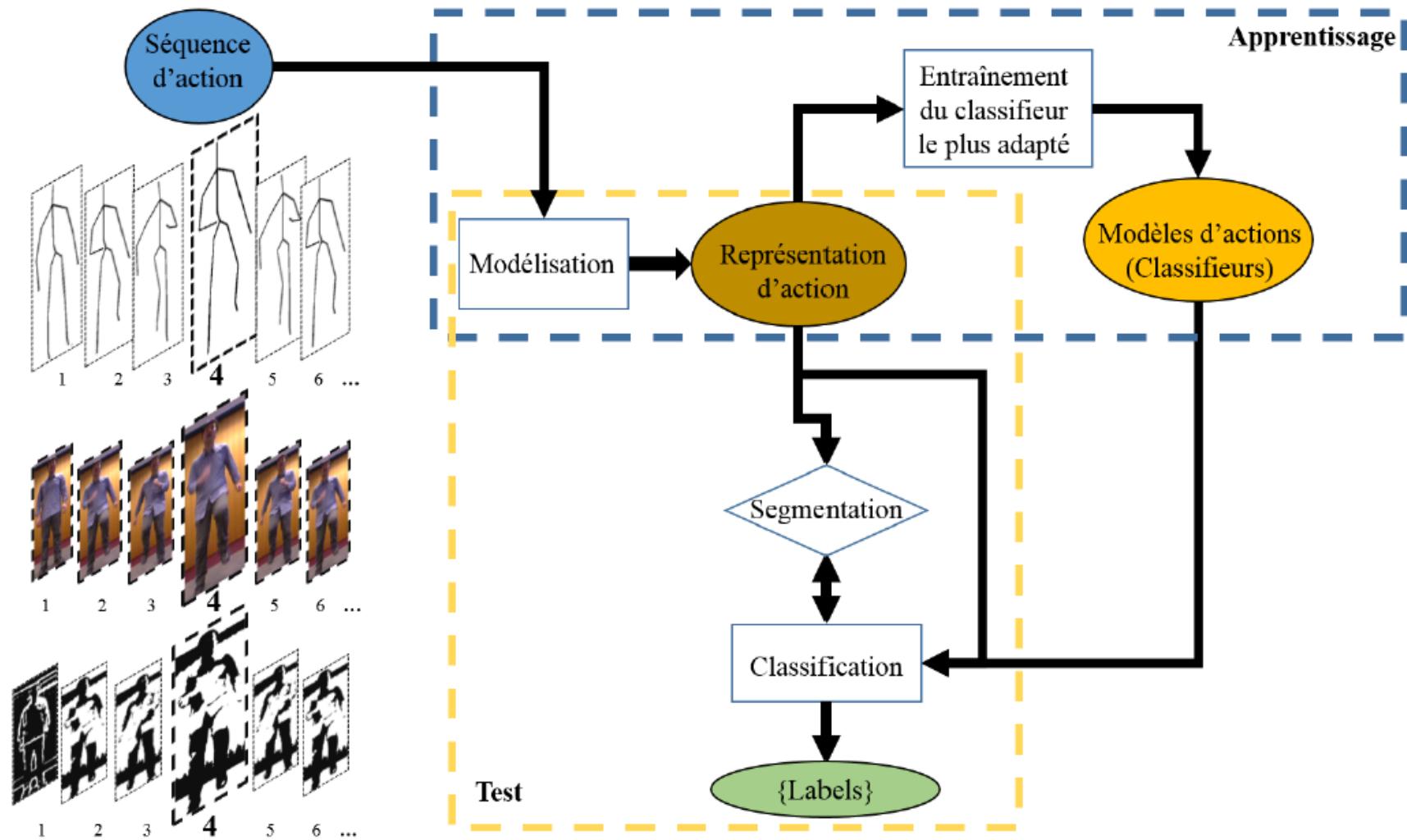


Movement and Artificial Intelligence

Human Motion Recognition
Part I
Sylvie Gibet

Classification of actions

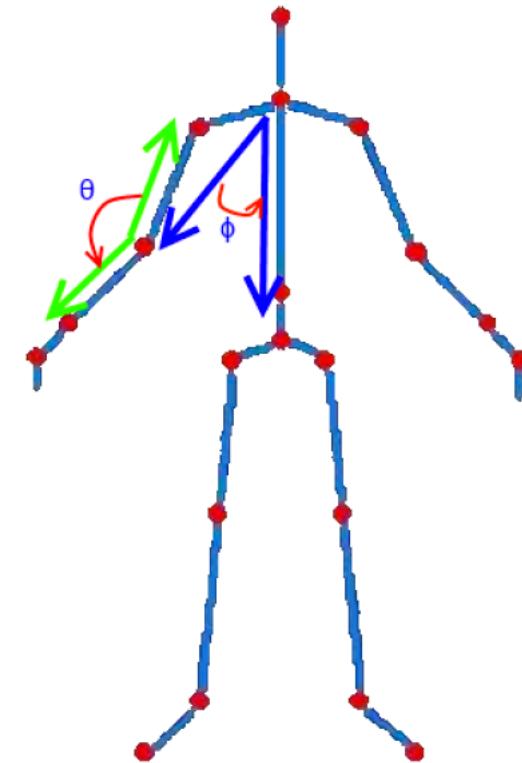
- Generic system for classifying actions Yacine Boulahia, 2018



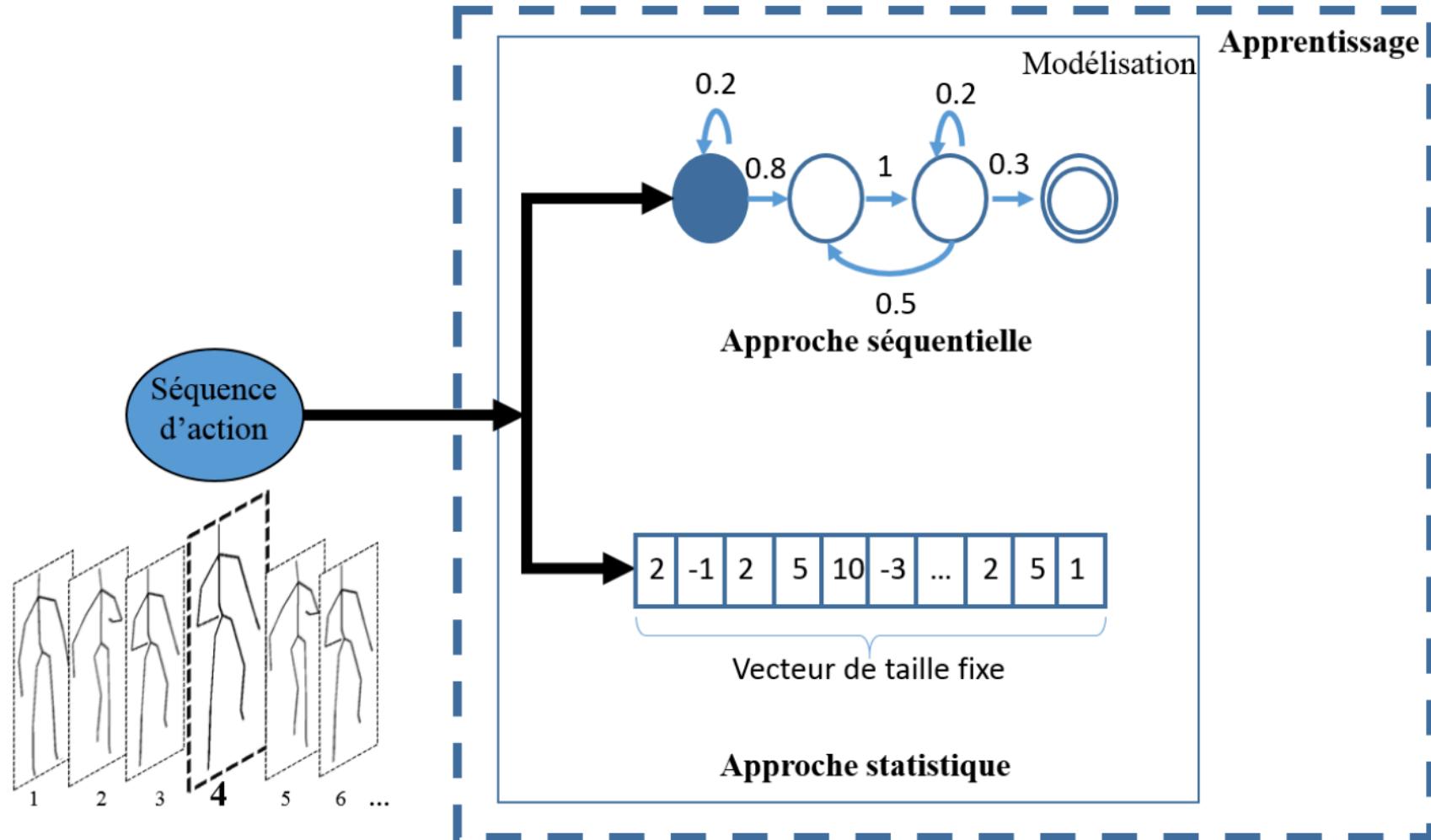
Motion representation

■ Motion

- Pattern (postures)
- Duration: sequence of actions



Motion representation



Classification of segmented actions

- Sequential approaches
- Statistical approaches
- Deep learning

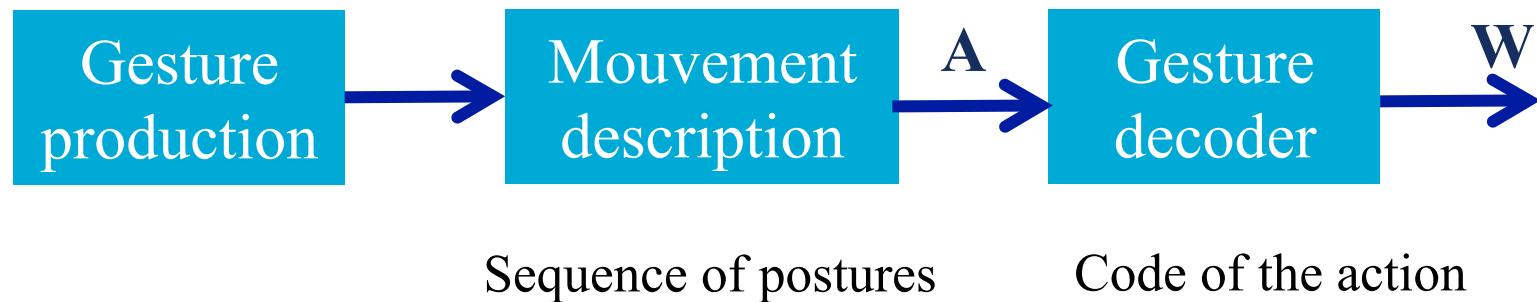
Classification of segmented actions

■ Sequential approaches

- Hidden Markov Models (HMM)
- Elastic distances (DTW: Dynamic time warping)
- Similarity measure (LCSS: Longest common subsequence)

Sequential approaches

■ HMM (gesture recognition)



- Recognition is achieved by maximizing the probability of the sequence of the meaningful gesture W , given the sequence of postures A , i.e. choose the sequence W such as:

$$P(\hat{W}|A) = \max_W P(W|A)$$

Sequential approaches

■ HMM

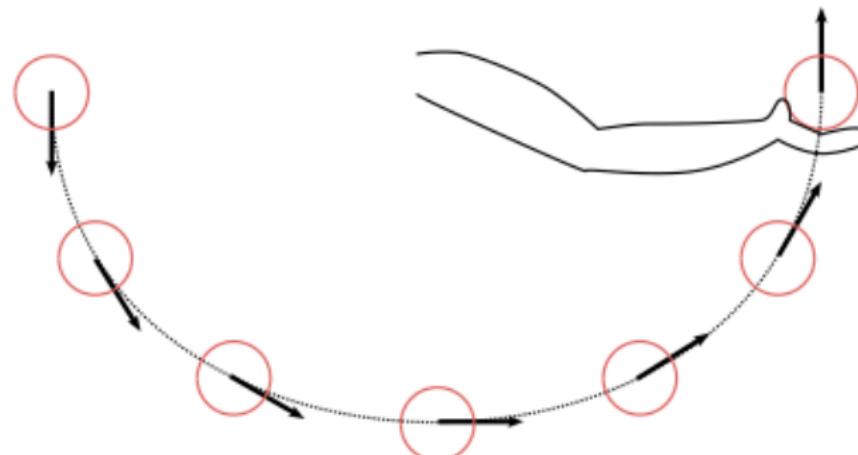


Figure 1. Movement of the example arc gesture.

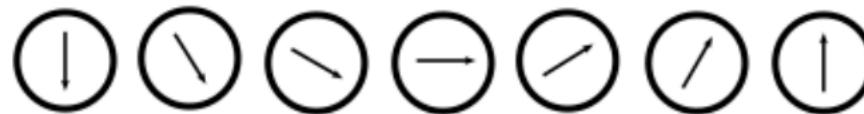


Figure 2. States of the example arc gesture (when modeled by an HMM with 7 emitting states).

Sequential approaches

■ HMM

2 approches en reconnaissance de gestes

- Modèle discret : vocabulaire de symboles discrets associés à une probabilité d'émission de chaque symbole par état
- Modèles continu : fonction de densité multinormale par état définissant la probabilité d'une observation donnée par état

Sequential approaches

- HMM
- 2 modèles :
 - Modèle ergodique : toutes les transitions d'un état vers un autre sont possibles
 - Modèle gauche-droite : modèle partiellement connecté car contenant des contraintes (plus simple, moins de données d'apprentissage)

Sequential approaches

■ HMM

- Modèle défini par : $\lambda = \{A, B, \Pi\}$
 - N : nombre d'états du modèle
 - États $s = \{s_1, s_2, \dots, s_N\}$
 - État au temps t , q_t état de s
 - M : nombre de symboles d'observations (i.e., observations discrètes)
 - Symboles $v = \{v_1, v_2, \dots, v_M\}$
 - Observation au temps t , o_t observation de v
 - $A = \{a_{ij}\}$: matrice de probabilité de transition entre états
 - $a_{ij} = P(q_{t+1} = s_j | q_t = s_i), 1 \leq i, j \leq N$
 - $B = \{b_j(k)\}$ matrice de probabilités d'émission (d'observation) des symboles dans chacun des états
 - $b_j(k) = P(v_k \text{ à } t | q_t = s_j, 1 \leq j \leq N, 1 \leq k \leq M)$
 - $\Pi = \{\Pi_i\}$: distribution des états initiaux
 - $\Pi_i = P(q_1 = s_i), 1 \leq i \leq N$

Sequential approaches

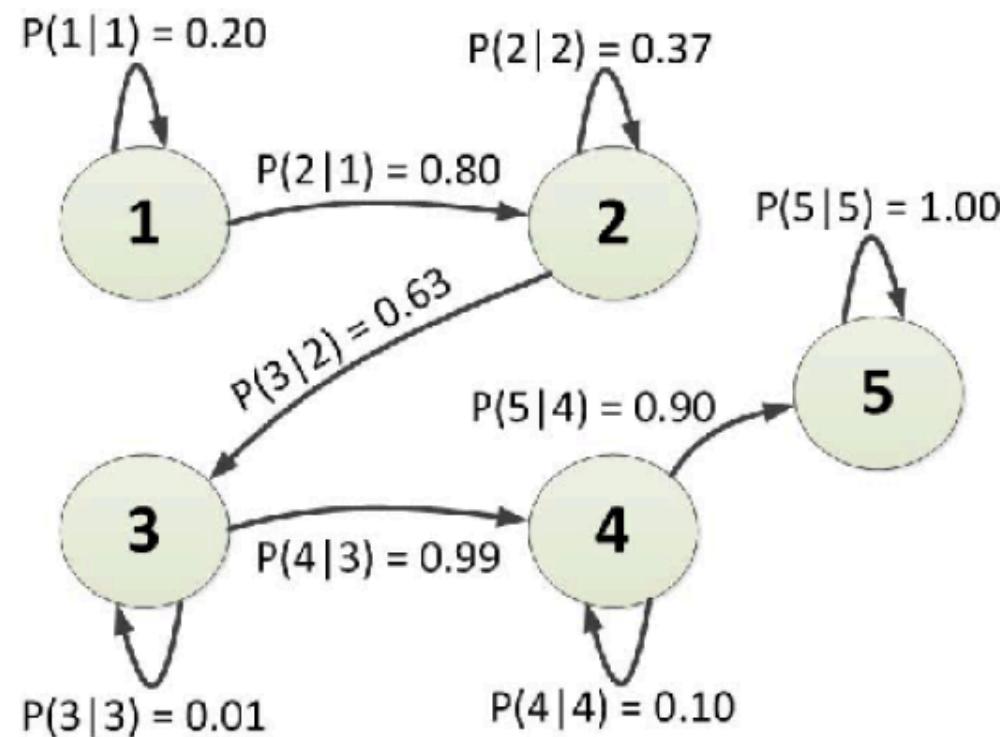
■ HMM

- Choix du nombre d'états : dépend de la complexité du processus à modéliser :
 - un nombre détats élevé peut générer un modèle trop spécifique
 - un petit nombre d'états peut conduire à des actions indiscernables
- Aucune règle dans la littérature : ci-dessous, exemple de l'action s'asseoir (modèle à 5 états gauche-droite)

Sequential approaches

■ HMM

- Ci-dessous, exemple de l'action s'asseoir (modèle à 5 états gauche-droite)



[Ghojogh et al, 2017] – IEEE Sensors Journal Benyamin Ghojogh, Hoda Mohammadzade, and Mozhgan Mokari. Fisherposes for human action recognition using kinect sensor data. IEEE Sensors Journal, 2017.

Sequential approaches

■ HMM: example

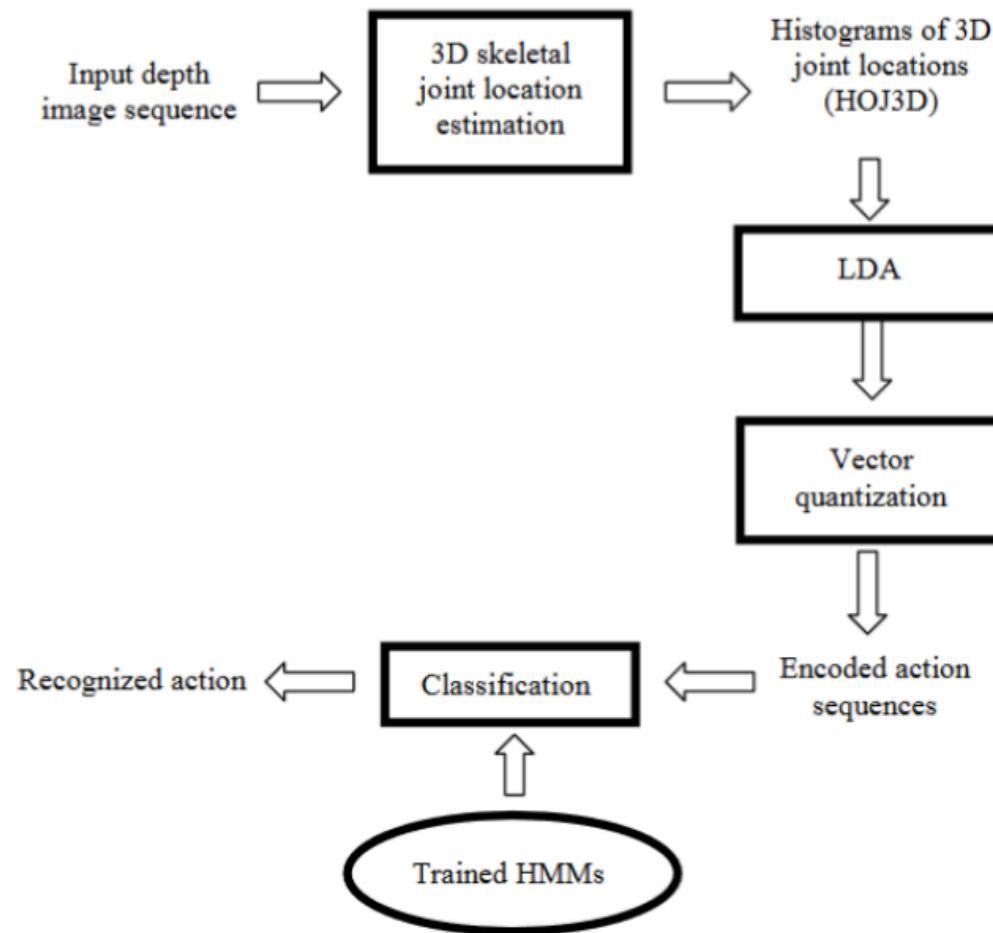
- New representation: each posture is modeled by an histogram of dim 84: weighted membership of nine selected joints to n regions that make up a sphere surrounding the subject.
- Dimension reduction: Linear Discriminant Analysis
- Puis clustering avec KMEANS: chaque groupe de postures est représenté par le n° du cluster
- HMM à 6 état entraîné par rapport à chaque classe d'action
- Viterbi algorithms: probability of generate the sequence of observation V, knowing HMM_i : P(V / HMM_i) for each model
- Predicted class:

$$C^* = \arg \max_i P(V/HMM_i), i = 1 \dots M$$

with M classes of action

Sequential approaches

■ HMM: example



[XIA et al., CVPRW 2012] – View invariant human action recognition using histograms of 3d joints

Classification of segmented actions

■ HMM

Librairie GRT (Gesture Recognition Toolkit):

<http://www.ickgillian.com/software/grt>

Voir rapport de master M1 (Clément Reverdy)

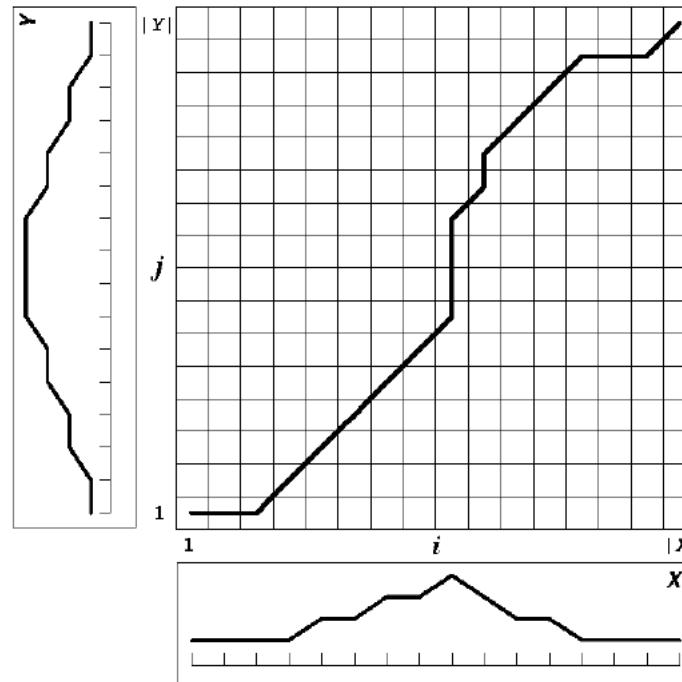
Reconnaissance des actions sur les données :

- MSRAction (données Kinect)
- HDM 05

Sequential approaches

■ DTW

- Elastic distances (DTW: Dynamic time warping)
 - Reconnaissance de la parole, de l'écriture, de la langue des signes ([Bellman, 1960], Bellman et Kalaba, 1959])



Sequential approaches

■ DTW

- Elastic distances (DTW: Dynamic time warping)
 - Reconnaissance d'actions dans un flot non segmenté : stream-DTW (proposé par Dupont 2017)
 - Calcul de façon incrémentale la valeur de similarité entre une séquence de référence et une potentielle action située dans une fenêtre qui glisse sur le flux non segmenté
 - En effet, recalculer pour chaque fenêtre la DTW serait trop coûteux en temps réel ($O(m.n)$) et empêcherait la reconnaissance en temps réel
 - Mise à jour de la valeur de similarité en temps linéaire

Sequential approaches

■ DTW

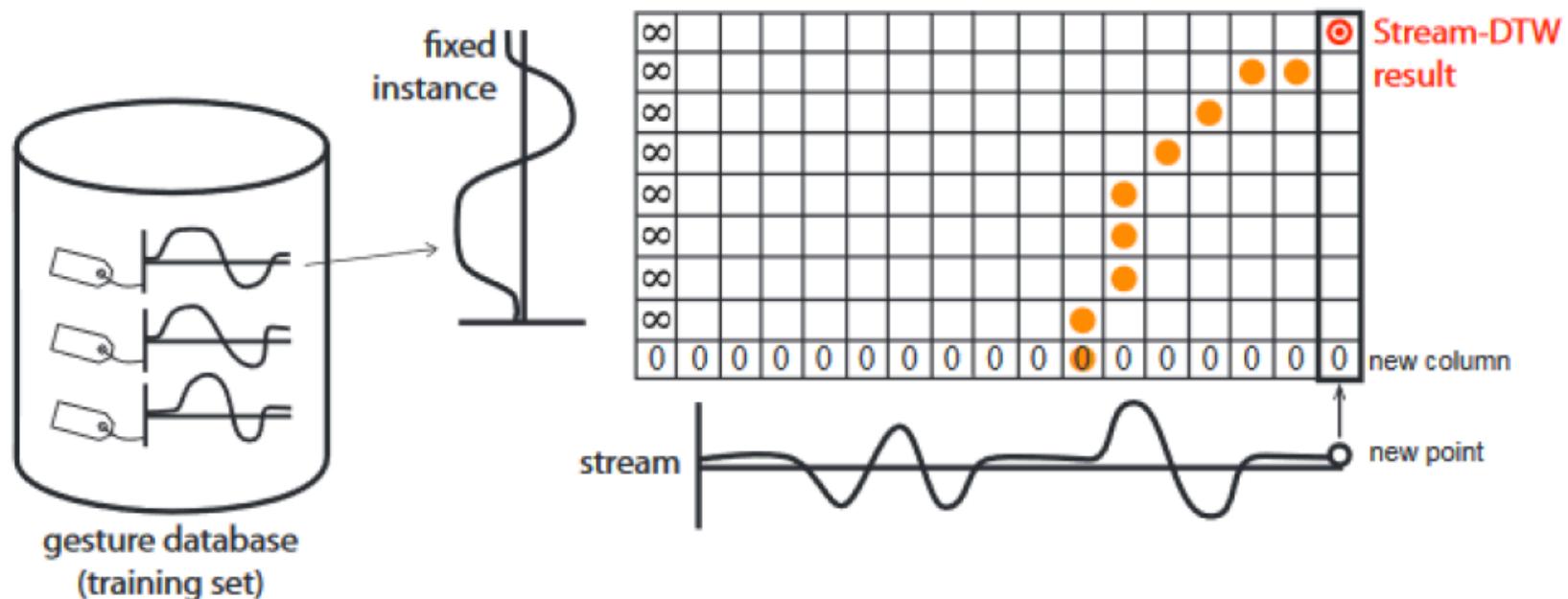
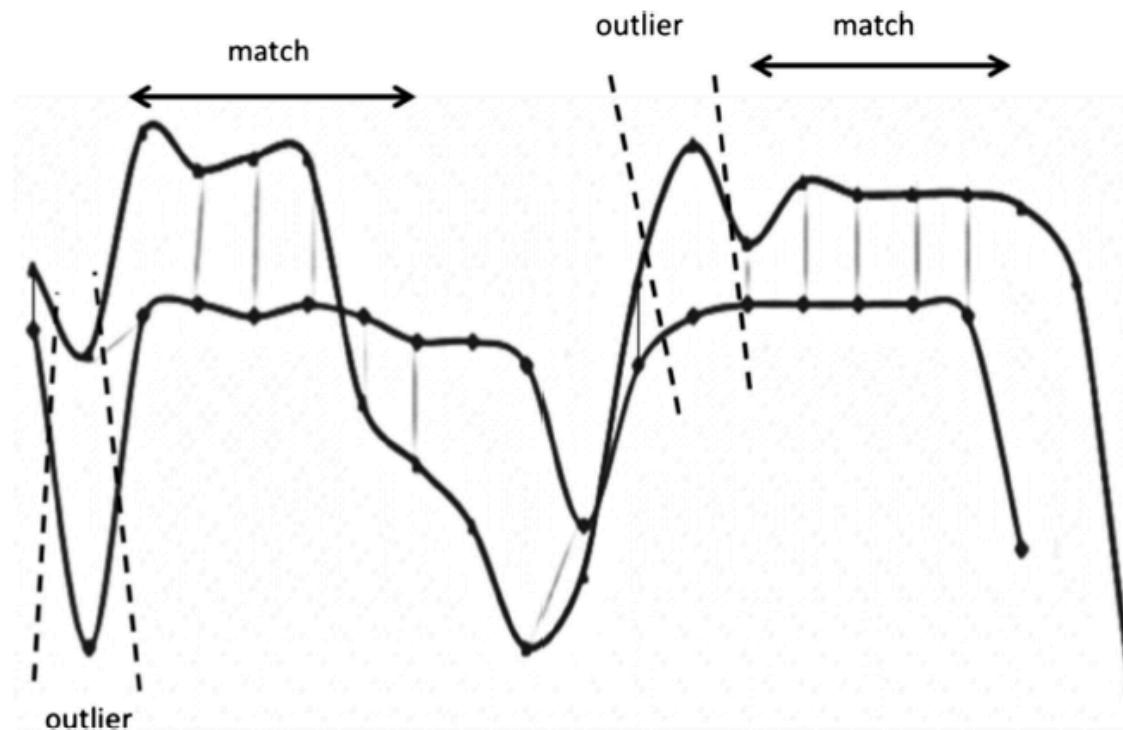


FIGURE 2.19 – Illustration d'une matrice de coût mise à jour de manière incrémentale suivant la technique déployée dans Stream-DTW, proposée dans [Dup17].

Sequential approaches

■ LCSS

- Autres métriques de similarité : LCSS: plus longue sous-séquence commune aux séquences à comparer



Classification of segmented actions

■ Statistical approaches

- k-NN (k-nearest neighbors)
- SVM (with different kernels)
- Multilayer perceptron
- Decision tree

Statistical approaches

■ Hand-crafted features

- Physical attributes and constraints
- Properties and correlations of the data
- Geometric and kinematic features

■ Commonly used features

- Positions of the skeletal joints
- Orientations of the joints
- Distances between joints
- Curvature of the joint trajectories
- Presence of symmetries in the skeletal
- Interpretable metric calculated from the skeletal data

Statistical approaches

■ Représentations brute

- Principe : concaténer, dans un seul grand vecteur, les données brutes du squelette issues de chaque frame.
(coordonnées cartésiennes 3D, angles articulaires, relations géométriques).
- Exemple 1 : *Eigen-Joints* [Yang and Tian 2014] :
 - dimension du vecteur de descripteurs : 2970
 - Information temporelle : différence entre chaque frame et la frame initiale
 - PCA : réduit le vecteur à 128 descripteurs
 - Classifieur : K-NN – appartenance de l'échantillon de test suivant :

$$C^* = \arg \min_C \sum_{i=1}^M \| d_i - NN_C(d_i) \|^2$$

Statistical approaches

■ **Représentations brute**

- Exemple 2 : [Patsadu et al., JCSSE, 2012 – Human gesture recognition using kinect data]
 - Normalisation des coordonnées cartésiennes de 20 articulations
 - Information temporelle : absente
 - Classifieur :
 - Perceptron multicouches (MLP) -> meilleures performances
 - Machines à vecteurs de support (SVM)
 - Arbres de décision

Statistical approaches

■ Descripteurs haut niveau

- Cov3DJ [Hussein et al., IJCAI 2013 – Human action recognition using a temporal hierarchy of covariance descriptors on 3d joint locations]
- Principe : représentation basée sur la matrice de covariance des positions cartésiennes des articulations dans une séquence de frames.
- Si le corps est représenté par K joints et que l'action est représentée sur T frames, on a :

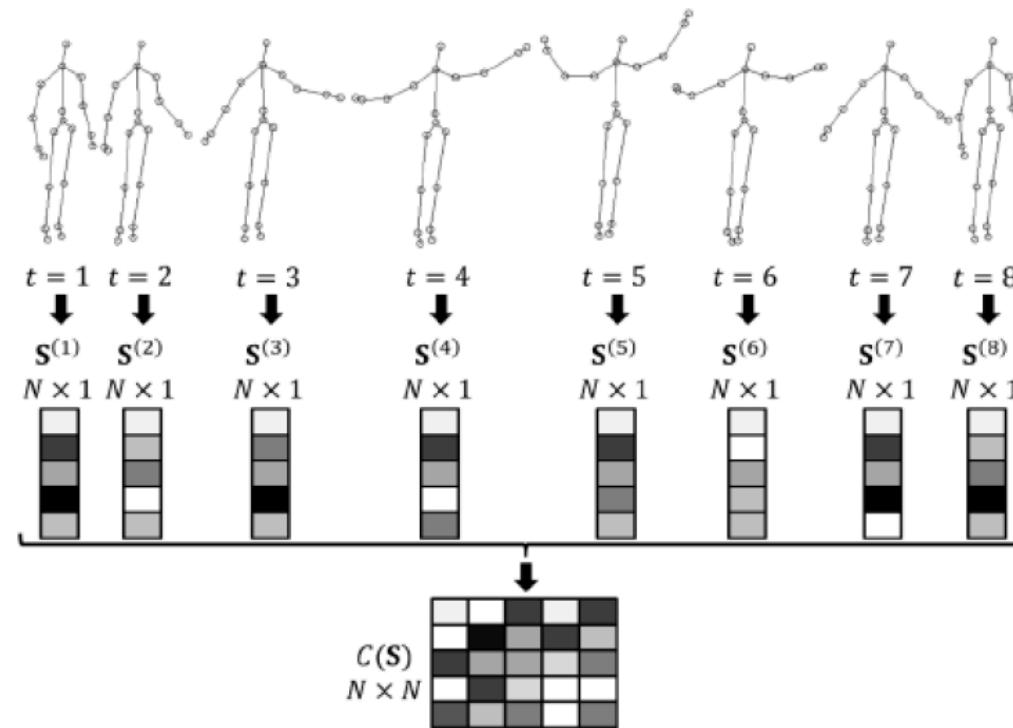
$S = [x_1, \dots, x_K, y_1, \dots, y_K, z_1, \dots, z_K]^T$: vecteur de toutes les positions

- Descripteur de covariance :

$$C(S) = \frac{1}{T-1} \sum_{t=1}^T (S^{(t)} - \bar{S})(S^{(t)} - \bar{S})'$$

Statistical approaches

■ Descripteurs haut niveau

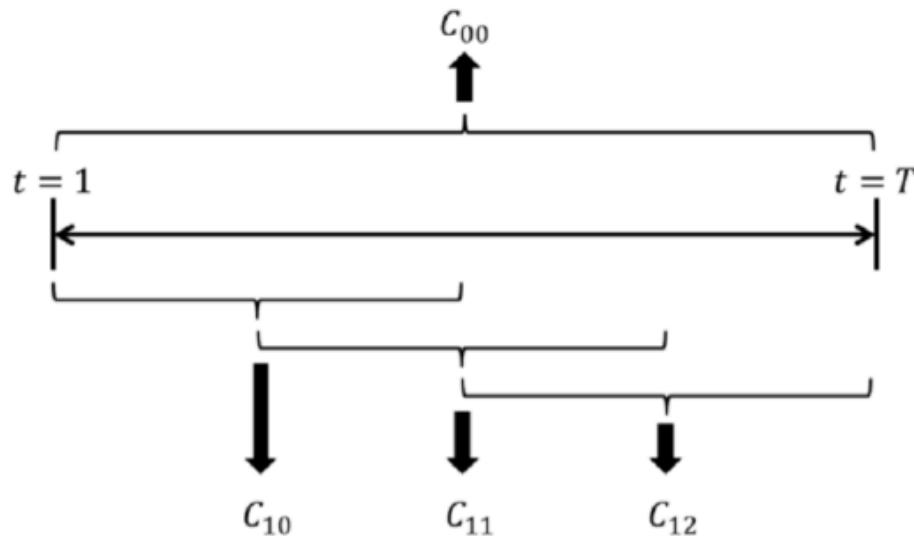


- Pour 20 articulations, $N = 3 \times 20 = 60$
- Le triangle supérieur de la matrice de covariance comporte $N.(N+1)/2 = 1830$ éléments (dimension du descripteur)

Statistical approaches

■ Descripteurs haut niveau

- Encodage de l’information temporelle : calculs suivant une hiérarchie temporelle à plusieurs niveaux :



- Matrice de cov. Au ième niveau : couvre $T / 2^i$ frames
avec T : longueur de la séquence

Statistical approaches

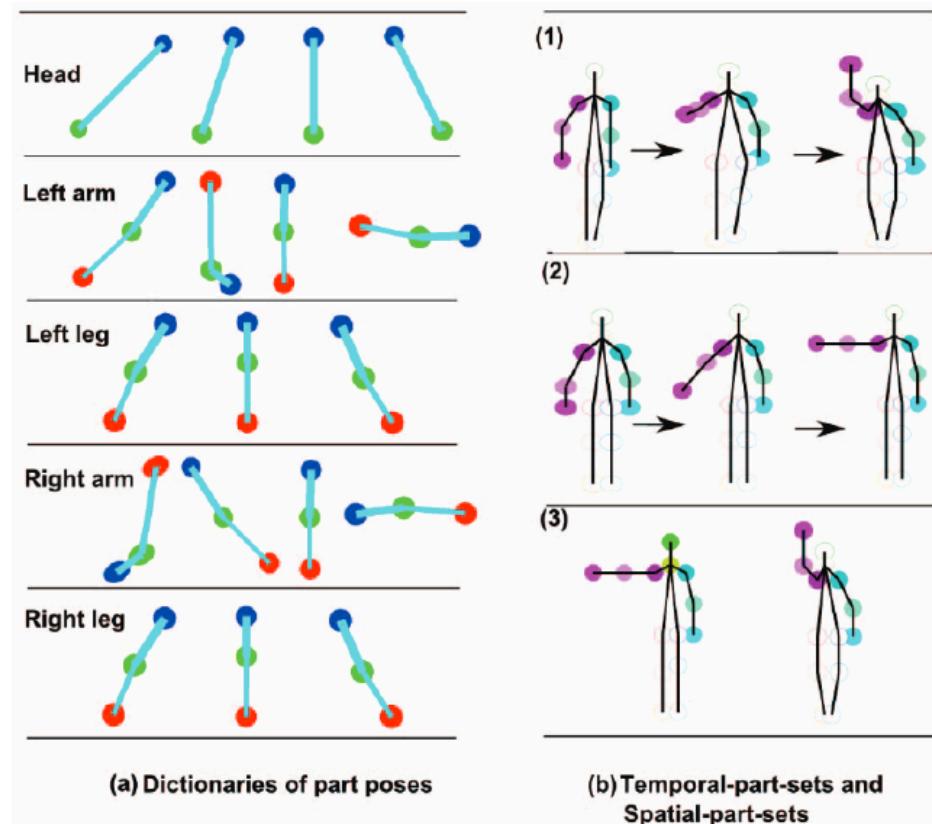
■ Dictionnaire de mots

- Représentation à base de sac de mots : on projète chaque squelette (pose) en un seul code (mot) en utilisant un dictionnaires contenant tous les codes possibles.
- Exemple : on scinde le squelette en 5 parties (tête, bras gauche, bras droit, jambe gauche, jambe droite) et on construit 5 dictionnaires de poses pour chacune des parties et pour chaque classe d'actions.
- k-means : utilisé pour former les 5 dictionnaires de poses simples pour une classe d'actions

Statistical approaches

■ Dictionnaire de mots

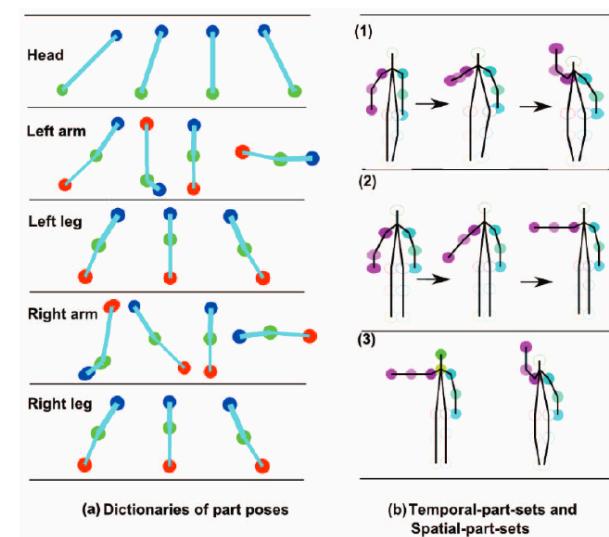
- Classes regroupées 2 à 2 pour construire 5 autres dictionnaires composés de n-uplets de ($n=2, n=3, \dots$) poses



Statistical approaches

■ Dictionnaire de mots

- Représentation finale d'une action : résultat de la concaténation des projections de l'action sur les dictionnaires spatiaux et temporels des 5 parties du corps
- Classification : plusieurs SVM binaires en mode un-contre-un
--> la classe finale est celle ayant été le plus prédite par les classifieurs.



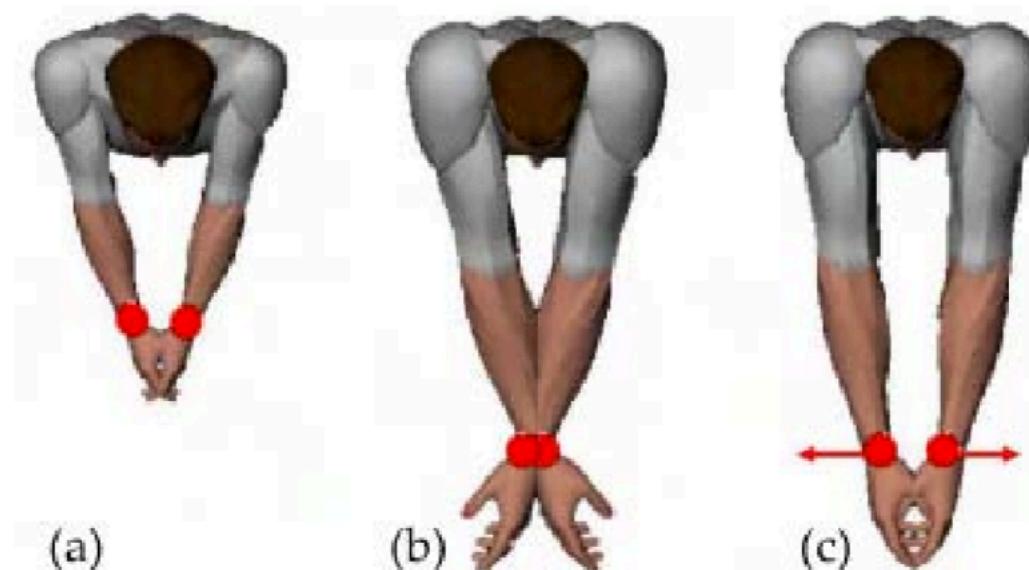
Statistical approaches for recognition of segmented actions

- **Exemple:** [PhD, Y. Boulahia, Rennes, juillet 2018]
 - Actions : large panel d'applications
 - Données squelettiques uniquement (et non RGB)
 - Transfert du savoir-faire de la modélisation de tracés manuscrits 2D vers la reconnaissance d'actions 3D

Statistical approaches for recognition of segmented actions

■ Défis à relever

- Variabilité morphologique des sujets



Statistical approaches for recognition of segmented actions

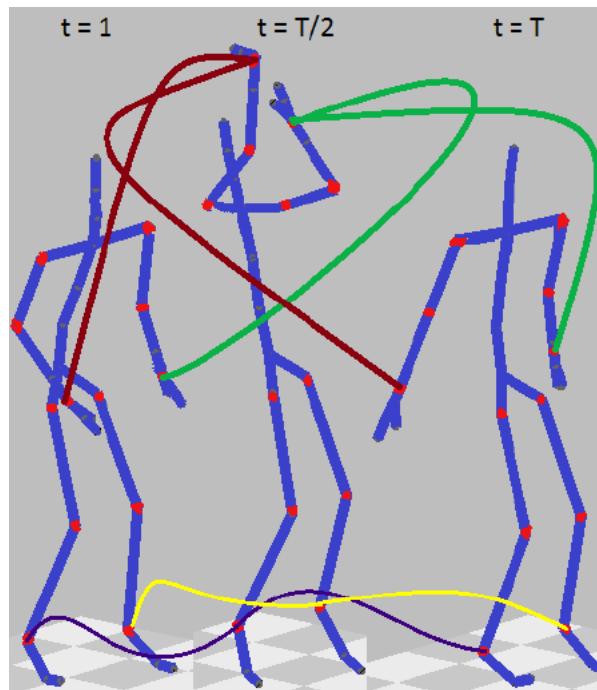
■ Défis à relever

- Représentation des corrélations spatiales entre les différentes trajectoires des articulations
 - Concevoir des descripteurs qui permettent d'extraire suffisamment d'informations sur chacune des trajectoires, mais aussi leur inter-dépendance spatiale (corrélation spatiale)
 - Problème : plus on capture d'information, plus le nombre de descripteurs augmente !

Statistical approaches for recognition of segmented actions

■ Défis à relever

- Certaines actions sont caractérisées par la connaissance simultanée des informations de plusieurs articulations



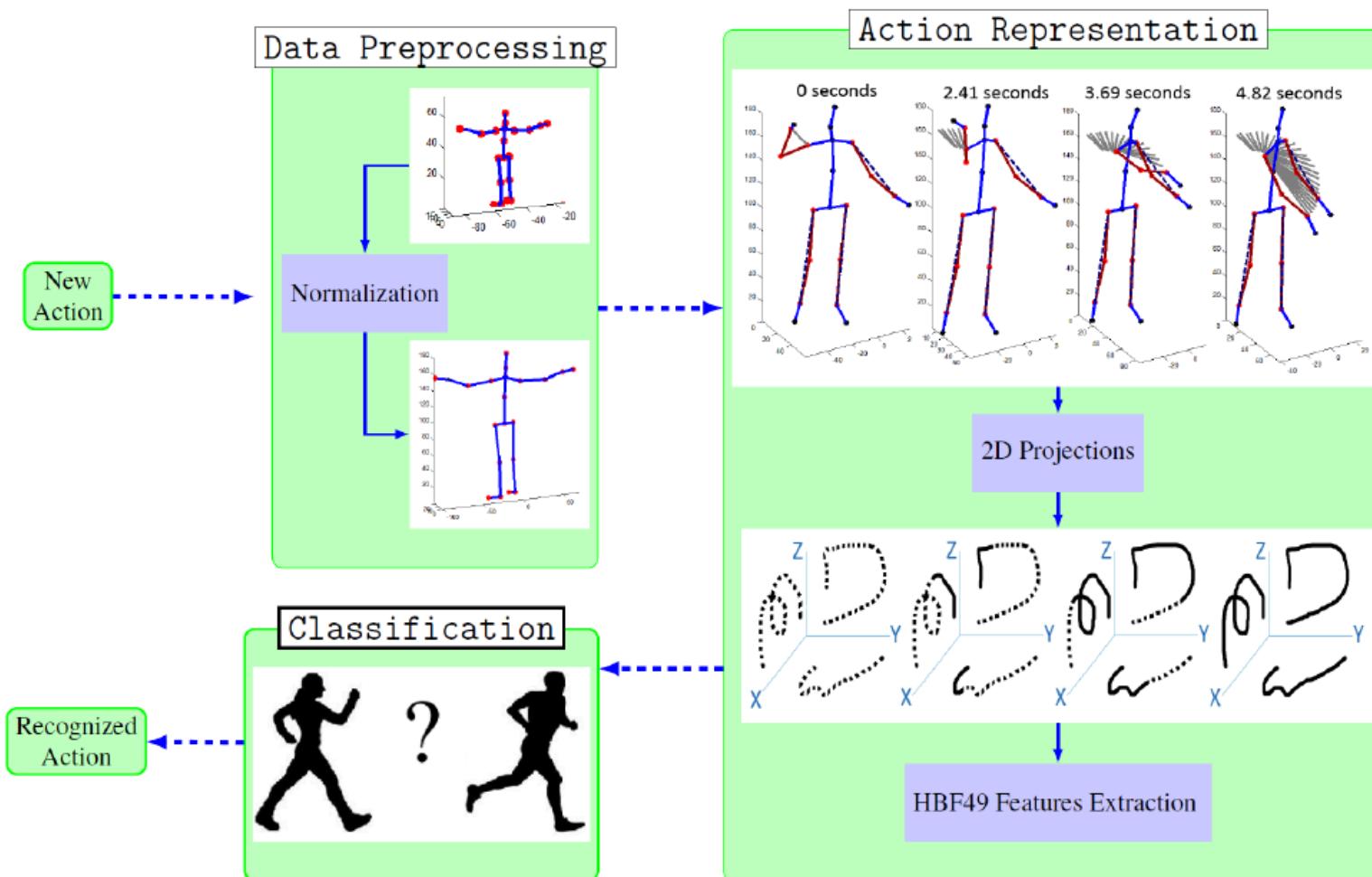
Statistical approaches for recognition of segmented actions

■ Défis à relever

- Comment représenter les dépendances temporelle propres à une action, et portées par plusieurs articulations ?
- HMM, LCSS ?

Statistical approaches for recognition of segmented actions

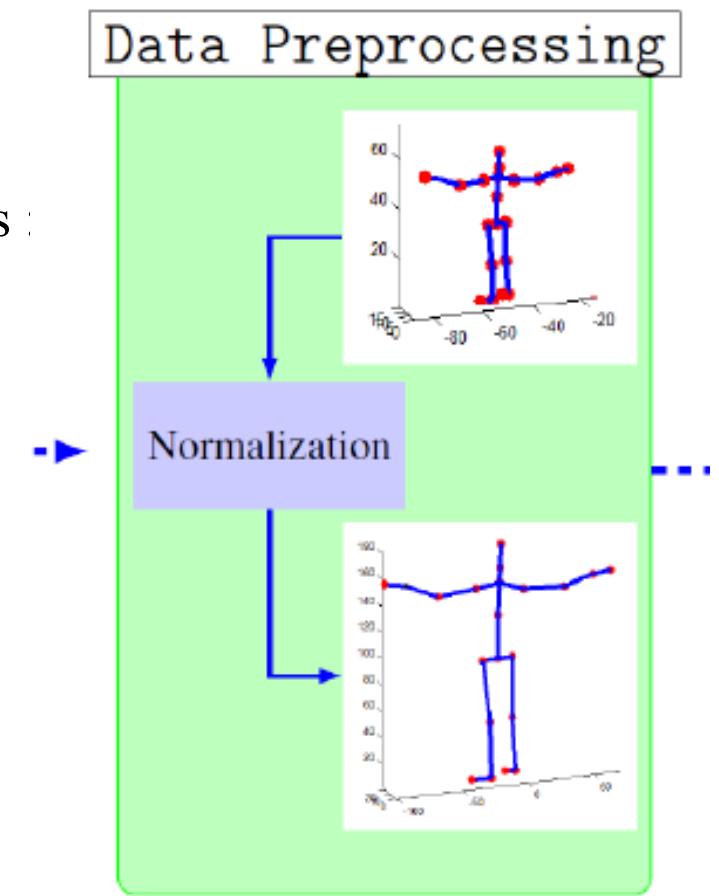
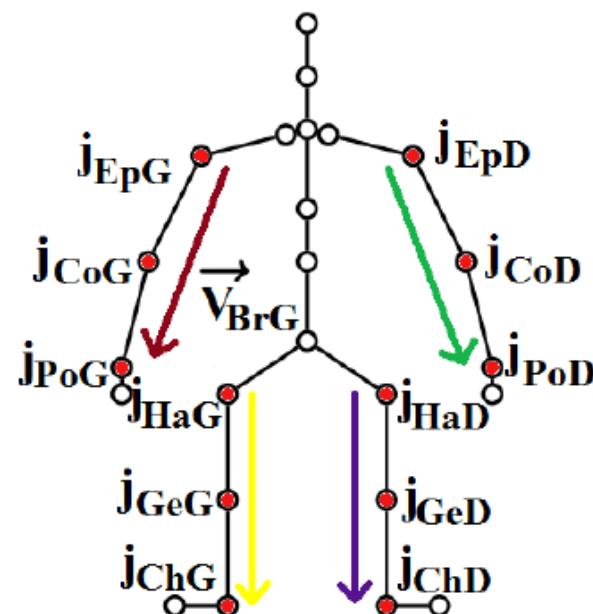
■ Approche 3DMM (Boulahia)



Statistical approaches for recognition of segmented actions

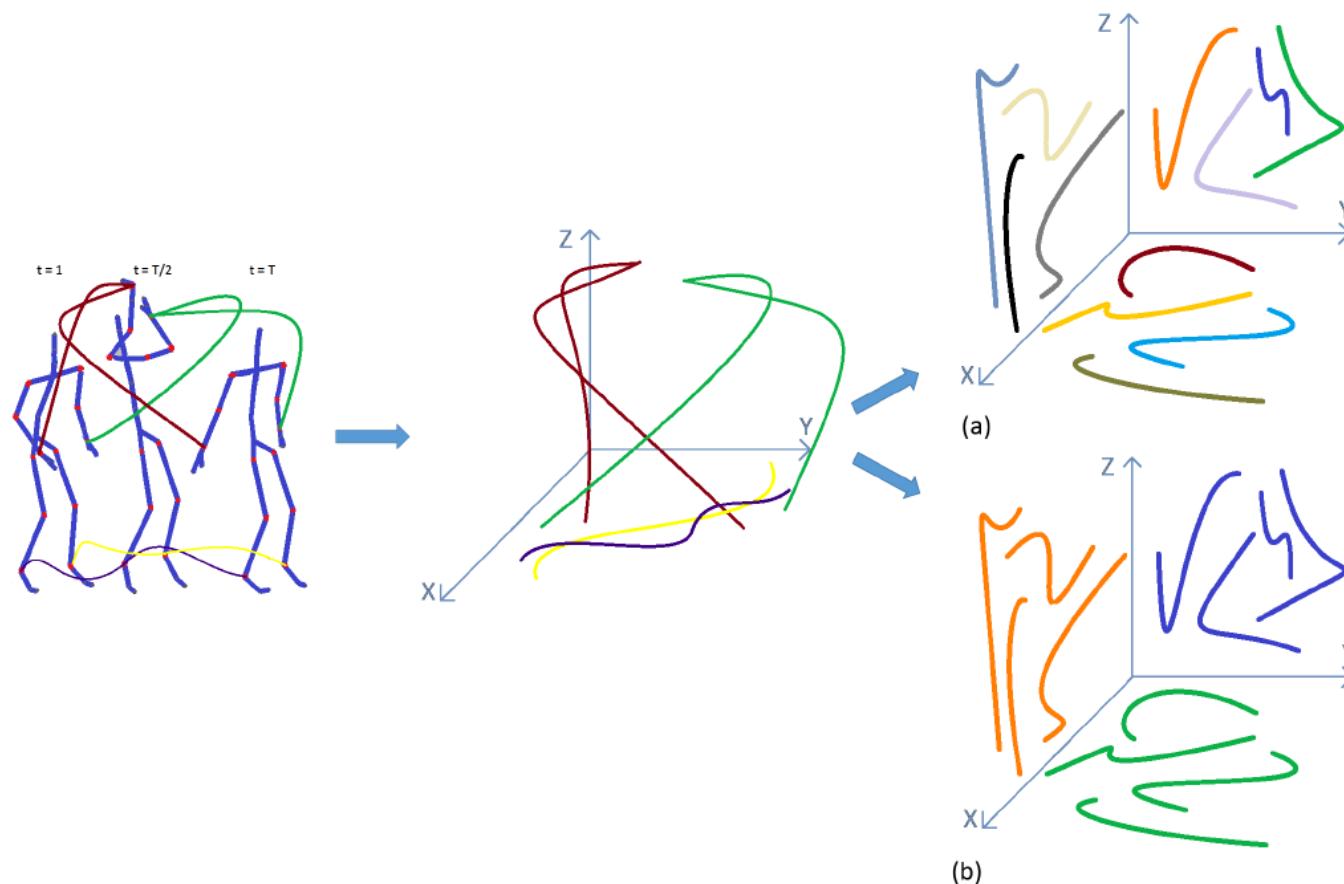
■ Approche 3DMM : prétraitement anamorphologique

- Normalisation des dimensions du squelette
- Articulations sélectionnées et vecteurs anamorphologiques associés :

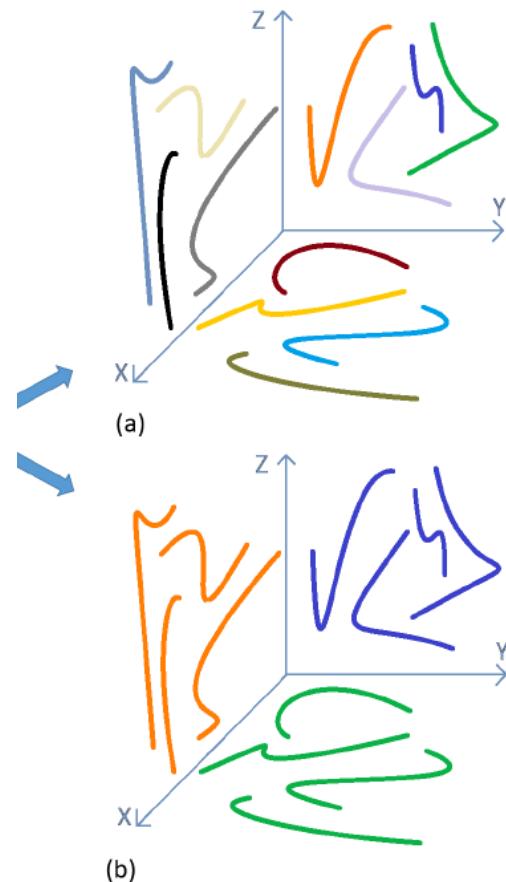


Statistical approaches for recognition of segmented actions

■ Approche 3DMM : approche multistrokes



Statistical approaches for recognition of segmented actions

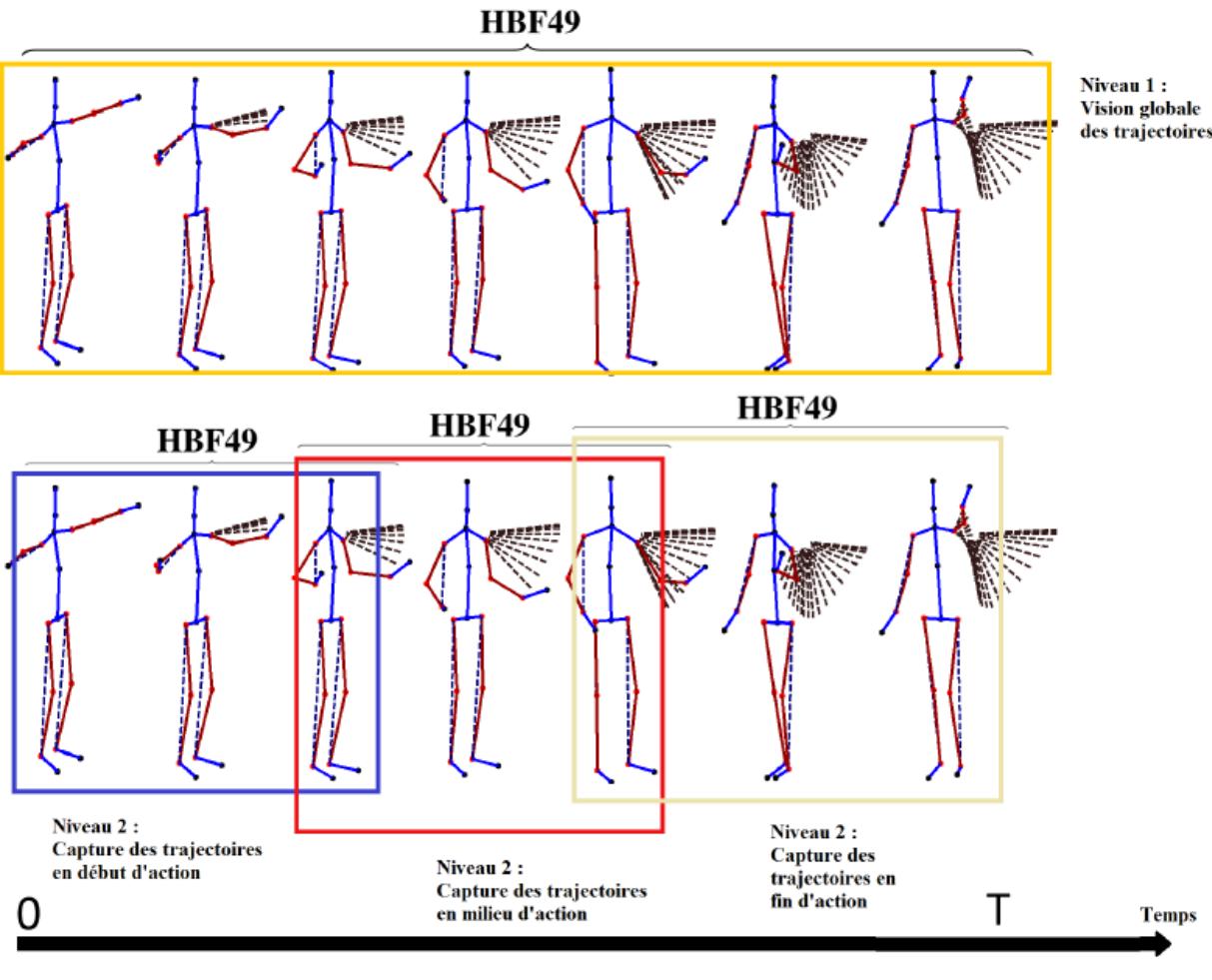


■ Approche 3DMM : approche multistrokes (pb corrélation)

- Projectoire des trajectoires sur les 3 plans
- Version monostroke : chaque trajectoire est considérée indépendamment des autres : tracé 2D composé d'un seul trait
- Version multistrokes: tracé composé de toutes les trajectoires projetées appartenant au même plan
- Exemple : 4 pseudo-symboles monostrokes (a) ou un pseudo-symbole multistrokes (b) -> choixde (b) : représentation réduite qui tien compte de la corrélation spatiale entre les strokes.
- Pour une action : $49 \text{ descripteurs} * 3 = 147$

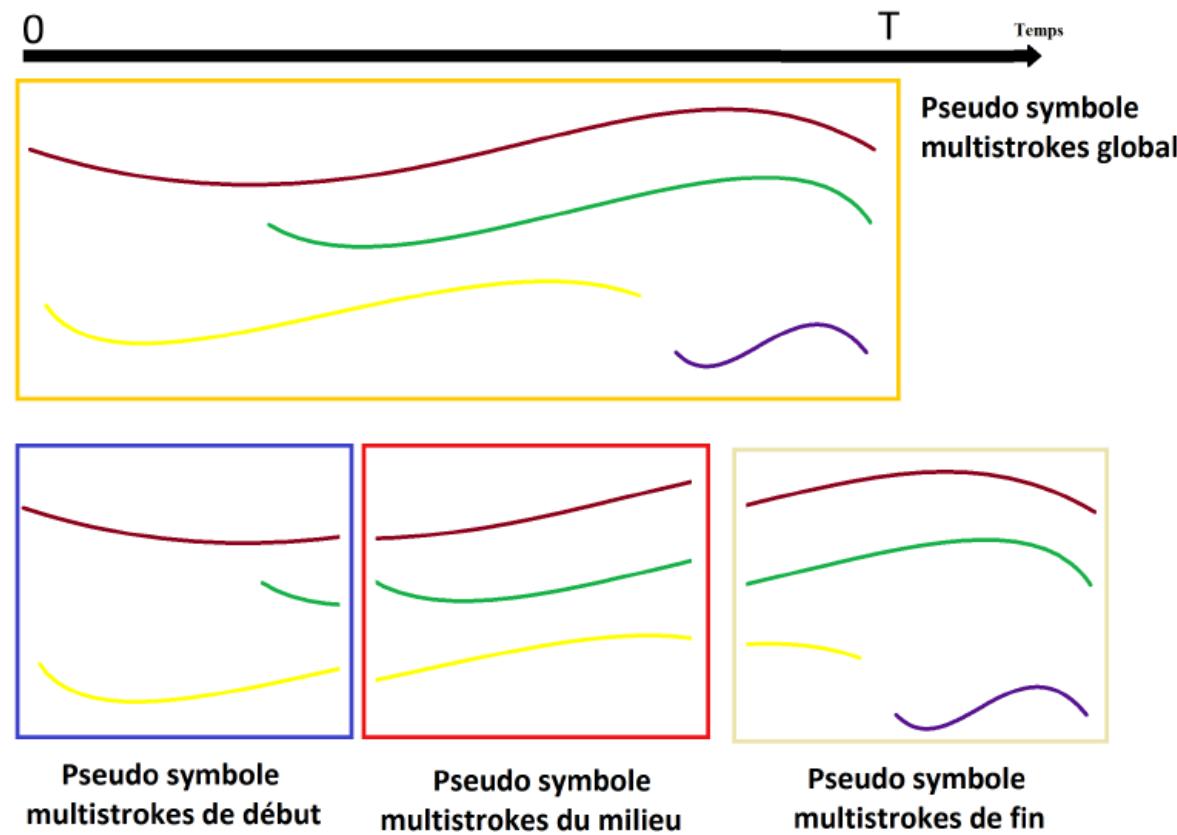
Statistical approaches for recognition of segmented actions

■ Approche 3DMM : hiérarchie temporelle



Statistical approaches for recognition of segmented actions

■ Approche 3DMM : hiérarchie temporelle

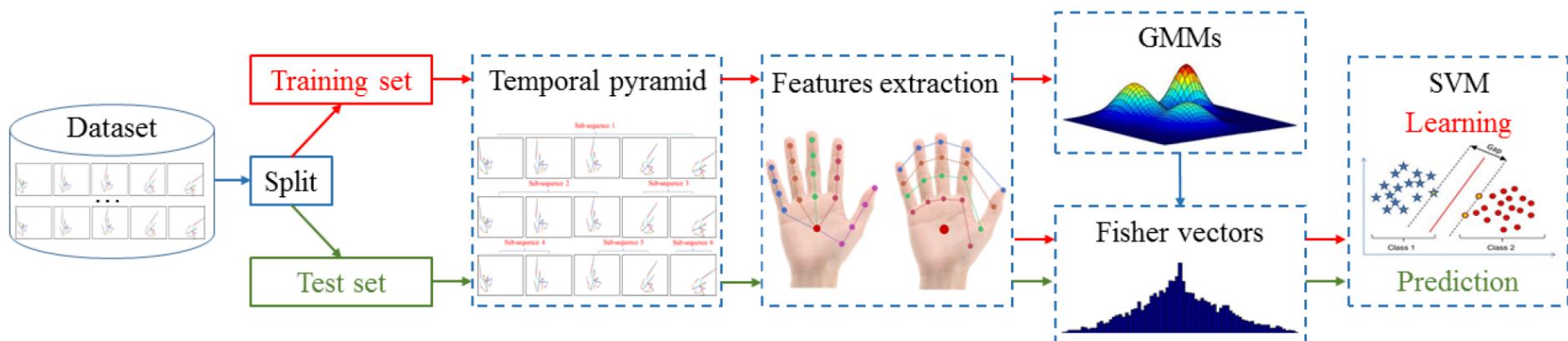


- $147 * 3 = 441$ nouveaux descripteurs

Statistical approaches for recognition of segmented actions

■ [PHD Quentin De Smedt, Lille, January 2018]

- Sequences are divided into overlapping subsequences using a temporal pyramid strategy
- 3 sets of features: directions, rotations, and handshape descriptors
- Fisher kernel used to transform these sets into a statistical representation
- SVM for hand-gesture classification



Classification of segmented actions : quantité de données

■ Approches avec peu de données

- Approches statistiques

■ Approches avec de grandes quantités de données

- Deep learning

Classification of segmented actions

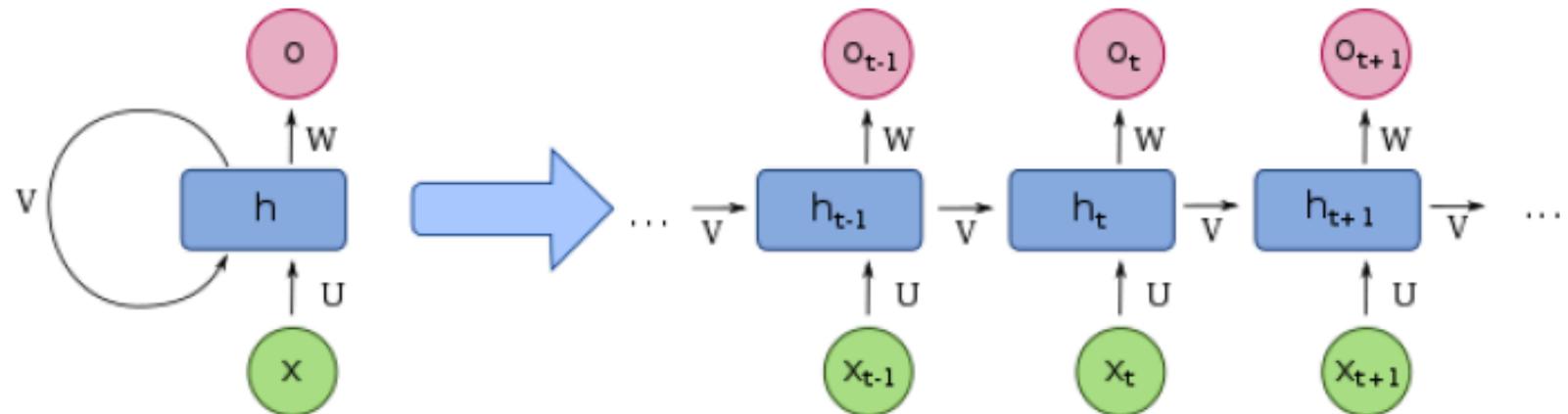
■ Deep learning

- Recurrent Neural Network (RNN)
- Long-Short Term Memory Neural Network (LSTM)
- Convolutional Neural Network (CNN)

Deep learning

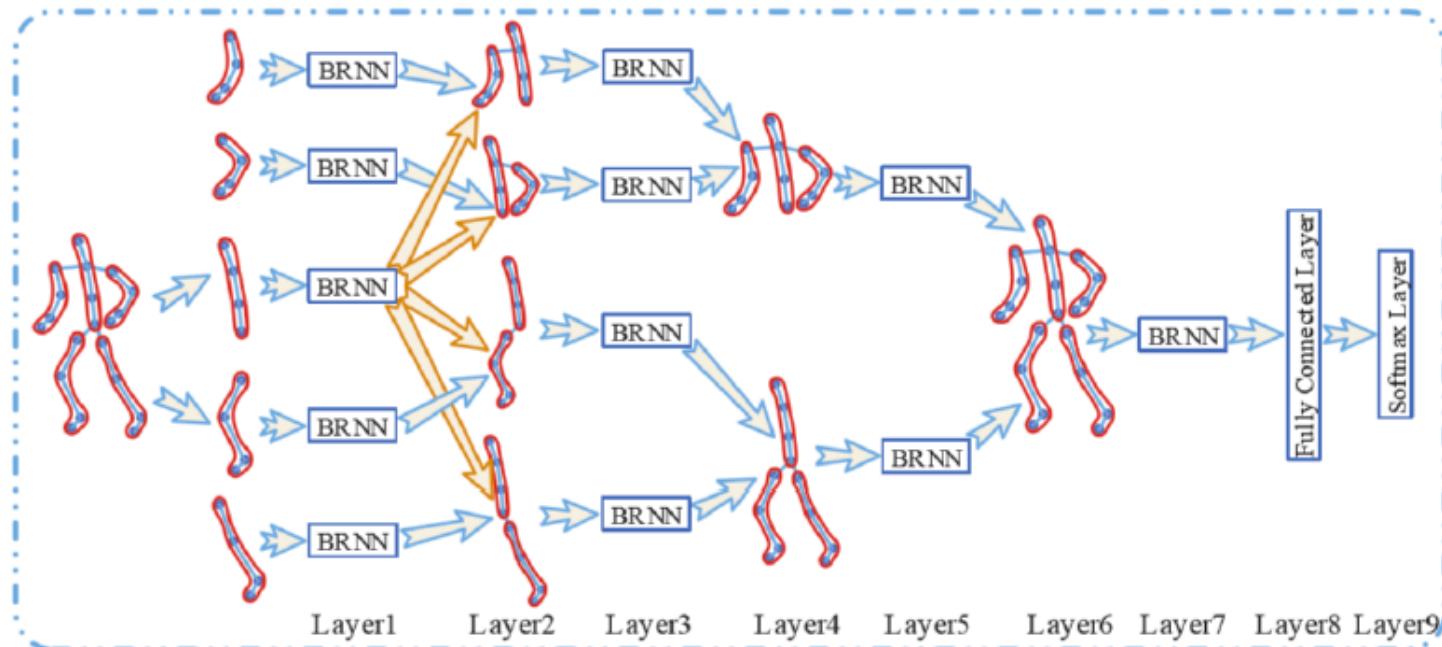
■ Recurrent Neural Network (RNN)

- Définition : réseau de neurones artificiel présentant des connexions récurrentes.
- Basic RNNs : les noeuds représentent des neurones (fonction h) qui sont organisés en couches successives
- Chaque noeud à un niveau donné est connecté de manière directe avec le noeud du niveau suivant.



Recurrent Neural Network (RNN)

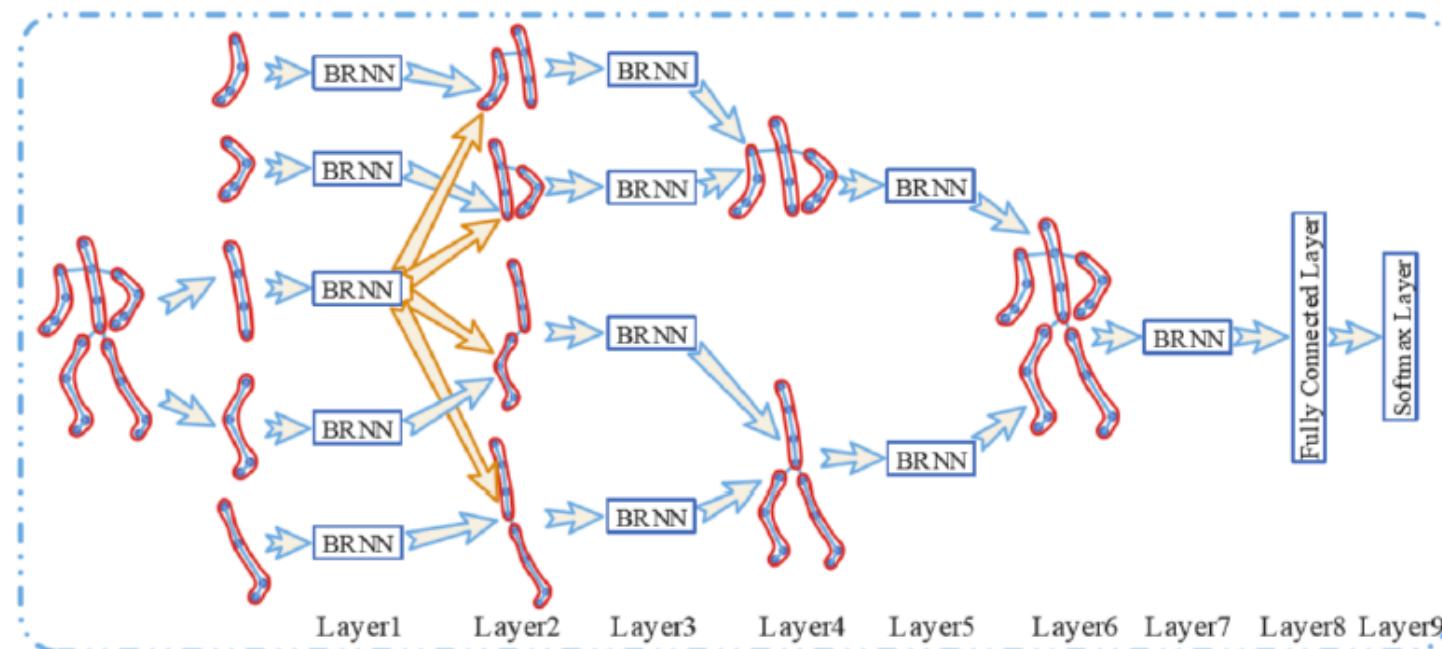
- Skeleton action recognition :
 - Approche à plusieurs couches : squelette subdivisé en plusieurs parties
 - 1ère couche: les données de chacune des parties servent à entraîner un réseaux de neurones récurrents bidirectionnel (BRNN).
 - 2ème couche : représentation issue du BRNN relatif au tronc avec celle des 4 autres BRNN -> obtient 4 nouvelles représentations



[Du, Wang, Wang, IEEE conf. on Computer Vision and Pattern Recognition, 2015]

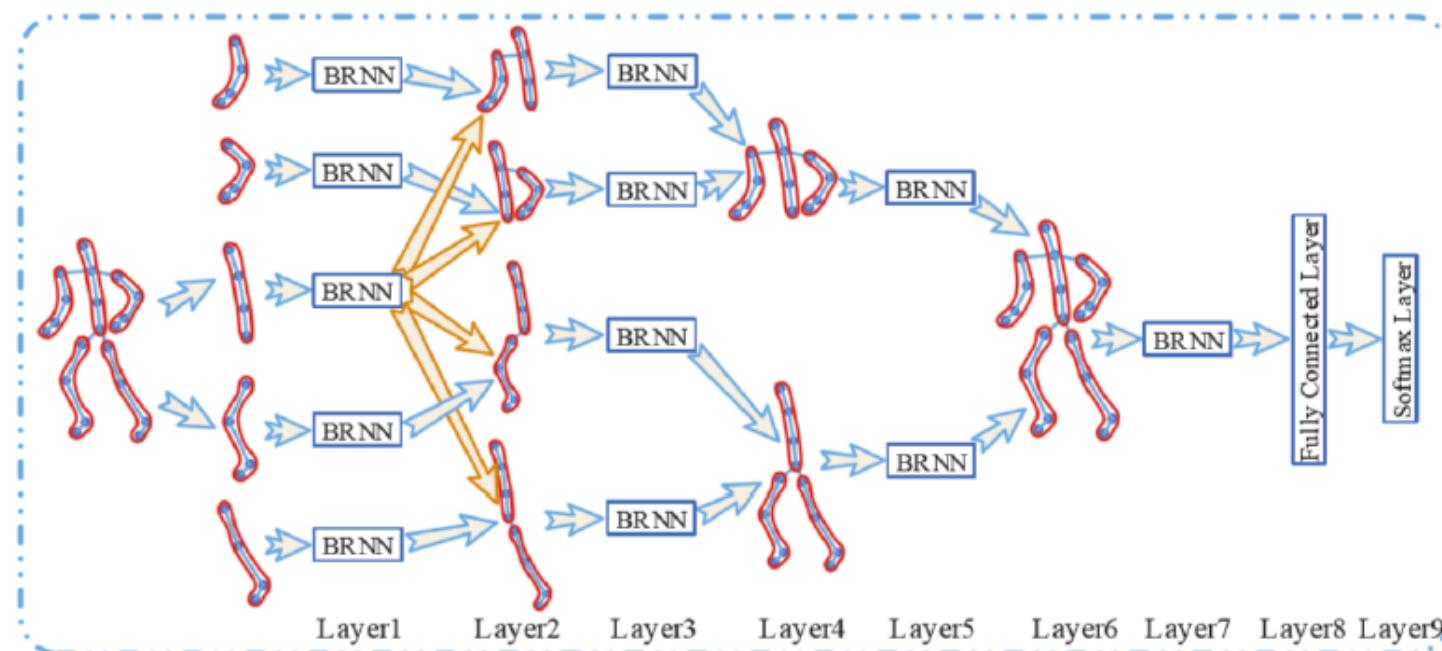
Recurrent Neural Network (RNN)

- Skeleton action recognition
 - 3ème couche : les 4 représentations résultantes sont fournies séparément à 4 BRNN
 - Procédure fusion/apprentissage : réitérés pour les couches 4, 5, 6 et 7
 - Représentation finale du squelette reconstitué



Recurrent Neural Network (RNN)

- Skeleton action recognition
 - Interprétation : les BRNN empilés peuvent être considérés comme extracteurs des descripteurs spatiaux et temporels de la séquence d'action
 - 1 Couche complètement connectée
 - + 1 Couche Softmax rajoutées pour classifier l'action.



RNN-Vanishing Gradient Problem

- Technically: learning long-term dependencies with gradient descent and backpropagation is difficult
 - NN's weights: update proportional to the partial derivative of the error function in each iteration of training, and consequently:
 - sometimes the gradient will be small, thus preventing the weight from changing its value
 - can potentially stop the NN from further training
 - Example: with activations functions *htan*, gradients are in the range of (0,1)
 - with backpropagation, the gradient is computed in an n -layer network: the gradient (error) decreases exponentially with n , and the training is very slow
- This problem affects:
 - Multi-layers feedforward networks
 - Recurrent networks

RNN-Vanishing Gradient Problem

- Intuition: RNN unable to store information about past inputs for very long
 - Diminish their ability to model long-range structure
 - Make them prone to instability when generating sequences
- If the network's predictions are only based on the last few inputs, and these inputs are themselves predicted by the network
 - It won't recover from past mistakes

Generating sequences with RNN

- RNN: Rich class of dynamic models that can be used to generate sequences in domains such as music, text, motion capture data
- Assuming the predictions are probabilistic, novel sequences can be generated from a trained network by iteratively sampling from the network's output distribution, then feeding in the sample as input at the next step.
 - It makes the network treat its invention as if they were real, much like a person dreaming
 - The stochasticity injected by picking samples induces a distribution over sequences
 - The distribution is conditional, since its predictive distribution depends on the previous inputs

Generating sequences with RNN

- Like other NN, RNN use their internal representation to perform a high-dimensional interpolation between training examples.
 - It results by the capability of the RNN to synthesize and reconstitute the training data in a complex way, and rarely generate the same thing twice.
- BUT:
 - RNN are unable to store information about past inputs for very long, thus diminishing their ability to model long-range structures
 - This amnesia makes them prone to instability when generating sequences: indeed, if the network's predictions are only based on the last few inputs, and these inputs were themselves predicted by the network, it has little opportunity to recover from past mistakes

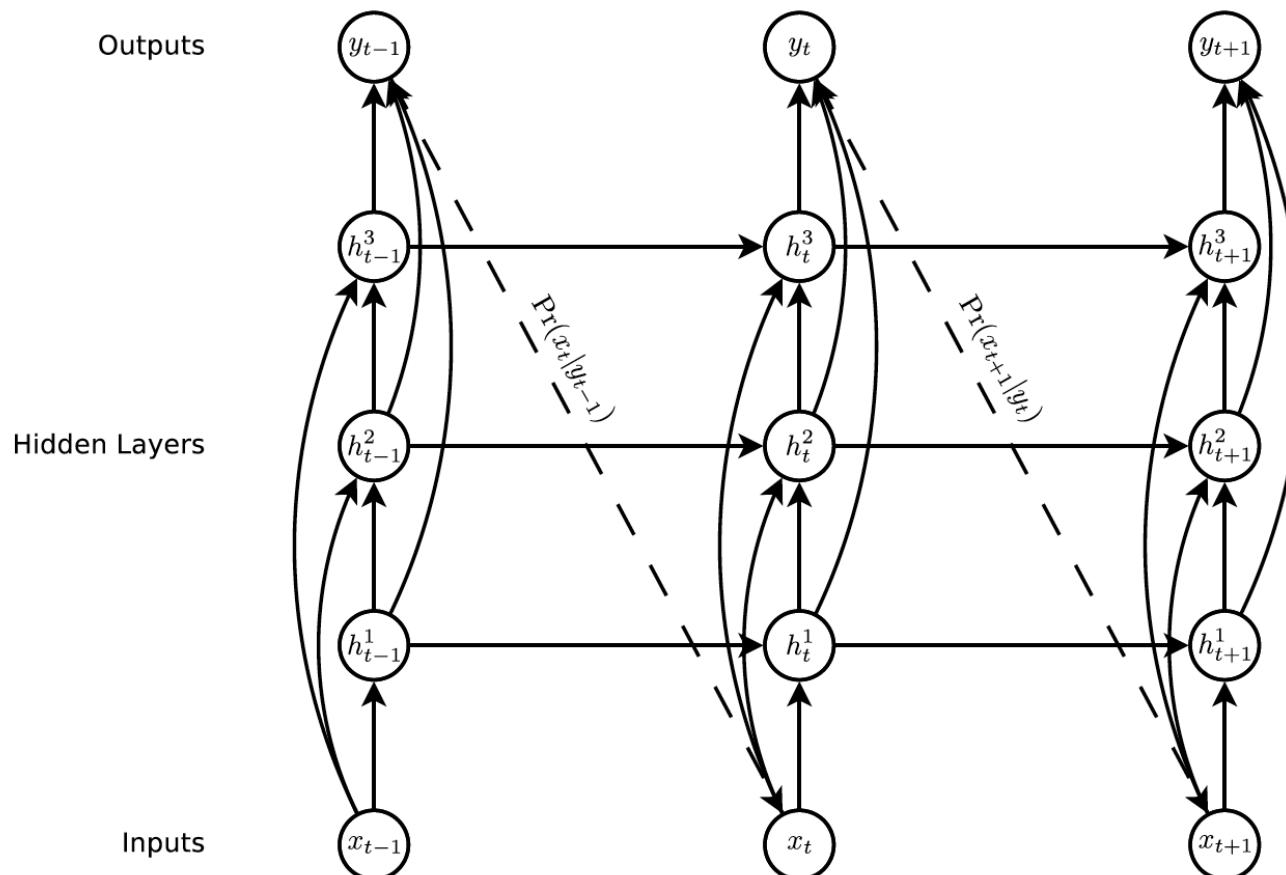
Long Short-Term Memory

■ LSTM:

- LSTM is a deep learning system that avoids the vanishing gradient problem.
- Uses RNN architecture designed to store and access information
- LSTM can use its memory to generate complex, realistic sequences containing long-range structures
- Used in a variety of sequence processing tasks, such as speech or handwriting recognition and generation

Long Short-Term Memory

[Hochreiter & Schmidhuber, 1997]



[A. Graves, 2014]

Long short-term memory (LSTM)

■ Principle

- Input vector sequence $\mathbf{x} = (x_1, \dots, x_T)$ passed through weighted connections
- to a stack of N recurrently connected hidden layers
- to compute first the hidden vector sequences $\mathbf{h^n} = (h_{1^n}, \dots, h_{T^n})$
- and then the output vector sequence $\mathbf{y} = (y_1, \dots, y_T)$
- Each output vector y_t is used to parameterize a predictive distribution $\Pr(x_{t+1} | y_t)$ over the possible next inputs x_{t+1} .

■ Algorithm

- First input x_1 of every input sequence: always null vector
- The network emits a prediction for x_2 with no prior information
- ...
- The network is deep in both space and time
- Skipping connections: from inputs to all hidden layers, and from hidden layers to outputs: makes it easier to train deep networks
- If $N = 1$: single layer next step prediction RNN

Long short-term memory (LSTM)

■ Algorithm

- The output sequence is computed as:

$$\hat{y}_t = b_y + \sum_{n=1}^N W_{h^n y} h_t^n$$
$$y_t = \mathcal{Y}(\hat{y}_t)$$

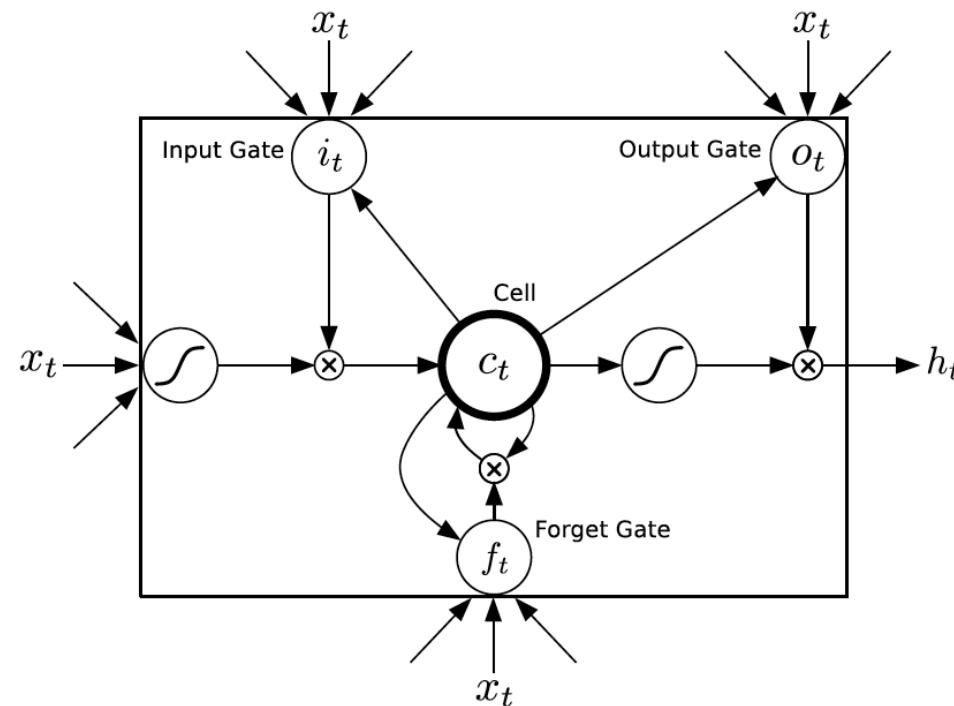
where \mathcal{Y} is the output layer function.

- The complete network therefore defines a function, parameterized by the weighted matrices, from input histories $\mathbf{x}_{1:t}$ to output vectors y_t .

Long short-term memory (LSTM)

■ Hidden-layer functions H

- Sigmoid function
- Memory cells to store the information



Long short-term memory (LSTM)

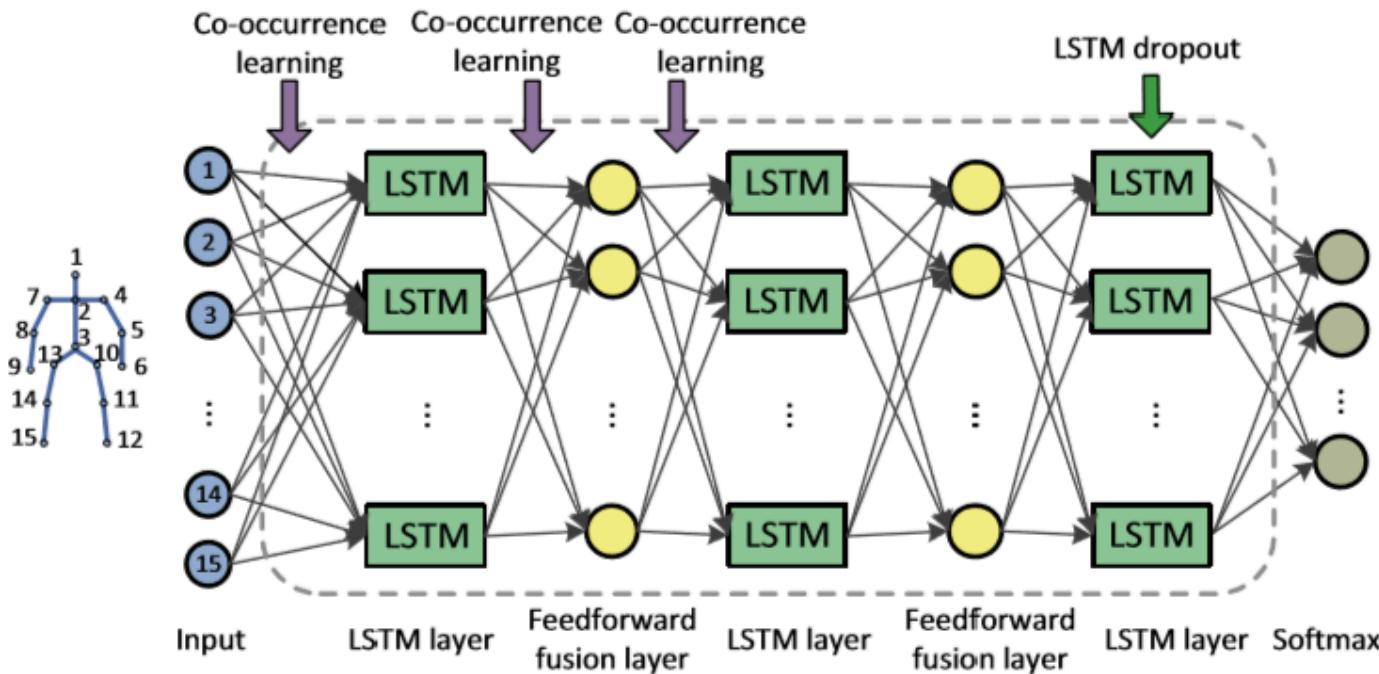
■ LSTM algorithm

- Original LSTM algorithm: custom-designed approximate gradient calculation: the weights are updated after every timestep
- The full gradient can be instead calculated with backpropagation through time
 - Possible problem: the derivatives may become excessively large, leading to numerical problems

Deep learning for action recognition

■ Long short-term memory (LSTM)

- Utilisés pour modéliser la progression temporelle d'une série
- Détermine automatiquement les assemblages les plus pertinents d'articulations au lieu de spécifier à l'avance comment les articulations doivent être groupées



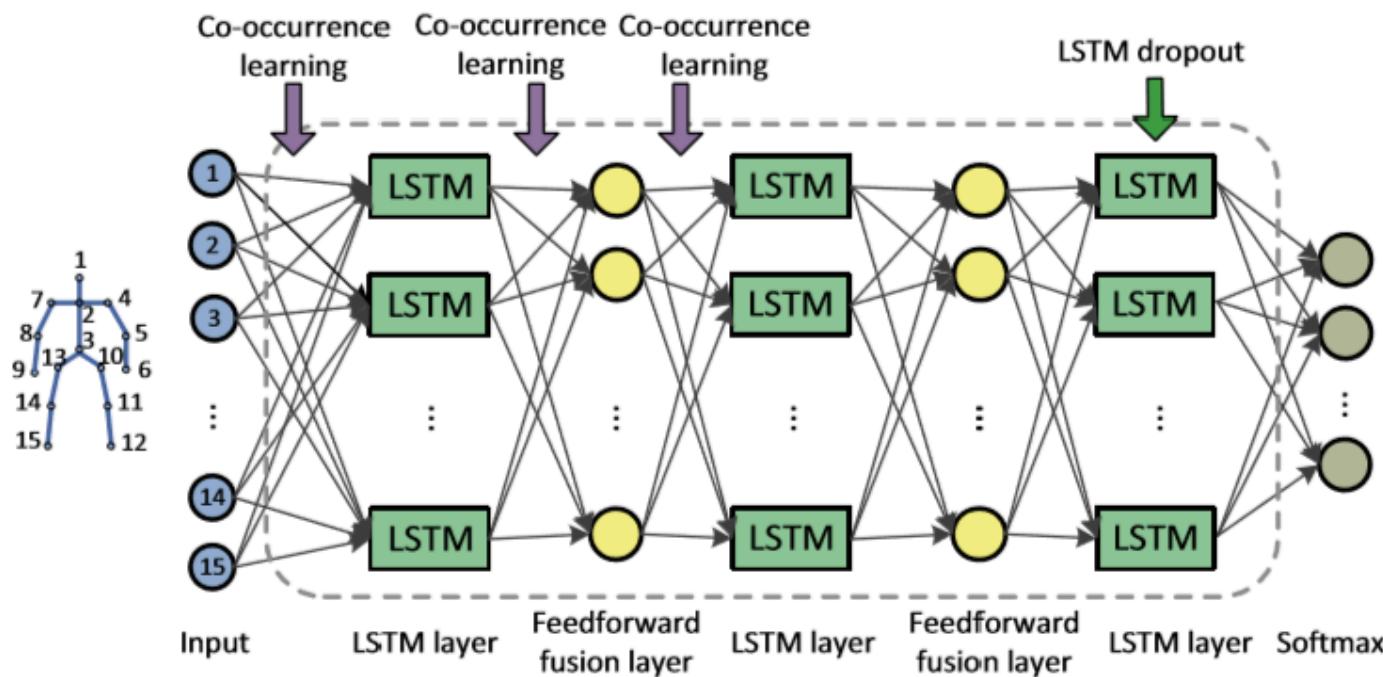
[Zhu, Lan, Xing, Zeng, Li, ‥ AAAI, 2016] – co-occurrence feature learning for skeleton based action recognition using regularized lstm networks

Deep learning for action recognition

■ Long short-term memory (LSTM)

– Architecture :

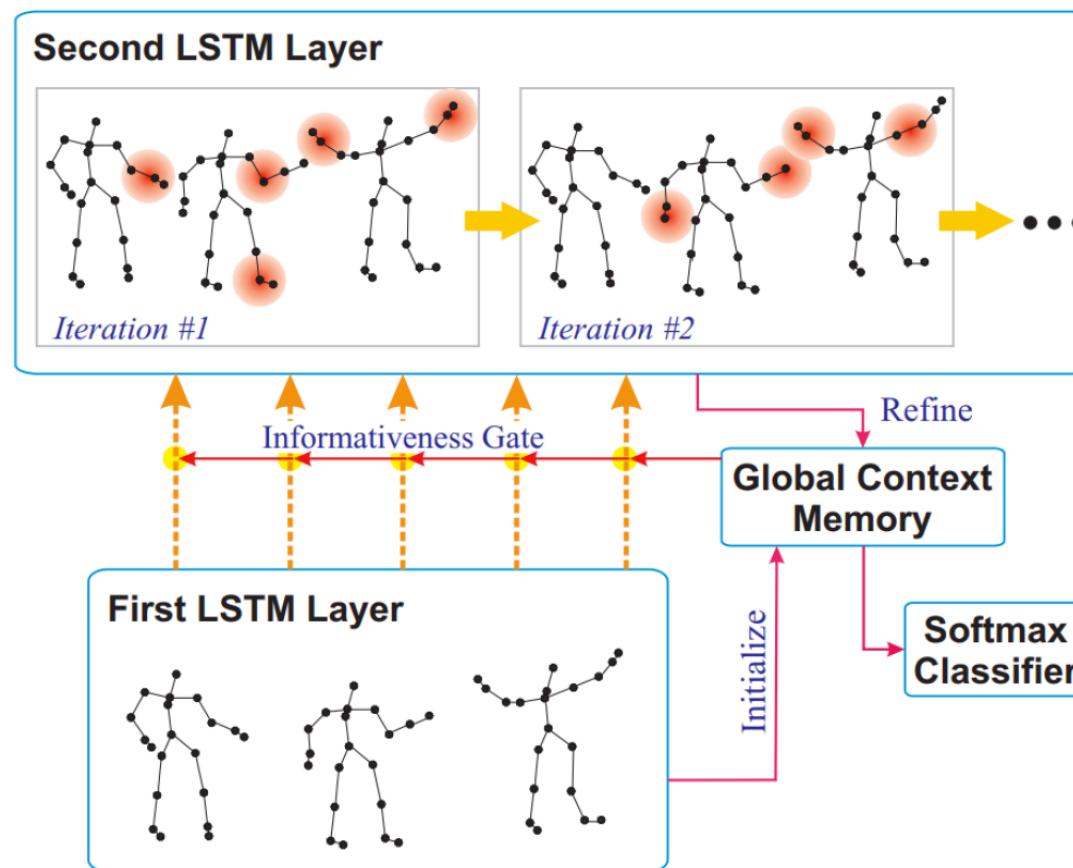
3 couches LSTM, 2 couches de fusion des représentations intermédiaires, une couche Softmax pour la classification



Deep learning for action recognition

■ Long short-term memory (LSTM)

[Liu, Wang, Hu, Duan, Kot, IEEE conf. on Computer Vision and Pattern Recognition (CVPR). 2017]: Global context-aware attention lstm networks for 3d action reco.



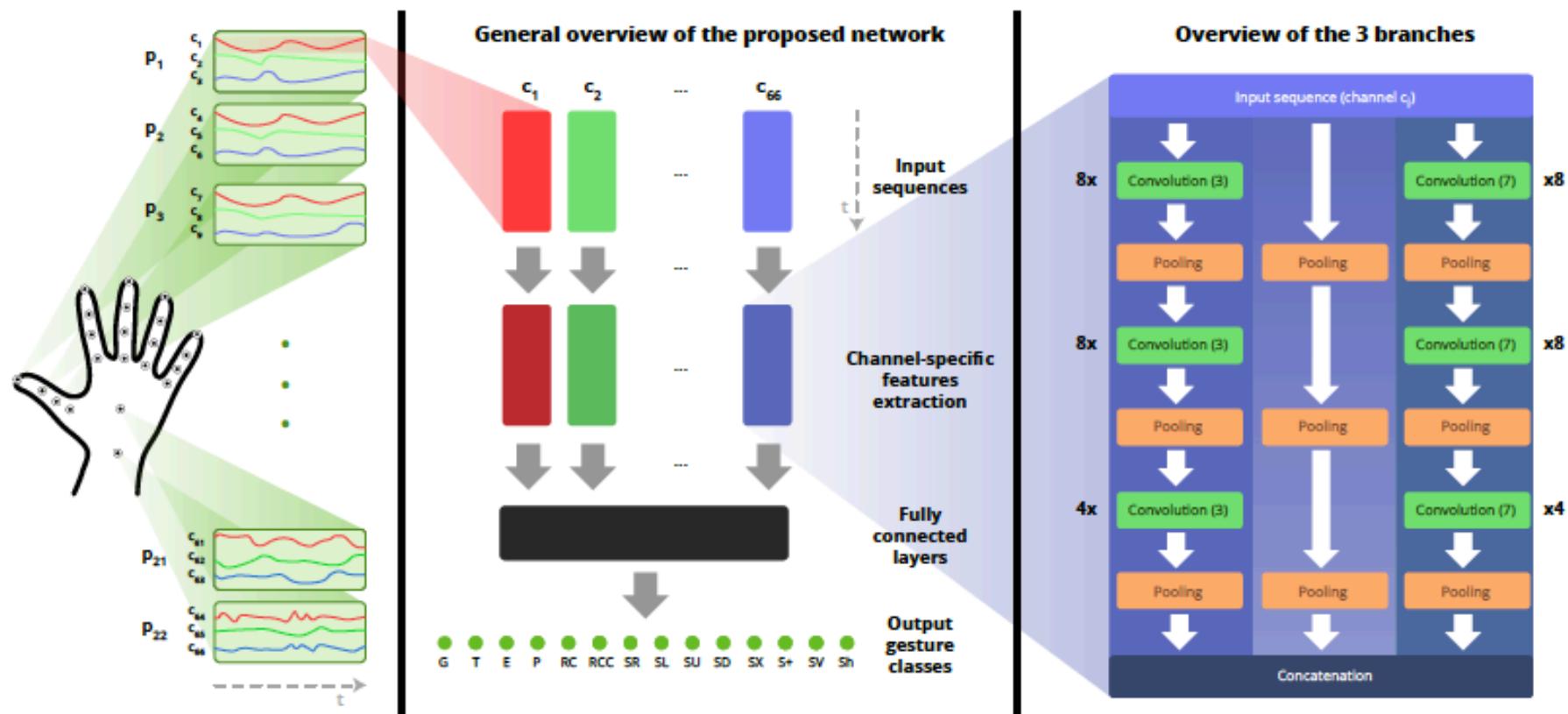
Deep learning and convolution neural networks

■ Convolutional neural networks

- Approche statistique plutôt que séquentielle
- Très hauts scores pour la classification d'images
- Puissants extracteurs de descripteurs spatiaux

Deep learning and convolution neural networks

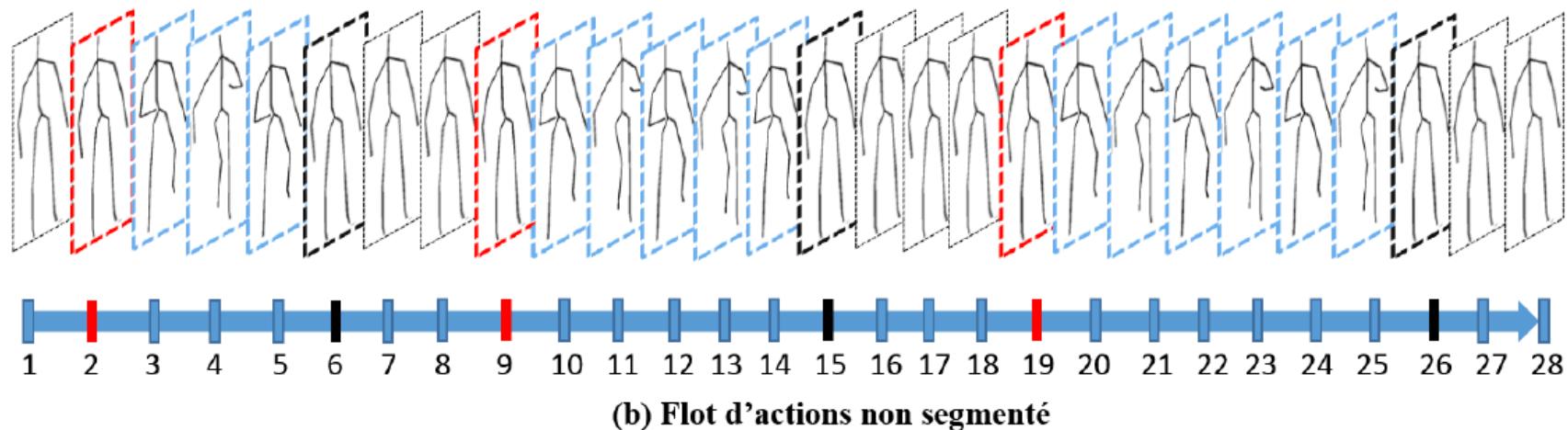
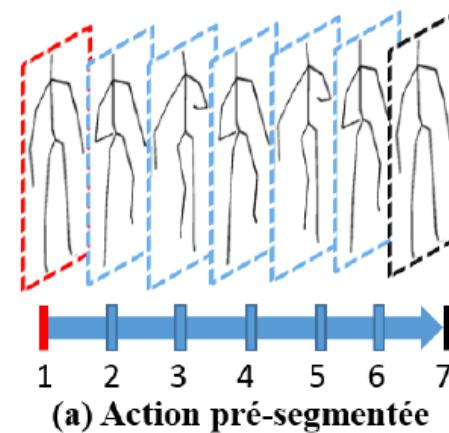
- [Devineau, Xi, Moutarde, Yang, IEEE Automatic Face and Gesture Recognition, 2018]



Offline Action Recognition / Online Action Detection

- Pre-segmented action recognition: off-line labeling of actions, and recognition of the action when it is achieved
- OAD : Online Action Detection: important research area in computer vision and machine learning
 - Continuous stream of data: the starting and ending of gestures is not known!
 - In-line action recognition

Offline Action Recognition / Online Action Detection



On-line Action Detection

■ Main difficulties

- Temporal variability
- Spatial inter-classes variability
- Spatial intra-classes variability

On-line Action Detection

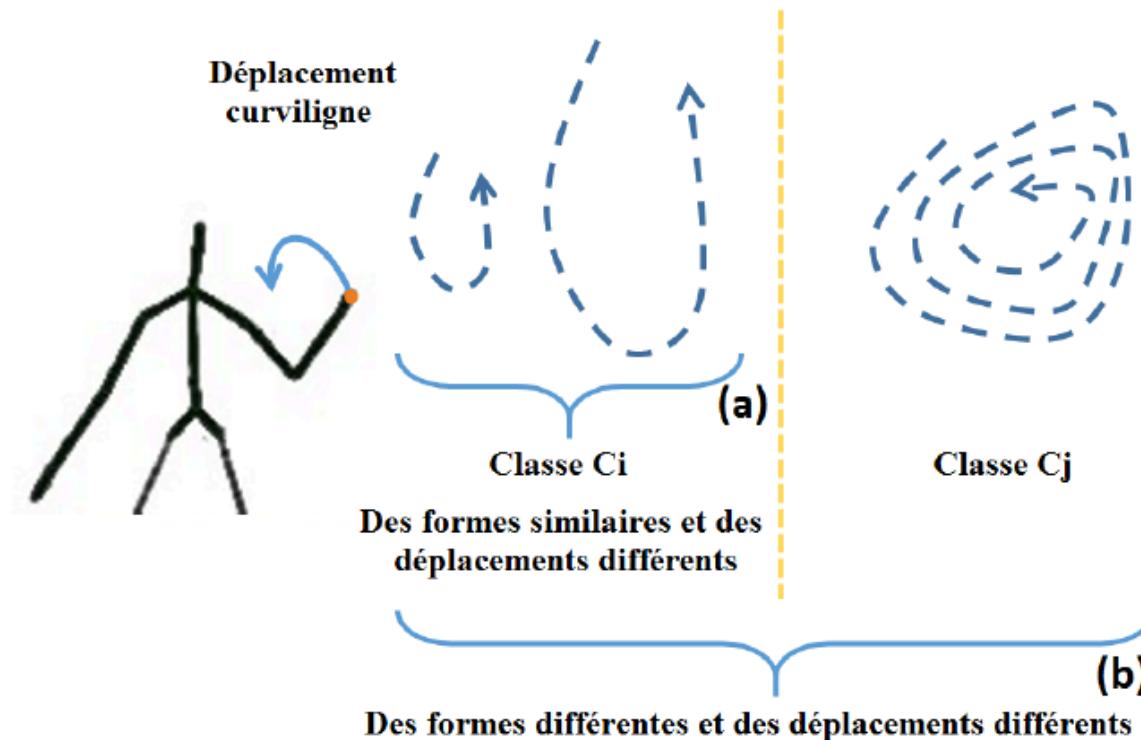
■ Temporal variability

- Actions performed with different velocities
- Pauses inserted inside the movement sequences
- > Difficult to predict the duration of the action

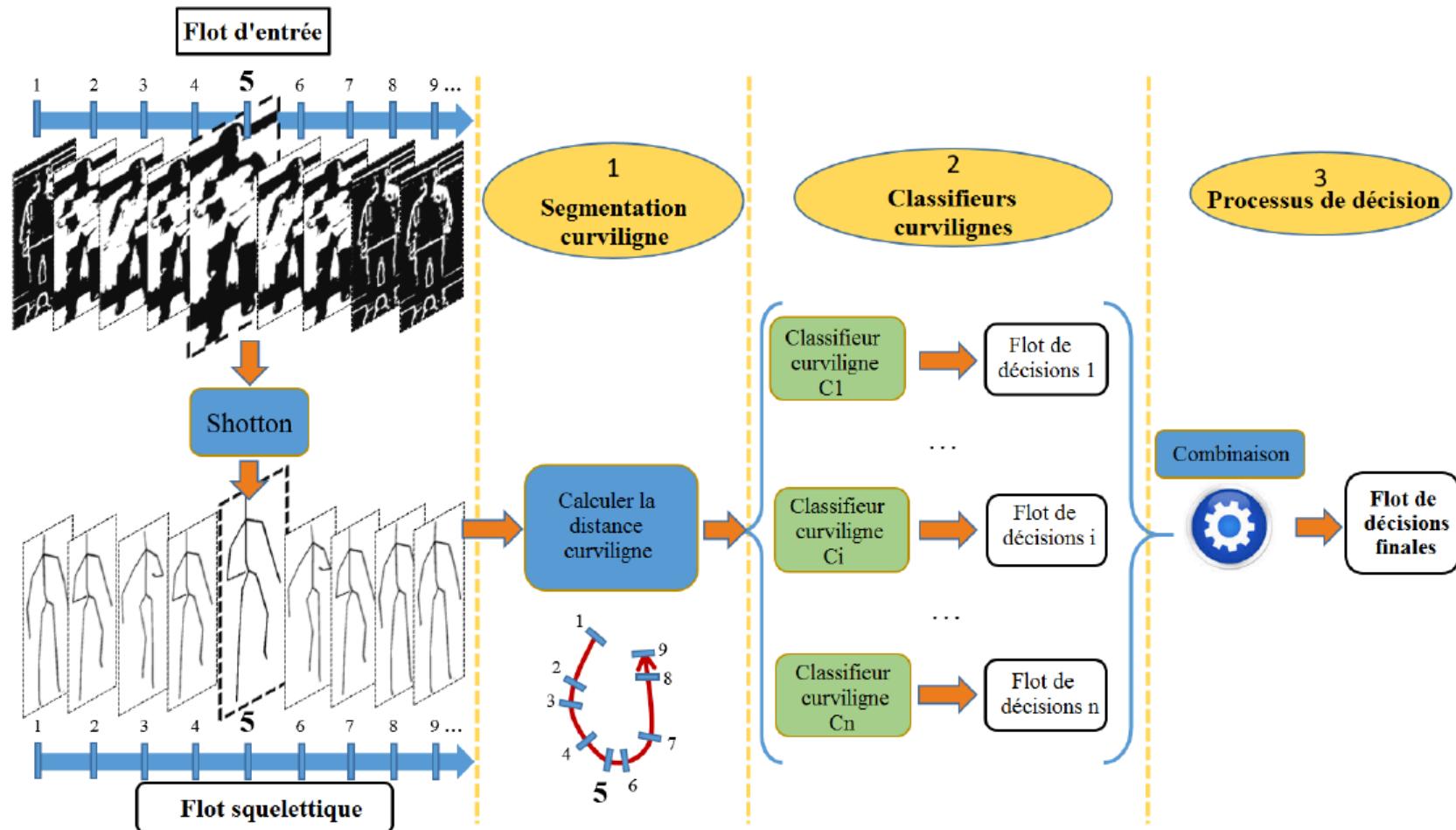
On-line Action Detection

■ Spatial inter-classes and intra-classes variability

- Similar lengths for 2 instances of Ci and Cj and similar starting shapes ((b): two different classes with similarities, with repetitions)
- Different lengths of the trajectories for 2 instances of the same class



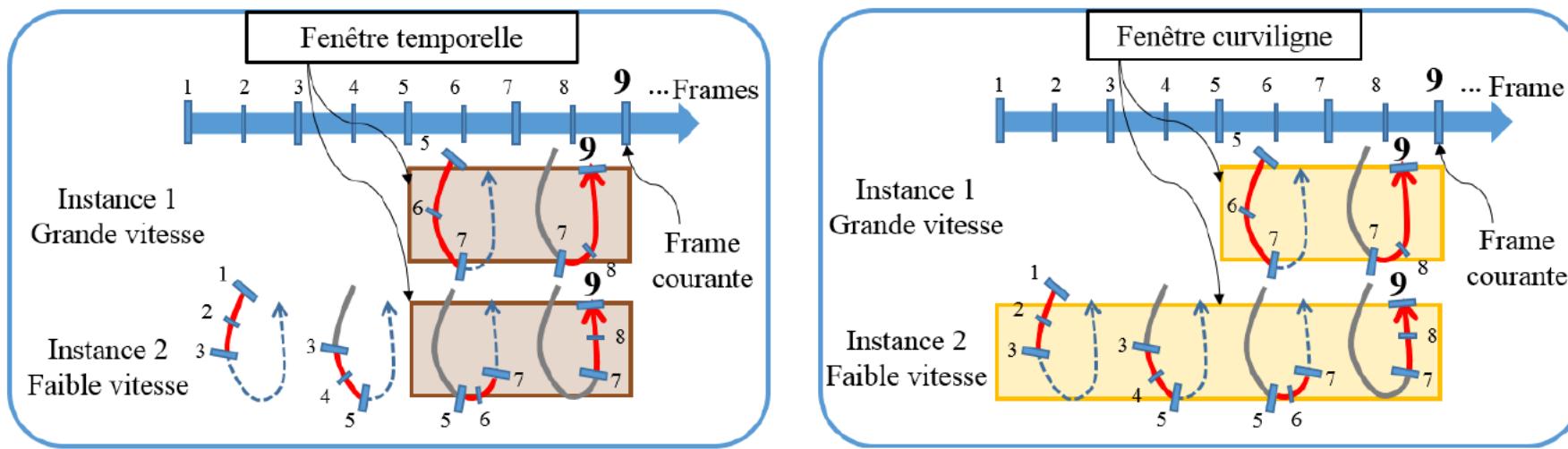
On-line Action Detection (Boulahia et al, IRISA-Intuidoc, Mimetic)



On-line Action Detection (Boulahia et al, IRISA-Intuidoc, Mimetic)

■ Originality of the approach

- Curvilinear segmentation : temporal window vs. curvilinear window



On-line Action Detection (Boulahia et al, IRISA-Intuidoc, Mimetic)

■ Curvilinear segmentation

- Distance $CuDi(F_S, F_E)$ for a segment of the movement, from frame F_S to frame F_E :

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

- Where d_i^{Moy} : average instantaneous displacement for each frame i and V trajectories

$$d_i^{Moy} = \sqrt{\sum_{v=1}^V (d_i^v)^2}$$

- And d_i^v : value of the displacement of the vème skeletal vector

$$d_i^v = \sqrt{(x_i^v - x_{i-1}^v)^2 + (y_i^v - y_{i-1}^v)^2 + (z_i^v - z_{i-1}^v)^2}$$

On-line Action Detection (Boulahia et al, IRISA-Intuidoc, Mimetic)

■ Originality of the approach

- C_i : classifiers, $i \leq n$
- G_i : class of action; all the actions have different curvilinear displacements, so there are as many classifiers as actions
- Curvilinear classifiers C_i : we calculate the quantity of displacement $CuDi_i$ of the class of action G_i , as the average of all the curvilinear displacements of instances belonging to class G_i (with K the total number of instances of class G_i)

$$CuDi_i = \frac{1}{K} \times \sum_{k=1}^K CuDi(F_{S_k}, F_{E_k})$$

- Each classifier is a SVM multi-classes, trained with the strategy one-against-all.

Datasets for segmented action recognition

- HDM05 – Skeleton dataset – Opto-electronic capture
[Müller et al., Mocap database hdm05, 2007]

	Label des classes	Nombre de trames	Nombre de séquences	# séquences par sujet				
				bd	bk	dg	mm	tr
1	Déposer par terre (Deposit floor)	11,623	32	6	6	6	8	6
2	Courir sur place (Elbow to knee)	5756	13	3	3	3	1	3
3	Grab high	7506	29	4	6	6	6	7
4	Sauter pieds joints (Hop both legs)	2327	12	4	3	3	1	1
5	Courir en demi-cercle (Jog)	4142	17	2	5	3	3	4
6	Coup de pied vers l'avant (Kick forward)	6225	29	6	6	6	6	5
7	S'allonger par terre (Lie down floor)	13,100	20	4	6	4	4	2
8	Tournez les bras en arrière (Rotate both arms backward)	1742	17	4	4	3	3	3
9	Marcher (Sneak)	3480	16	3	3	4	3	3
10	S'accroupir (Squat)	10,035	50	10	12	12	4	12
11	Lancer un ballon (Throw basketball)	5710	14	3	3	3	2	3
Total	11 classes	71,646	249	49	57	53	41	49

Datasets for segmented action recognition

■ UTKinect-Action – [Xia et al., CVPRW 2012]

	Label des classes	Nombre de séquences par classe	Nombre de sujets
	Marcher (Walk)	20	10
	S'asseoir (Sit down)	20	10
	Se lever (Stand up)	20	10
	Ramasser (Pick up)	20	10
	Porter (Carry)	20	10
	Jeter (Throw)	20	10
	Pousser (Push)	20	10
	Tirer (Pull)	20	10
	Agiter les mains (Wave hands)	20	10
	Battre des mains (Clap hands)	20	10
Total	10 classes	200 séquences	10 sujets

Datasets for segmented action recognition

- M2S dataset SKBM13 [Sorel et al., IJPRAI, 2013]

	Label des classes	Nombre de séquences par classe	Nombre de sujets
	Applaudir (Applause)	70	-
	Croiser les bras (Cross arms)	68	-
	Prendre milieu (Grasp chest)	68	-
	Prendre haut (Grasp high)	70	-
	Prendre bas (Grasp hip)	68	-
	Mains sur les hanches (Hand hip)	70	-
	Mains dans les poches (Hand pocket)	68	-
	Saluer au niveau de la tête (Hello head)	70	-
	Saluer haut (Hello high)	70	-
	Coup de poing (Punch)	68	-
	Coup de poing remontant (Uppercut)	72	-
	Claque revers (Slap back)	70	-
	Claque paume (Slap palm)	72	-
	Jeter (Throw)	70	-
	Toucher le menton (Touch chin)	68	-
Total	15 classes	1042 séquences	10 sujets

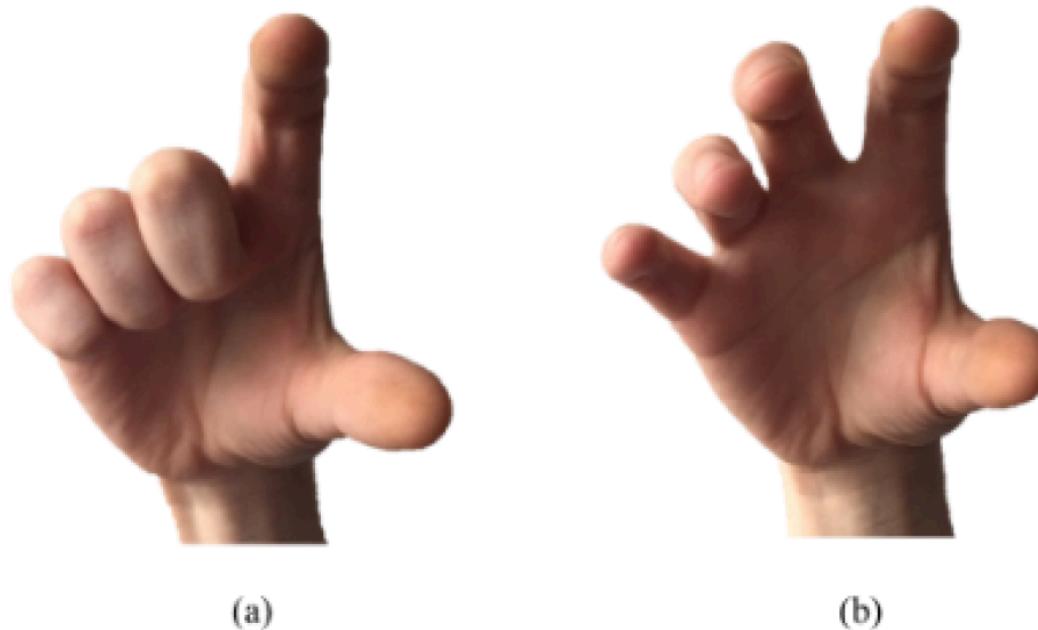
Datasets for online hand gesture recognition

- DHG dataset [De Smedt et al., IEEE CVPR, 2016]

	Label des classes	Étiquette	Nombre de séquences par classe	Nombre de sujets
	Saisir (Grab)	G	200	20
	Étendre (Expand)	E	200	20
	Pincer (Pinch)	P	200	20
	Rotation sens horaire (Rotation CW)	R-CW	200	20
	Rotation sens antihoraire (Rotation CCW)	R-CCW	200	20
	Appuyer (Tap)	T	20	10
	Défiler vers la droite (Swipe Right)	S-R	200	20
	Défiler vers la gauche (Swipe Left)	S-L	200	20
	Défiler vers le haut (Swipe Up)	S-U	200	20
	Défiler vers le bas (Swipe Down)	S-D	200	20
	Marquer un X (Swipe X)	S-X	200	20
	Marquer un V (Swipe V)	S-V	200	20
	Marquer un + (Swipe +)	S-+	200	20
	Secouer (Shake)	Sh	200	20
Total	14 classes	-	2800 séquences	20 sujets

Datasets for online hand gesture recognition

- DHG dataset : example of grasping gesture



(a)

(b)

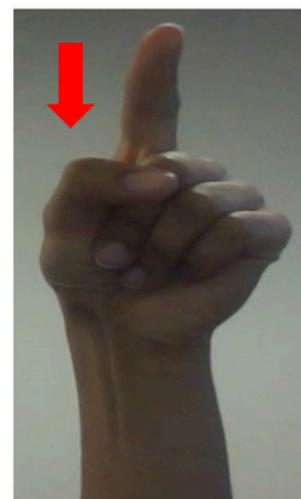
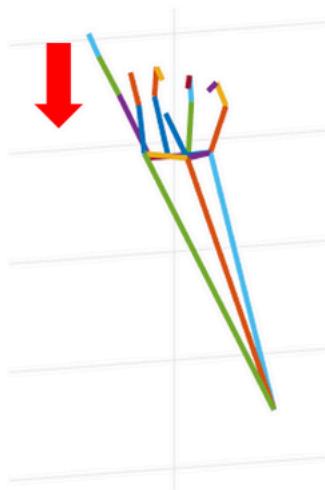
Datasets for online hand gesture recognition

- LMDHG dataset – données pré-segmentées [Boulaya et al., IPTA 2017]
<https://www-intuidoc.irisa.fr/english-leap-motion-dynamic-hand-gesture-lmdhg-database/>

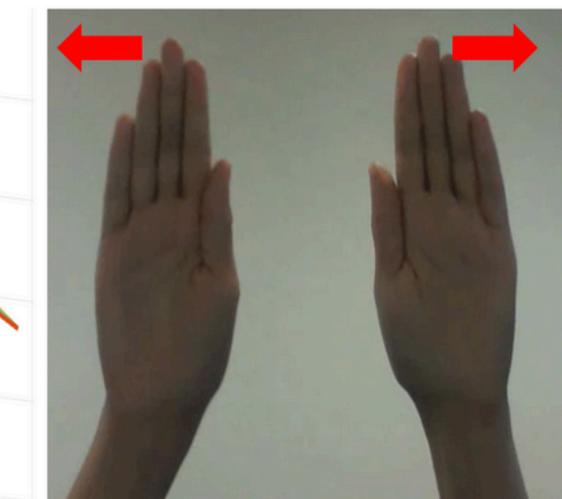
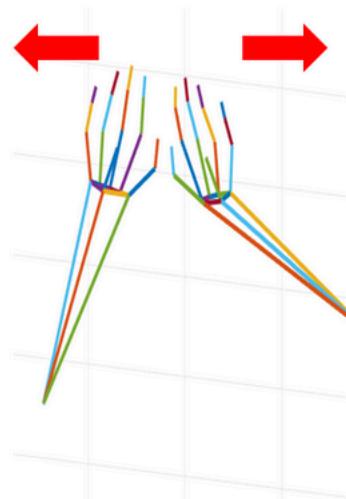
Classes de geste	#Mains	Étiquette
Point to	1	HG1
Catch	1	HG2
Shake with two hands	2	HG3
Catch with two hands	2	HG4
Shake down	1	HG5
Shake	1	HG6
Draw C	1	HG7
Point to with two hands	2	HG8
Zoom	2	HG9
Scroll	1	HG10
Draw Line	1	HG11
Slice	1	HG12
Rotate	1	HG13

Datasets for online hand gesture recognition

- LMDHG dataset



(a)



(b)

Datasets – skeletal action recognition

- [MRC⁺07] Meinard Müller, Tido Röder, Michael Clausen, Bernhard Eberhardt, Björn Krüger, and Andreas Weber. Mocap database hdm05. *Institut für Informatik II, Universität Bonn*, 2 :7, 2007.
- [XCA12] Lu Xia, Chia-Chih Chen, and JK Aggarwal. View invariant human action recognition using histograms of 3d joints. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 20–27. IEEE, 2012.
- [SKBM13] Anthony Sorel, Richard Kulpa, Emmanuel Badier, and Franck Multon. Dealing with variability when recognizing user’s performance in natural 3d gesture interfaces. *International Journal of Pattern Recognition and Artificial Intelligence*, 27(08) :1350023, 2013.

Datasets – Hand gesture online recognition

- [DSWV16] Quentin De Smedt, Hazem Wannous, and Jean-Philippe Vandeborre. Skeleton-based dynamic hand gesture recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–9, 2016.
- Boulahia, S. Y., Anquetil, E., Multon, F., & Kulpa, R. Dynamic hand gesture recognition based on 3D pattern assembled trajectories. In *7th IEEE International Conference on Image Processing Theory, Tools and Applications (IPTA 2017)*, (pp. 1-6).