSTM + Co-occurrence + In-depth Dropout [7]	97.3
Our	**99.4**

TABLE 1 – Reconnaissance, HDM05 : Comparaison des résultats de notre approche avec ceux obtenus par les approches de l'état de l'art sur la base HDM05.

entrainés séparément sur cinq parties du squelette humain [6], il semble que considérer le squelette entier et diviser plutôt le mouvement en plusieurs segments curvilignes est susceptible de mieux modéliser le mouvement. Enfin, notre approche fait mieux que toutes les architectures à base de LSTM qui sont proposées dans [7].

Cette performance pourrait être due à deux spécificités de notre approche. Le fait que notre approche traite un mouvement de manière progressive, c'est-à-dire frame par frame, et cumule les scores le long de ce traitement, augmente la robustesse de la décision finale. Cette décision, qui n'est pas nécessairement la dernière classe d'action prédite, est en fait basée sur la détection multiple par plusieurs classifieurs à différents niveaux spatio-temporels. De plus, notre approche améliore la recherche de relations temporelles à l'intérieur d'un mouvement en focalisant et guidant cette recherche sur des sous-segments de tailles curvilignes prédéfinies. Cette façon de faire est meilleure que ce que font les LSTM, dont la recherche de telles relations est entièrement automatique et ne prend pas en compte les spécificités des actions modélisées pour guider cette recherche.

4.2 Résultats de détection précoce d'actions dans un flot non segmenté

Nous présentons dans cette Section les résultats de deux expérimentations menées sur la base MSRC-12 [5] afin d'évaluer les deux systèmes proposés pour faire de la détection précoce. La base de données MSRC-12 contient des séquences de données squelettiques, consistant en 20 positions articulaires. Elle comprend 12 classes d'actions effectuées par 30 sujets pour un total de 594 séquences. Les données de cette base sont répertoriées en cinq catégories suivant les cinq modalités de collecte, à savoir : images, texte, vidéo, images + texte et vidéo + texte. Pour permmetre une compraison avec les résutlats présentés dans [1], nous utilisons uniquement les données de la modalité "vidéo + texte" en suivant le même protocole de validation croisée à 10 folds.

La première expérimentation est conduite suivant l'interprétation la plus répandue de la détection précoce, et que nous avons qualifiée précédemment de détection précoce simplifiée. Dans ce premier cas, l'évaluation est menée indépendamment pour chaque frame située entre le point

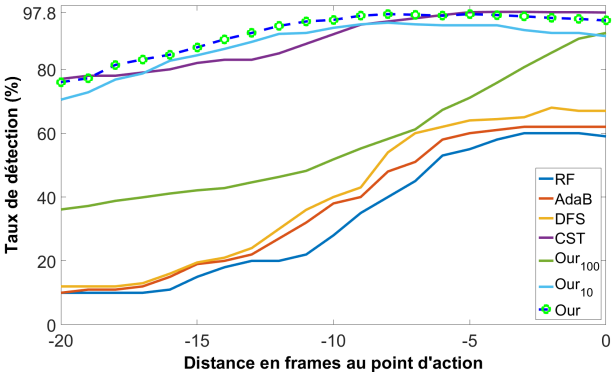

FIGURE 3 – Détection précoce simplifiée, MSRC-12 : Résultats obtenus pour 20 frames avant les points d'action sur la base MSRC-12 en détection précoce simplifiée. Our = notre approche ; Our_10 et Our_100 sont les modèles à 10% et 100% de tailles curvilignes ; Se referer à [1] pour les autres approches.

d'action et à moins 20 frames de ce point d'action. Une moyenne est ainsi donnée pour chaque frame dans cet intervalle. Les résultats de notre approche et ceux obtenus par les précédentes approches sont rapportés dans la Figure 3. Nous rapportons aussi dans cette figure les résultats de variantes de notre approche, à savoir **Our-10** et **Our-100**, utilisant un seul modèle ayant une taille de 10% et 100%, respectivement.

Sur la base des résultats de la Figure 3, il est possible de voir que l'approche à trois modèles est globalement meilleure que la plus performante des approches proposée par [1]. En particulier, notre approche est nettement supérieure pour les instants les plus éloignées du point d'action. Ceci suggère notamment que, du point de vue de la précocité, notre approche est plus intéressante même si l'approche de [1] prend un peu le dessus en se rapprochant des points d'action. Cette supériorité traduit une meilleure capacité d'identification qui est notamment due d'une part à la puissance de la notion de fenêtres curvilignes qui permettent d'adresser les problèmes de variabilités temporelles et d'autre part à la puissance de représentation des descripteurs HIF3D [10]. En outre, par rapport aux deux courbes, traduisant les performances des deux approches de **Our-10** et **Our-100**, notre approche est supérieure et permet en effet de combiner les avantages de chacune d'elles.

Dans la dernière expérimentation, nous évaluons notre approche dans le cadre plus complexe mais plus réaliste de la détection précoce (première interprétation). Le même protocole de validation croisée est suivi. Néanmoins lors de l'évaluation, nous calculons la mesure F_score qui combine précision et rappel et dans laquelle l'ensemble des faux positifs et des faux négatifs sont comptabilisés. Les résultats sont donnés sous forme de courbe dans la Figure 4.

Vu qu'aucune approche n'a considéré auparavant ce contexte d'évaluation, nous avons rapporté non seulement

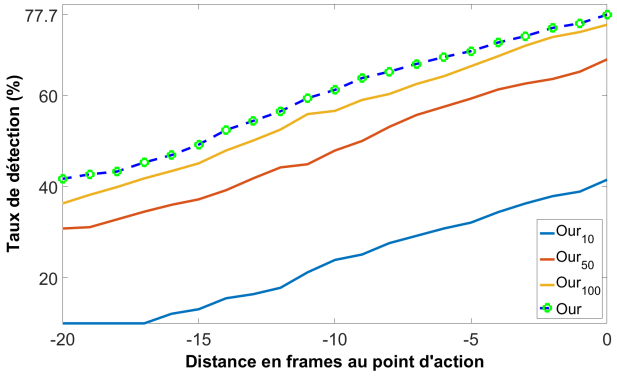

FIGURE 4 – Détection précoce, MSRC-12 : Résultats obtenus pour 20 frames avant les points d'action sur la base MSRC-12 en détection précoce.

les résultats de notre approche (**Our**) mais aussi ceux des trois modèles séparément, à savoir **Our_10**, **Our_50** et **Our_100**. Nous relevons alors que notre approche (**Our**) permet de combiner les avantages de chacun des trois modèles et réussit à atteindre des scores intéressants bien avant l'instant de point d'action. Ces résultats peuvent servir de référence pour de futurs travaux considérant ce cadre plus complexe mais plus réaliste de la détection précoce.

5 Conclusion

Dans ce papier nous avons exploré la problématique de la détection précoce d'actions non segmentées. Pour ce faire, nous avons proposé une approche à base de trois modèles curvilignes qui permettent de scruter le flot d'entrée à court, moyen et long terme. Ces modèles sont basés sur le concept innovant de la distance curviligne qui représente une alternative de segmentation en permettant notamment d'adresser la variabilité des vitesses d'exécution des actions. En outre, nous avons présenté une approche dérivée afin d'adresser le problème plus simple et plus classique de reconnaissance d'actions pré-segmentées. Cette adaptation vise notamment à évaluer le concept de fenêtre curviligne dans ce contexte et permettre une comparaison aux approches à base d'apprentissage profond.

L'évaluation menée sur deux bases d'actions squelettiques dans les contextes de reconnaissance et de détection précoce, a permis de montré l'intérêt de notre approche. En effet, d'une part nous avons réalisé de meilleures performances que des approches très récentes à base d'apprentissage profond dans le cadre de la reconnaissance d'actions pré-segmentées. D'autre part, nous avons montré que notre approche réalise des résultats prometteurs dans le cadre de la détection précoce.

Nos travaux futurs porteront probablement sur l'insertion de notre approche de détection précoce dans des scénarios réels et mesurer ainsi l'utilisabilité de notre approche pour fluidifier l'interaction avec la machine.

Références

[1] V. Bloom, V. Argyriou, and D. Makris, "Linear latent low dimensional space for online early action recognition and prediction," *Pattern Recognition*, vol. 72, pp. 532–547, 2017.

[2] H. J. Escalante, E. F. Morales, and L. E. Sucar, "A naive bayes baseline for early gesture recognition," *Pattern Recognition Letters*, vol. 73, pp. 91–99, 2016.

[3] D. Huang, S. Yao, Y. Wang, and F. De La Torre, "Sequential max-margin event detectors," in *Proceedings of the European conference on computer vision*, 2014, pp. 410–424.

[4] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber, "Mocap database hdm05," *Institut für Informatik II, Universität Bonn*, vol. 2, p. 7, 2007.

[5] S. Fothergill, H. Mentis, P. Kohli, and S. Nowozin, "Instructing people for training gestural interactive systems," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2012, pp. 1737–1746.

[6] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition, 2015*, 2015, pp. 1110–1118.

[7] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, X. Xie *et al.*, "Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks." in *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, vol. 2, 2016, p. 8.

[8] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "Ntu rgb+ d : A large scale dataset for 3d human activity analysis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1010–1019.

[9] S. Nowozin and J. Shotton, "Action points : A representation for low-latency online human action recognition," *Microsoft Research Cambridge, Tech. Rep. MSR-TR-2012-68*, 2012.

[10] S. Y. Boulahia, E. Anquetil, R. Kulpa, and F. Multon, "Hif3d : Handwriting-inspired features for 3d skeleton-based action recognition," in *Proceedings of the 23rd IEEE International Conference on Pattern Recognition*, 2016, pp. 985–990.

[11] C.-C. Chang and C.-J. Lin, "Libsvm : a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, p. 27, 2011.

[12] K. Cho and X. Chen, "Classifying and visualizing motion capture sequences using deep neural networks," in *Proceedings of the IEEE International Conference on Computer Vision Theory and Applications*, vol. 2, 2014, pp. 122–130.