Reconnaissance d'actions

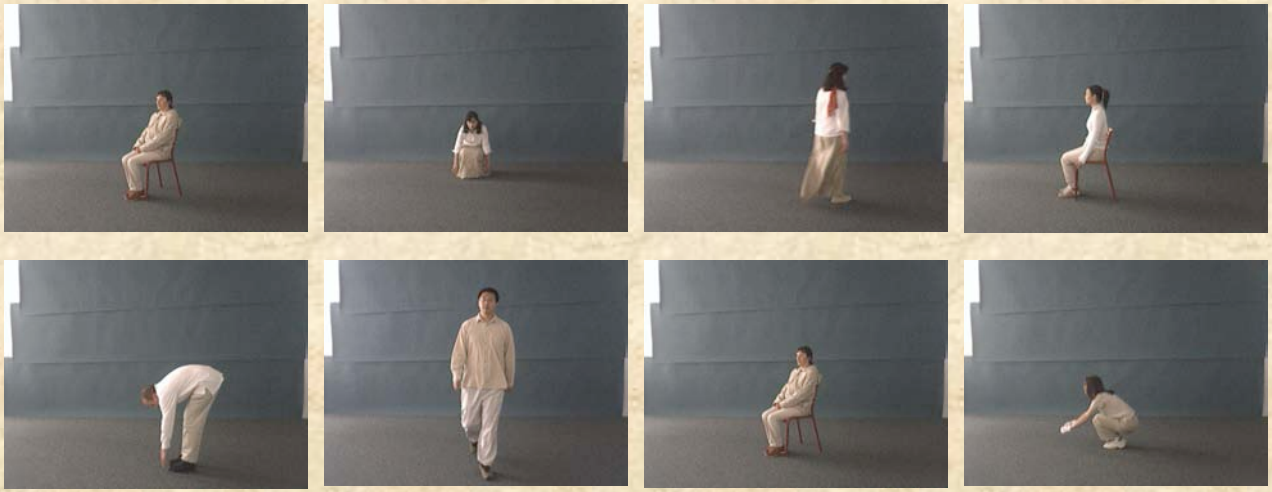
Catherine Achard, Arash Mokhber,
Xingtai Qu et Maurice Milgram

Institut des Systèmes Intelligents et Robotique
Université Pierre & Marie Curie, Paris, France

But:Reconnaître des actions usuelles

La base comporte 8 actions réalisées par 7 personnes (1614 séquences):

(1)	s'accroupir	(5)	marcher
(2)	se relever de la position accroupie	(6)	se pencher
(3)	s'asseoir	(7)	se relever de la position penchée
(4)	se redresser	(8)	saute



Différents points de vue :

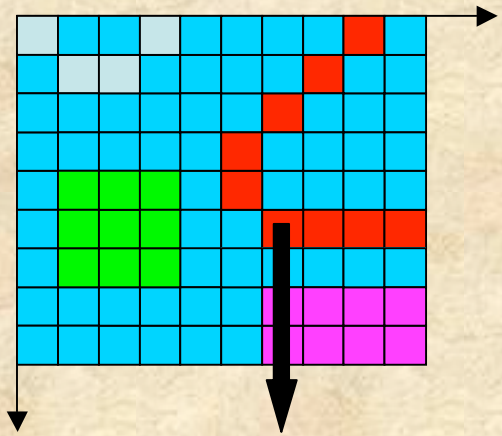
- **de face**
- **à +/- 45°,**
- **à +/- 90°**

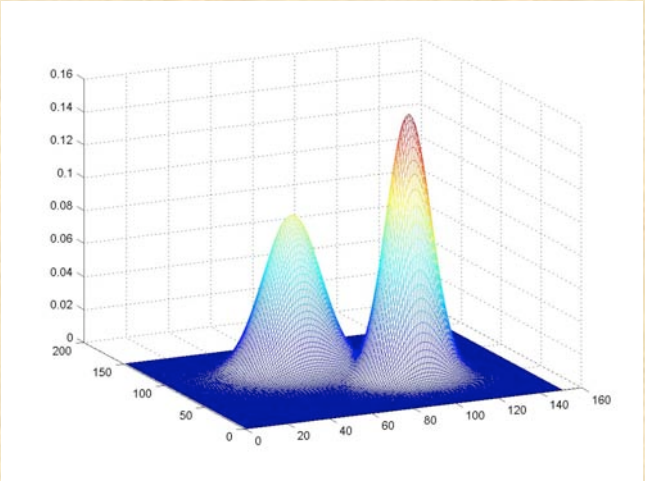
➡ **37 actions.**

Idée :

➢**Extraire les silhouettes binaires et réaliser un apprentissage des actions par des chaînes de Markov cachées**

➢Détection de mouvement par modélisation du fond par une mixture de gaussiennes





1 pixel → 1 GMM

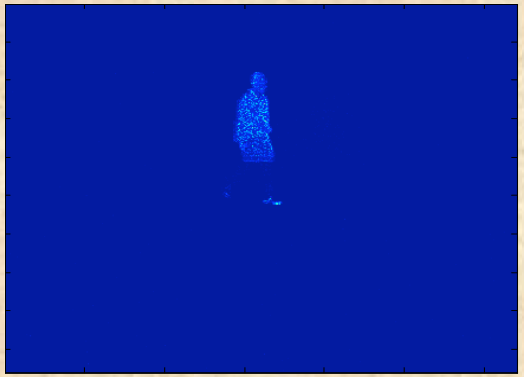
Distance entre un pixel et les centres des gaussiennes

$$D(\vec{x},\vec{m}_i) = (\vec{x}-\vec{m}_i)^T \Sigma_i^{-1} (\vec{x}-\vec{m}_i)$$

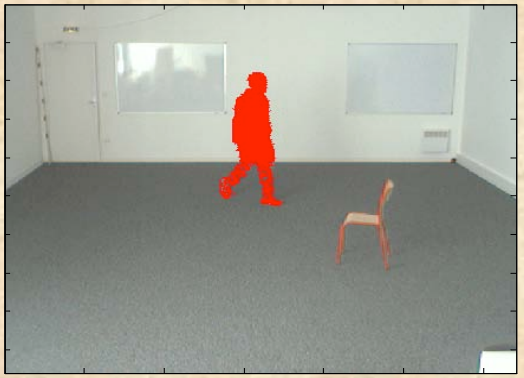
Mise à jour de l'image de référence:

$$\mu_t = (1-\alpha)\mu_{t-1} + \alpha X_t$$
$$\Sigma_t = (1-\alpha)\Sigma_{t-1} + \alpha (X_t-\mu_t)(X_t-\mu_t)^T$$

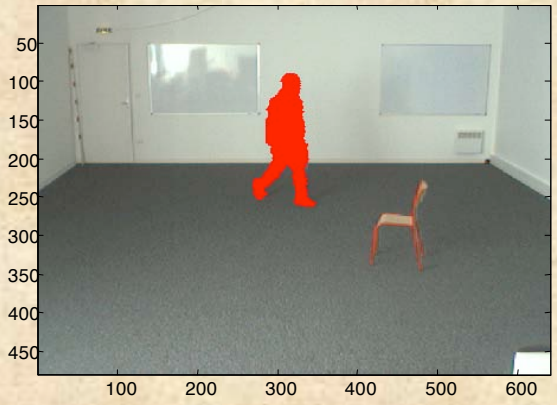
Distance aux gaussiennes



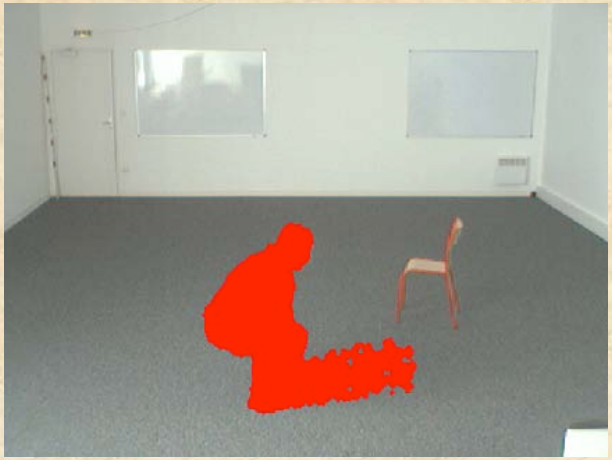
Seuillage



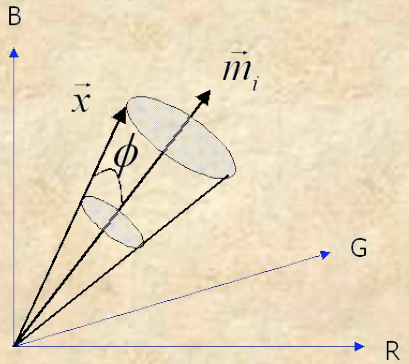
Morphologie



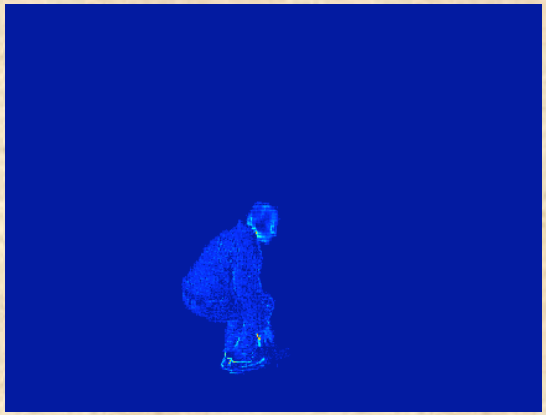
Problème lié aux ombres



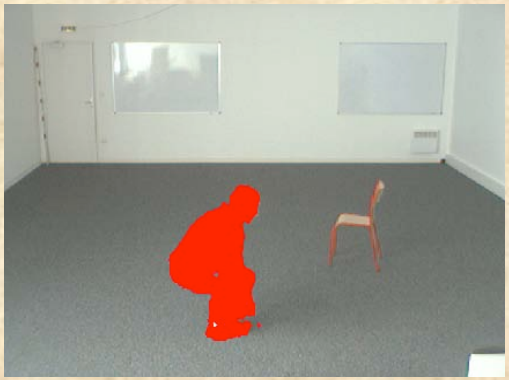
Modélisation de l'ombre par un volume conique



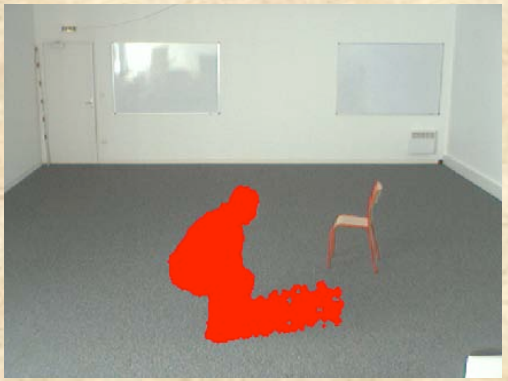
Carte d'angle



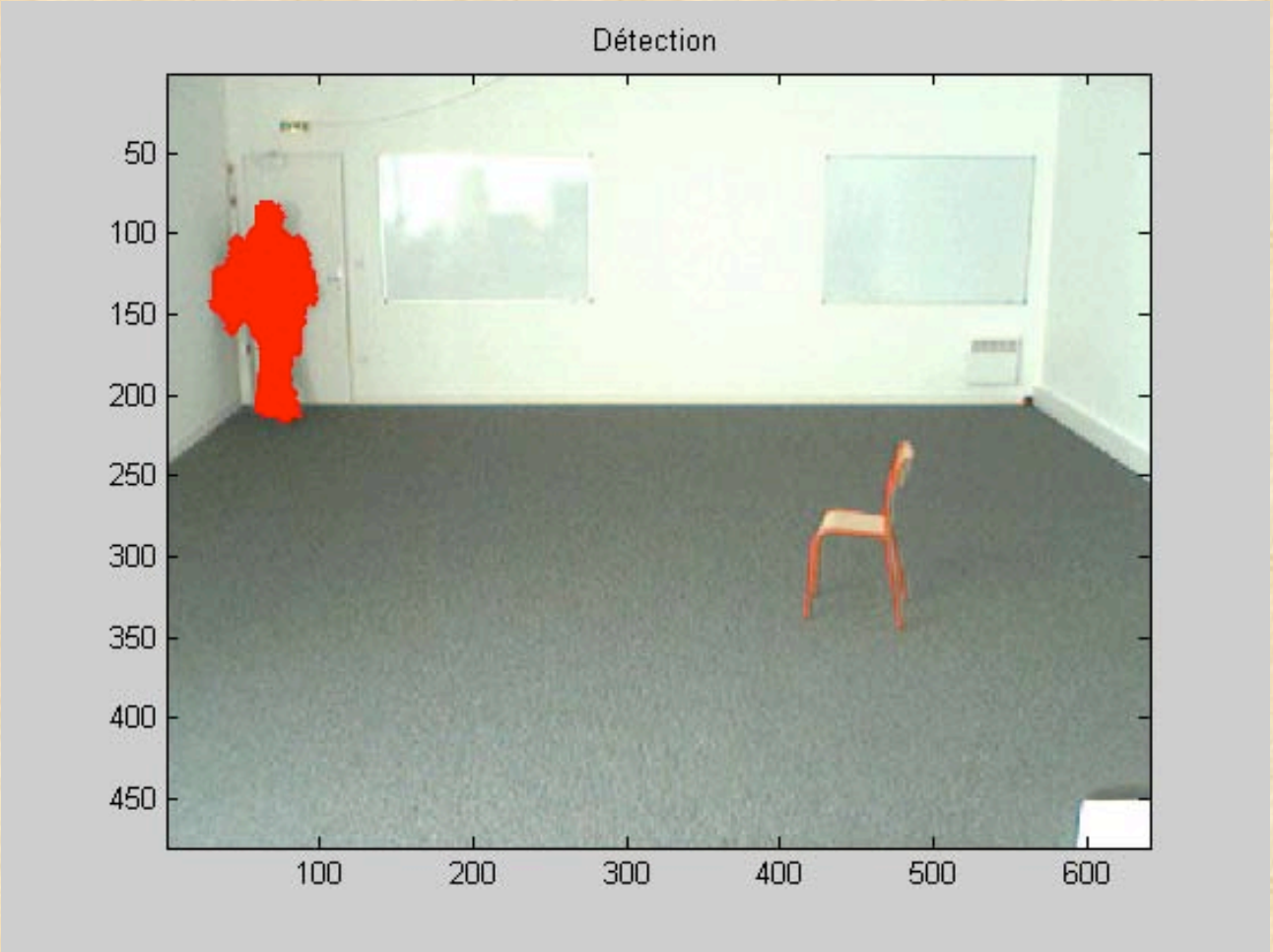
Résultats de détection

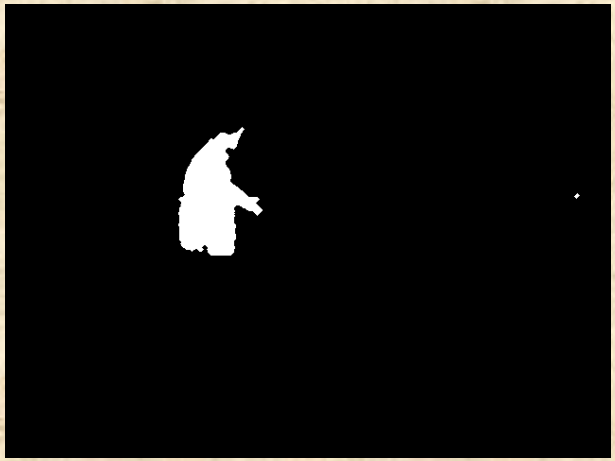


Avec modélisation de l'ombre



Sans la modélisation


Détection
50
100
150
200
250
300
350
400
450
100
200
300
400
500
600



Caractérisation des silhouettes par ses moments géométriques

$$A_{pq} = E\left\{x^p y^q\right\}$$

Centrage des moments pour être invariants en translation:

$$AC_{pq} = E\left\{(x - A_{10})^p (y - A_{01})^q\right\}$$

Normalisation qui préserve le ratio largeur hauteur pour être invariant à l'échelle :

$$M_{pq} = E\left\{\left(\frac{x - A_{10}}{AC_{20}^{1/4} AC_{02}^{1/4}}\right)^p \left(\frac{y - A_{01}}{AC_{20}^{1/4} AC_{02}^{1/4}}\right)^q\right\}$$

Moments d'ordre 2 et 3

➔Une action = une chaîne temporelle de vecteur de dimension 6

$$o = \{M_{20}, M_{11}, M_{30}, M_{03}, M_{21}, M_{12}\}$$

$$O = \{o_1, o_2, \ldots, o_T\}$$

Ces chaînes sont apprises avec les HMM

Les chaînes de Markov cachées.

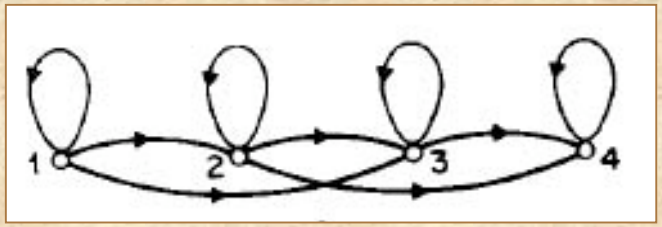
En général, un HMM est noté $\Lambda = (A, B, \pi)$ où:

- A est une matrice de transition entre ses états si.

$$a_{ij} = P[q_{t+1} = j \mid q_t = i] \quad 1 \le i, j \le N$$

- B est une matrice d'observation.

$$b_j(k) = P[o_t = v_k \mid q_t = j] \quad 1 \le j \le N, \; 1 \le k \le M$$



- π est un vecteur de probabilité initiale.

$$\pi_i = P[q_1 = i] \quad 1 \le i \le N$$

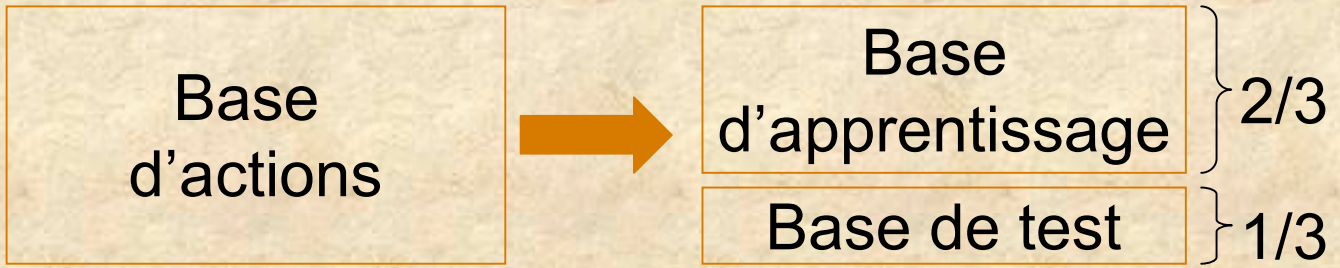
En pratique, nous avons utilisé une gaussienne pour modéliser le vecteur de caractéristique

$$\Lambda = (A, \pi, \mu, \sigma)$$

- L'ensemble des paramètres Λ_k est appris pour chacune des 37 classes sur une base d'apprentissage avec l'algorithme de Baum-Welch

- Il utilise la méthode EM (Expectation- Maximization) pour maximiser la vraisemblance que les HMMs génèrent les séquences d'apprentissage

Séparation de la base de données:



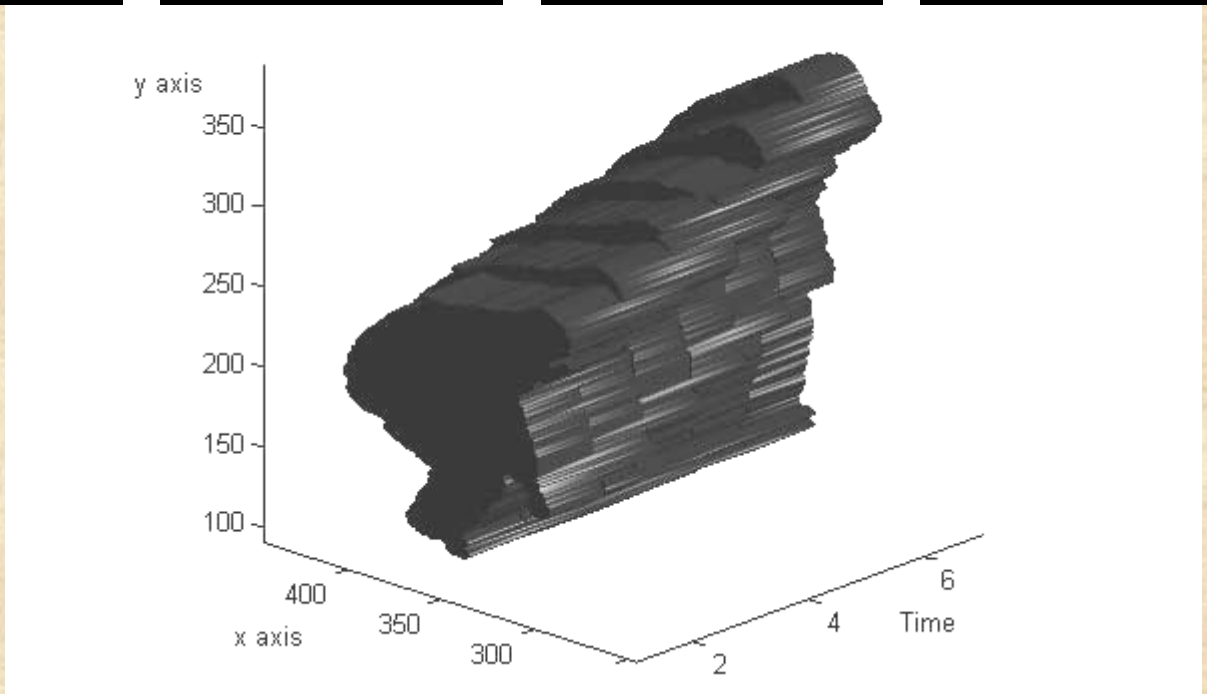
- Apprentissage
  - ➔ $\Lambda = (A, \pi, \mu, \sigma)$ pour chacune des 37 classes
- Reconnaissance
  - ➔ On recherche la chaîne qui maximise la probabilité d'observation $P(O/\Lambda)$

➔Très mauvais résultats dus à la normalisation

Accroupit face lève

Saute

Idée : travailler sur une fenêtre temporelle et caractériser des mico-volumes



Volume caractérisé par ses moments géométriques :

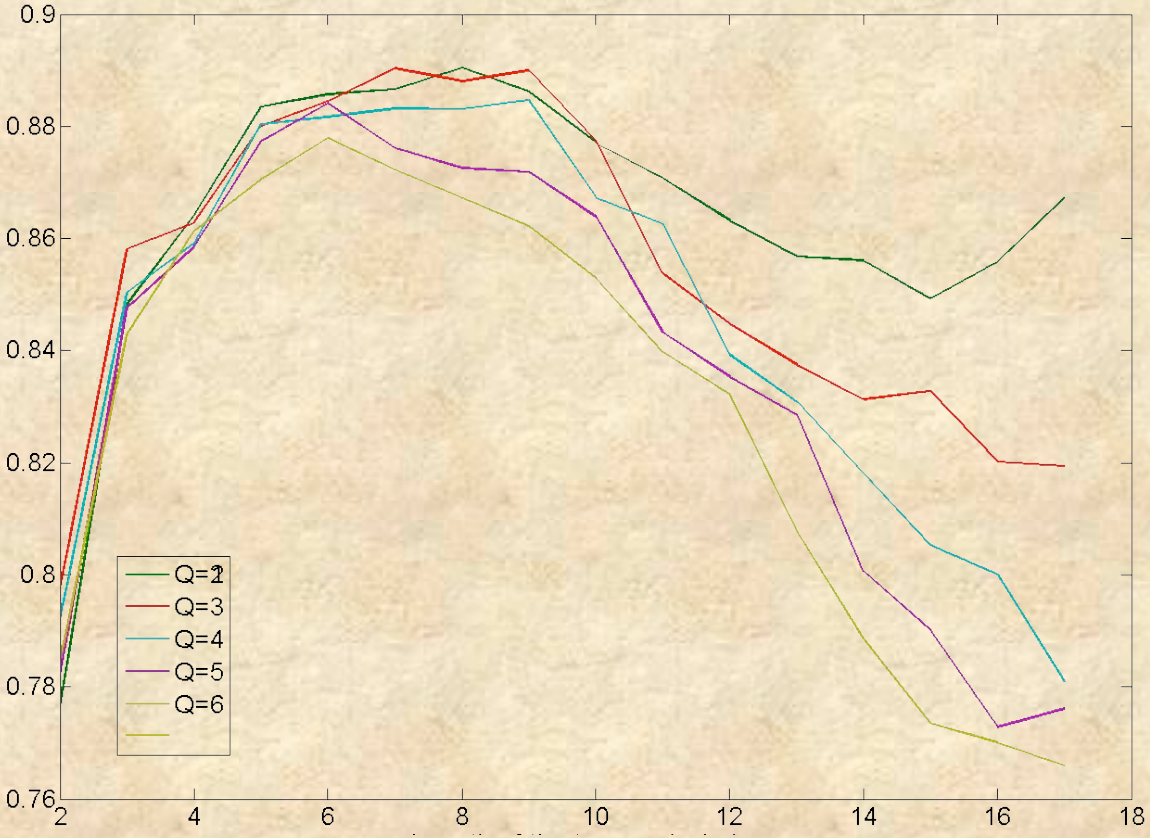
$$\mu_{ijk} = \frac{1}{A} \sum_{(x,y,t)\in A} \left(\frac{y - y_0}{\sqrt{\sigma_x \sigma_y}} \right)^i \left(\frac{x - x_0}{\sqrt{\sigma_x \sigma_y}} \right)^j \left(\frac{t - t_0}{\sigma_t} \right)^k$$

Moments invariants en

- translation,
- changement d'échelle spatiale

Moments du 2nd et 3ième ordre → vecteur de dimension 14

Etude en fonction de la longueur de la fenêtre temporelle et du nombre d'état des HMMs (taux moyens pour les 7 personnes)



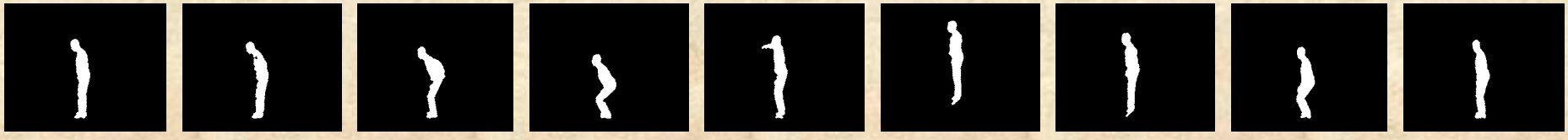
Meilleur résultat (89%) obtenu pour 3 états et 7 images par micro-volume

Matrice de confusion moyenne

(1) s'accroupir
(2) se relever de la position accroupie
(3) s'asseoir
(4) se redresser
(5) marcher
(6) se pencher
(7) se relever de la position penchée
(8) sauter

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	**88.6**	0	0.3	0	0	2.6	0	8.5
2	0	**93.8**	0	0.6	0	0	0.3	5.2
3	0.83	0	**81.5**	0.28	1.7	3.9	0.3	11.6
4	0	3.6	0	**81.5**	0	0	10.2	4.7
5	0	0	0.1	0.6	**95.4**	2.4	0.1	1.3
6	6.2	0	5.1	0	0.2	**84.6**	0	3.9
7	0	2.9	0.4	2.1	0.4	0.4	**91.5**	2.3
8	0	1.29	3.3	0	0.4	0.4	2.5	**92.1**

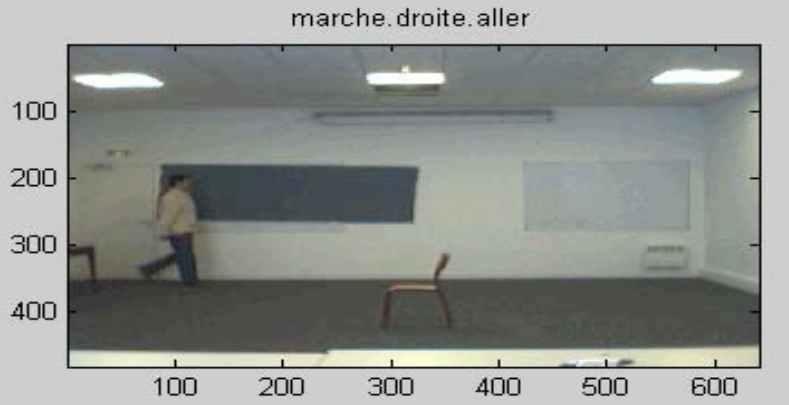
Une séquence d'image d'une personne qui saute

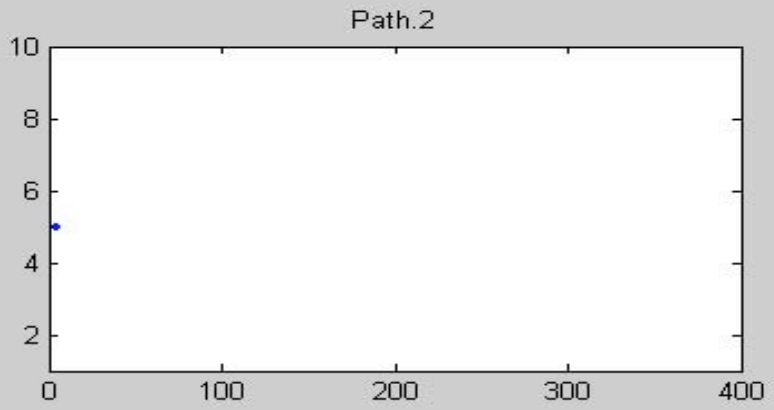


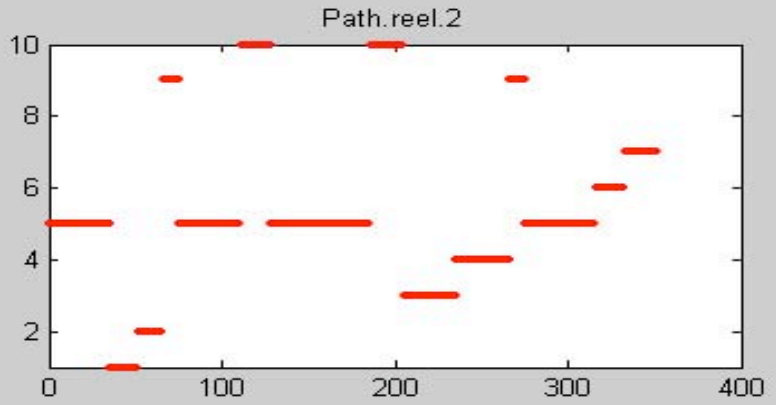
Base d'actions → Base d'apprentissage (6 personnes) / Base de test (1 personne)

Personne	1	2	3	4	5	6	7	Moy
Taux	95.3	93.6	80.2	90.8	88.2	93.3	70.3	89.0

Problème de la septième personne:


marche.droite.aller
100
200
300
400
100
200
300
400
500
600


Path.2
10
8
6
4
2
0
100
200
300
400


Path.reel.2
10
8
6
4
2
0
100
200
300
400

Autre idée: caractériser directement toute la séquence donc tout le volume par les moments géométriques

➔ Une séquence = un vecteur de dimension 14

Taux de reconnaissance obtenus en plaçant tour à tour chaque personne dans la base de test

Personne	1	2	3	4	5	6	7	Moy
Taux	89.9	90.2	82.7	97.2	92.1	95.2	77.1	89.5

Matrice de confusion moyenne

(1) s'accroupir
(2) se relever de la position accroupie
(3) s'asseoir
(4) se redresser
(5) marcher
(6) se pencher
(7) se relever de la position penchée
(8) sauter

	1	2	3	4	5	6	7	8
1	**97.2**	0.0	0.0	0.0	0.0	2.8	0.0	0.0
2	0.0	**90.5**	0.0	0.0	0.0	0.0	9.5	0.0
3	4.8	0.0	**84.1**	0.0	0.0	9.4	1.2	0.5
4	0.0	3.3	0.0	**76.1**	0.0	0.0	17.1	3.6
5	0.0	0.0	0.0	0.0	**98.4**	0.9	0.2	0.4
6	11.1	0.0	0.0	0.0	0.0	**88.3**	0.0	0.5
7	0.0	10.7	0.0	0.7	0.0	0.0	**88.2**	0.3
8	0.0	8.6	0.0	0.0	0.0	1.7	0.9	**88.8**

Volumes non binaires



Personne	1	2	3	4	5	6	7	Moy
Taux	92,6	88,3	88,1	93,1	89,8	96,2	77.1	90,0

Personne	1	2	3	4	5	6	7	Moy
Taux	95,8	85,9	85,7	94,4	88,3	97,1	74	89,7

Sans segmentation préalable,

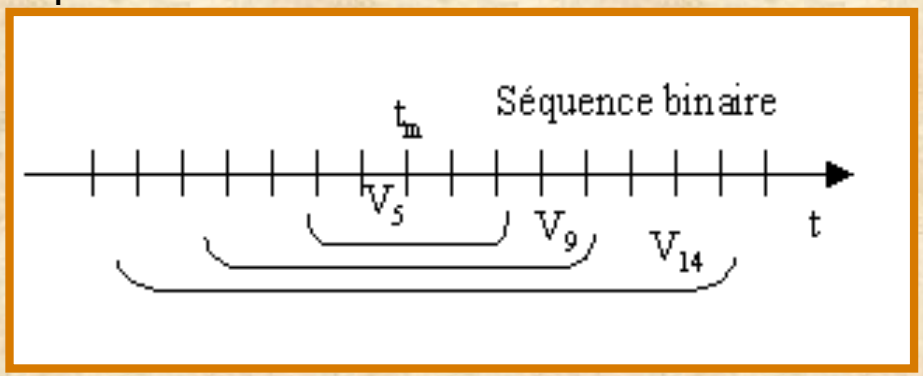
- Calcul du flot optique sur chaque image:

- Estimation des moments géométriques centrés, réduits

Personne	1	2	3	4	5	6	7	Moy
Taux	90,2	75,2	89	91,3	81,1	84,8	87,5	85,9

Segmentation

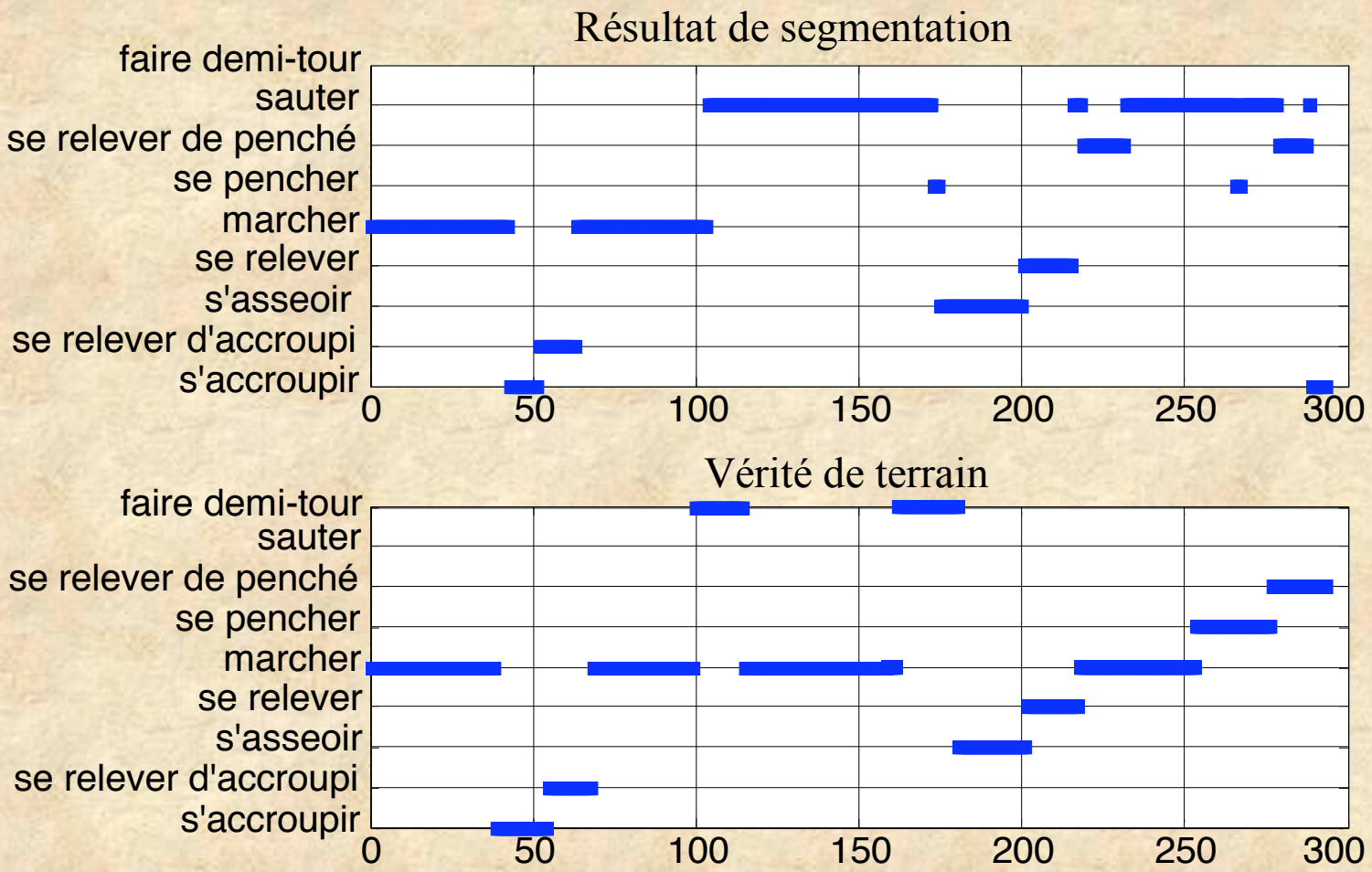
Calcul de vecteurs moments sur les volumes binaires V_i centrés autour de l'instant de référence



Volumes contenant 5 à 50 images

→ 46 vecteurs à comparer avec ceux de la base

→ Choix du vecteur le plus proche et attribution de l'étiquette de la référence la plus proche


Résultat de segmentation
faire demi-tour
sauter
se relever de penché
se pencher
marcher
se relever
s'asseoir
se relever d'accroupi
s'accroupir
0
50
100
150
200
250
300
Vérité de terrain
faire demi-tour
sauter
se relever de penché
se pencher
marcher
se relever
s'asseoir
se relever d'accroupi
s'accroupir
0
50
100
150
200
250
300

Conclusion

- Comme en signal, la caractérisation n'est pas robuste si elle est réalisée image par image
- Mise en place de deux méthodes de reconnaissance d'actions. La première modélise les actions par les HMMs, tandis que la seconde caractérise les actions de manière globale

Bilan

- Résultats similaires quant aux taux de reconnaissance
- Les approches par HMMs facilitent la segmentation des séquences

Perspectives

- Etendre la base de données et principalement, le nombre d'acteurs
- Comparer avec des méthodes « globales » basées sur l'algorithme adaboost
- Utiliser une version multi-classes d'adaboost pour détecter plusieurs actions
- Réaliser une sélection de caractéristiques