Fares Haddad. fhaddad@etu.info.unicaen.fr
Master Recherche Langue-Image-Document
Mai - Septembre 2007

Catégorisation de séquences d'activités humaines par marches aléatoires sur graphe

Responsable du stage : Youssef Chahir. chahir@info.unicaen.fr

Table des matières

Table des figures

Liste des tableaux

Remerciements

Je tiens tout d’abord à remercier mon responsable de stage Youssef Chahir pour sa confiance, sa patience, ses encouragements et l’aide précieuse qu’il m’a apporté tout le long du stage.

J'adresse mes remerciements à tous mes collègues, en particulier Mourad Maameri (le stagiaire qui a travaillé avec moi et qui a contribué à l'avancement de ce travail).

Et plus particulièrement, par ce document, je tiens à montrer toute ma reconnaissance à ma famille qui a permis l'aboutissement de mes années d'étude. À mes parents, à mes frères et sœurs, à mes neveux et nièces, à ma cousine samia et son époux amar.

Introduction

Ces dernières années, le problème de reconnaissance et de classification d'activités humaines a suscité l'intérêt de communautés de recherche plus larges, allant des neurosciences du mouvement, la biomécanique, l'informatique, les sciences de la communication et les sciences de l'ingénierie. Dans plusieurs applications, telles que la vidéo surveillance, l'archivage et l'indexation de vidéos, il est important de reconnaître les mouvements des personnes pour pouvoir interpréter leurs comportements. Cette reconnaissance d'activité nécessite l'extraction de données multiples, l'interprétation automatique des séquences vidéos, et fait appel à des techniques d'analyse vidéo (perception visuelle, estimation de mouvement,…) et des méthodes d'analyse et de classification de données. Ce problème d'identification devient crucial quand on a un nombre croissant d'individus sous différents points de vue de caméras, et dans des environnements complexes. Pour simplifier le problème d'identification des actions, une stratégie commune a été adoptée par une majorité de chercheurs qui consiste à traiter les actions d'un seul point de vue.

Dans ce travail, nous nous sommes intéressés à la détection, la reconnaissance et la représentation des actions de la vie de tous les jours « marcher », « courir », « sauter »… Le problème de reconnaissance de l'activité est généralement divisé en deux parties. La première consiste à détecter et suivre une personne en mouvement tandis que la seconde concerne la reconnaissance des parties du corps ou de la silhouette.

La plupart des algorithmes de détection du mouvement présents dans la littérature sont présentés comme des méthodes de soustraction de l'arrière-plan (background subtraction) que nous avons utilisé pour détecter les silhouettes des personnes en mouvement.

Une action humaine étant fortement liée au mouvement, nous proposons dans ce travail d'extraire la silhouette de la personne en action et de suivre l'objet en mouvement et de former un volume dans l'espace 3D (2d+t). Ce volume qui représente une action donnée sera caractérisé par des moments géométriques 3D qui sont invariants à la translation et au changement d'échelle. L'objectif du travail est d'implémenter une ou plusieurs méthodes d'extraction de silhouettes de personnes en actions et d'exploiter une nouvelle approche de catégorisation des actions basée sur l'exploration de graphe par marches aléatoires et l'analyse spectrale. L'idée de base est de considérer l'ensemble des actions (vidéos) comme un graphe pondéré, où les sommets du graphe sont représentés par les volumes 3D (séquences des actions), et les arêtes connectés représentent la similarité entre les noeuds. Cette mesure de similarité sera calculée par une distance euclidienne entre les vecteurs caractéristiques des actions.

Nous présentons dans la première section notre approche de segmentation d'objets binaires 3D et leur caractérisation globale par des moments géométriques 3D. Ensuite , après un rappel du principe de l'analyse spectrale et des marches aléatoires sur graphe, nous présentons les résultats de validation sur un corpus d'actions qui représentent une dizaine d'actions de plusieurs personnes.

Extraction de données d'une vidéo

1. Etat de l'art

De nombreuses recherches ont été menées ces dernières années sur la reconnaissance des activités humaines. Les différentes approches proposées peuvent être classées en plusieurs catégories :

approche 2D avec modèle statistique
approche 2D avec modèle explicite
approche 3D

Les méthodes d'identification des activités humaines sont généralement basées sur les modèles d'apparence 2D ou 3D. Une catégorie des travaux consiste à détecter les différentes parties du corps telles que la tête, les mains, les pieds ainsi que d'autres parties du corps telles que les articulations [B. Gaveau], [T. Zhao]. Haritaoglu et al. [B. Gaveau] proposent un système de reconnaissance globale d'actions qui est basé sur les projections horizontales et verticales de la silhouette de la personne, et de son orientation par rapport à la caméra (vue de face, vue de côté gauche, ...).

Dans le projet de [Pfinder, 1997] ils déterminent les différentes pièces du corps directement lors de la phase de segmentation en utilisant un modèle statistique multi-classe, de couleur et de forme pour obtenir une représentation 2D de la tête et les mains (fig1).

Figure 1: Extraction du corps et représentation 2D avec un blob statistique.

Certaines techniques ont besoins d'initialiser les mains et des pièces du corps comme dans [Bergler, 1998]. Pour que ces approches fonctionnent correctement toutes les pièces du corps doivent être détectées. Elles sont généralement très sensibles aux erreurs de segmentation.

Pour éviter ces inconvénients, les approches 2D avec modèle statistique identifient l'action sans avoir à détecter les différentes pièces du corps [Boulay, 2003]. Les actions sont habituellement décrites en terme statistique. [Bauberg, 1995] utilisent les points saillant sur le bord de la silhouette pour décrire sa forme .

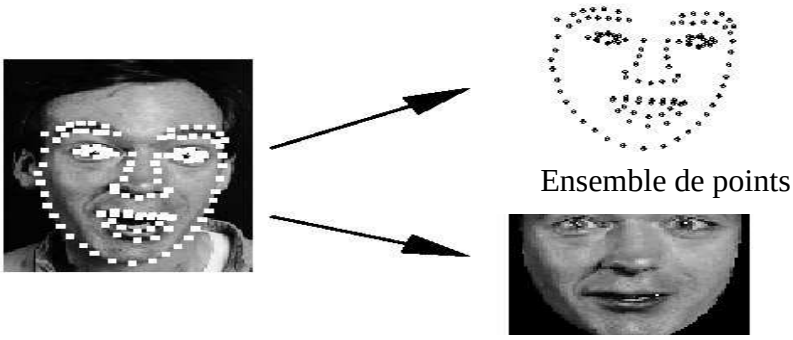


Figure 2: Extraction des points saillant d'un visage

I wasawa et al [S. Ingve] ont proposé une méthode qui consiste d'abord à déterminer le centre de gravité de la silhouette, ensuite qui calcule l'orientation de la moitié supérieure du corps, et enfin d'estimer les différentes parties significatives du corps en utilisant une analyse heuristique du contour de la silhouette. D'autres travaux, cherchent à suivre et interpréter le mouvement humain dans l'action. Efros et al. dans [F. Chung] comparent deux actions en se basant sur les caractéristiques extraites, dans l'espace spatio-temporel, à partir du flot optique. Manoir et Irani [L. Zelnik-Manor] proposent une analyse multi-échelle de distributions du gradient temporel. Laptev et Lindeberg [I. Laptev] comparent deux actions par appariement de points d'intérêt (Harris). D'autres techniques d'estimation de mouvement, ont été utilisées dans le problème d'identification des actions quand il s'agissait de mouvement affine. Yang et al. [M. Yang] propose de suivre la trajectoire de la main et de la tête, qui est un mouvement affine. Une distance entre deux actions est ensuite calculée en comparant les délais, par réseau de neurones. Blank et al [M. Blank] utilisent une pile de points de silhouettes qui sont extraites et évaluées en utilisant l'équation de Poisson pour chacun des points. La comparaison de deux actions se fait par distance euclidienne entre les vecteurs caractéristiques.

Bobick et Davis proposent d'utiliser les images d'énergie du mouvement et celles de l'historique du mouvement (MHI) , et la distance de Mahalanobis entre les moments de Hue 2D pour comparer entre deux actions [A. F. Bobick].

1.1 Présentation de la base :

La base de séquences que nous avons utilisé comporte 10 actions.
marcher (walk)
courir (run)
sauter sur les 2 pieds en se déplacent (jump)
sauter sur les 2 pieds sans se déplacer (jump in place)
toucher le sol avec la main droite et se remettre debout (bend)
lever la main gauche sans se déplacer (one-hand wave)
lever les 2 mains sans se déplacer (two-hands wave)
mouvement a pas chassé (gallop sideways)

Chaque action est exécutée par 9 personnes différentes. On a donc un total de 64 vidéos de 27 armatures chacune, avec des fonds uniformes.
La résolution de chaque vidéo est de 180 * 144, 25 fps.

Des images représentatives des vidéos utilisées :

Figure 3: Images représentatives des vidéos de la base

1.2 Détection de mouvement :

Représentation de la silhouette

Avant de représenter la silhouette, il faut d'abord identifier et détecter les personnes dans la vidéo. Nous citons parmi les travaux dans ce domaine la plateforme d'interprétation visuelle [Avanzi, 2005]. Cette plateforme détecte les pixels mobiles dans une vidéo avec la segmentation. La détection est faite en soustrayant l'image courante de l'image de référence pour obtenir une image binaire :

$$I(x,y,t)=|I(x,y,t)-I(x,y,t-1)|$$
$$B(x,y,t) = 1 \text{ si } D(x,y,t) > S$$
$$0 \text{ sinon}$$

Avec I(x,y,t) est l'image courante, D(x,y,t) est l'image de différence à la position x,y, à l'instant t, S est le seuil sélectionné.

L'image de référence est mise à jour périodiquement pour tenir compte des changements de la scène. Les pixels mobiles sont alors groupés dans des régions (blobs). Un ensemble de caractéristiques 3D tel que, la position, la longueur et la taille sont calculées pour chaque blob. Les blobs sont ensuite classifiées selon une distribution probabiliste des caractéristiques 3D, à l'intérieur des classes prédéfinies. Les positions 3D des personnes sont calculées dans une matrice de calibrage.

Estimation du fond statique

Le principe des méthodes de différence par rapport a un fond statique est de construire un modèle de la scène indépendant des objets mobiles qui la traversent, de manière a ce que :

1. la différentiation à la base de la détection se fasse entre l'image courante et une image de référence de la scène au lieu de se faire entre images consécutives ;
2. les caractéristiques fines de la scène en terme de bruit temporel soient connues, de manière à avoir un filtrage des fausses alertes spatialement plus adaptatif.

Le calcul du fond statique se fait généralement par une moyenne qui peut prendre différentes formes (moyenne arithmétique, moyenne récursive, max, min, etc. . .). Pour construire un modèle statique de la scène ; deux approches sont envisageables. La première consiste à accumuler l'information contenue dans la séquence d'images, mettant ainsi progressivement en évidence les parties immobiles de la scène, et effacent ce qui bouge. Un bon effacement demande cependant un grand nombre d'images. Dans la seconde approche, Il s'agit de remplacer les objets mobiles dans une image particulière I_t par des éléments du fond cachés dans cette image, mais visibles dans une autre. Pour ce faire, on analyse l'évolution de la différence entre l'image I_t et les images consécutives. Il est toutefois difficile de fixer un seuil permettant de distinguer les différences dues au mouvement, de celles qui sont provoquées par le bruit ou par des fluctuations de luminosité dans la scène.

Estimation du fond des vidéos de la base utilisée par la moyenne arithmétique :

$$I_t^{ref} = \frac{1}{t}\left(I_t + (t-1)\, I_{t-1}^{ref} \right)$$

Figure 4: Estimation du fond des séquences vidéos

Une fois le fond statique déterminé par la moyenne, la différentiation est faite pour la détection par :

$$O_t^2(s) = |I_t(s) - I_t^{ref}(s)| \quad .$$

Un intérêt majeur des techniques de fond statique est leur capacité à détecter des cibles très petites ou qui bougent lentement.

De la même façon que l'on calcule des primitives statistiques du premier ordre, la moyenne, nous pouvons calculer des statistiques du second ordre, la variance, l'écart-type.

Technique de détection robuste :

On définit les techniques de différence d'images suivantes : une technique utilisant un seuil θ de mouvement minimal (de différence de niveau de gris), une image I(s) : $O_t^1(s) = |I_t(s) - I_{t-1}(s)|$ et une technique utilisant des différences entre l'image courante et une image de référence $O_t^2(s) = |I_t(s) - I_t^{ref}(s)|$.

L'image de référence peut être construite de différente manière. La Figure 5 présente schématiquement l'algorithme de calcul d'une différence d'image robuste.

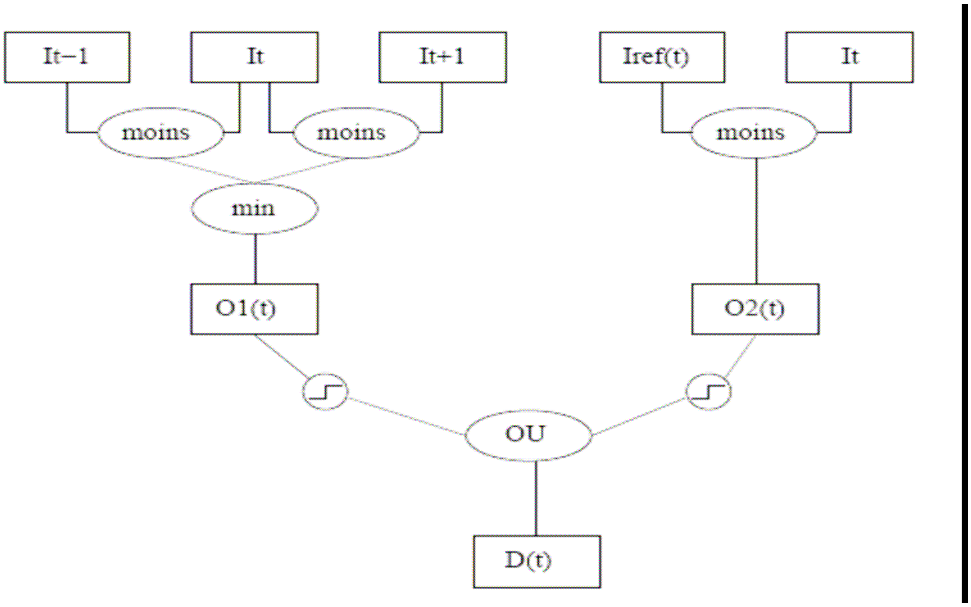


Figure 5: Différence d'image robuste

Le détecteur robuste est définie par :
un détecteur a court terme :

$$\Omega_t^1(s)=\min(O_t^1(s),O_{t+1}^1(s))$$

un détecteur a long terme :

$$\Omega_t^2(s)=O_t^2(s)$$

La différence d'image robuste est définie par :

$$D(t)=O_3(t)=\max(\text{seuillage}_\theta(\Omega_t^1(s)),\text{seuillage}_\theta(\Omega_t^2(s)))$$

La figure 6 montre un exemple d'application de cette technique sur la vidéo de l'action « marcher » de notre base.

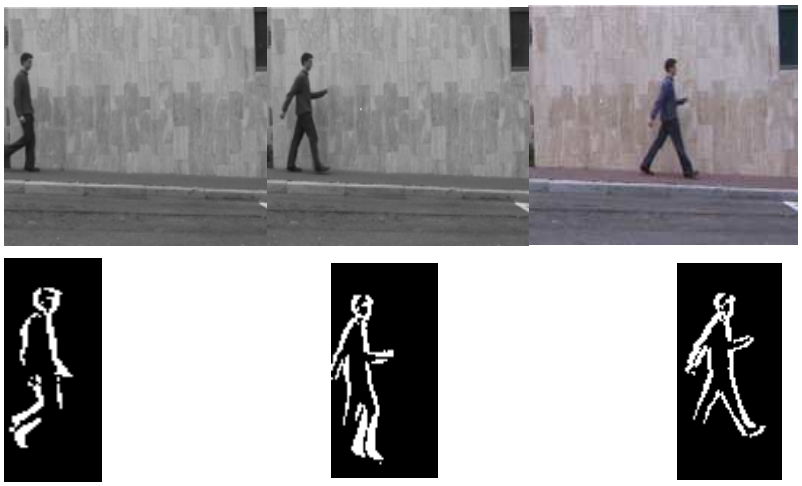


Figure 6: Segmentation des actions par la technique robuste

Il est possible de représenter les actions d'une séquence vidéo par une représentation statique, qu'on appelle, énergie du mouvement et historique du mouvement que nous présentons dans le paragraphe suivant.

Energie du mouvement (MEI) :

L'énergie du mouvement est essentiellement une image du mouvement cumulé. Elle indique l'emplacement spatial du mouvement. Elle est calculée comme suit :

$E_r(x,y,t)= 0$ si $B(x,y,t') = 0$, $t' \in \{t-r,\ldots\ldots\ldots\ldots,t\}$
1 sinon
r est le temps de capture de la séquence d'image .

Historique d'un mouvement (MHI):

Les caractéristiques temporelles du mouvement sont importantes pour l'analyse du mouvement. L'historique du mouvement caractérisant la séquence temporelle est défini par :

$H_r(x,y,t) = r$ si $B(x,y,t) = 1$
Max $(0, H_r(x,y,t-1)$ sinon

Le résultat est une fonction du mouvement de chaque pixel. La brillance d'un pixel est proportionnelle au changement de l'intensité, donc à la séquence du mouvement .

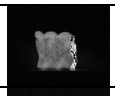
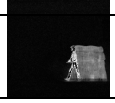
Action	Exemple	MHI (2D)
« jump »		
« pjump»		
« run »		
« side »		
« bend »		
« Walk »		
	a	b

Figure 7 : Action de bases et leur historique de mouvement

La figure 7b illustre la différence entre les actions. Il s'agit de l'historique des actions, qui est la projection 2d du volume. On peut remarquer, déjà qu'il y a des informations, telles que la durée d'une action, son rythme, et le sens d'une trajectoire qui sont importants. On peut espérer que la forme, la durée et le rythme d'une action soit capturé par le vecteur caractéristique du volume.

Choix des filtres morphologiques :

L'utilisation de la morphologie mathématique en détection et en estimation du mouvement est relativement récente. Elle a donne lieu a des développements intéressants pour les systèmes de détection, de poursuite et de reconnaissance d'objets, dans la mesure ou elle intègre naturellement des notions de plus haut niveau telle que la taille ou la forme des objets dans les traitements élémentaires. Les techniques morphologiques rencontrées dans la littérature peuvent être classées dans deux catégories majeures :

Les techniques qui font coopérer segmentation statique de type ligne de partage des eaux avec une analyse du mouvement. C'est le cas pour une segmentation topologique de l'image ou la topologie est contrainte par des plots représentant la position estimée ou prédite des objets.

Les techniques opérant par simplification hiérarchique de l'image en zones plates et calculant les attributs du mouvement sur ces zones plates et non sur les pixels.

L'utilisation d'éléments structurants temporels ou spatio-temporels permet d'aborder différemment la différentiation trame a trame. Elle permet à la fois de calculer des filtres spatio-temporels intéressants, et d'intégrer des changements temporels par accumulation (lorsque l'élément structurant est allongé dans l'axe temporel). Elle permet d'autre part de discriminer un déplacement donné en orientant l'élément structurant dans la direction correspondante.

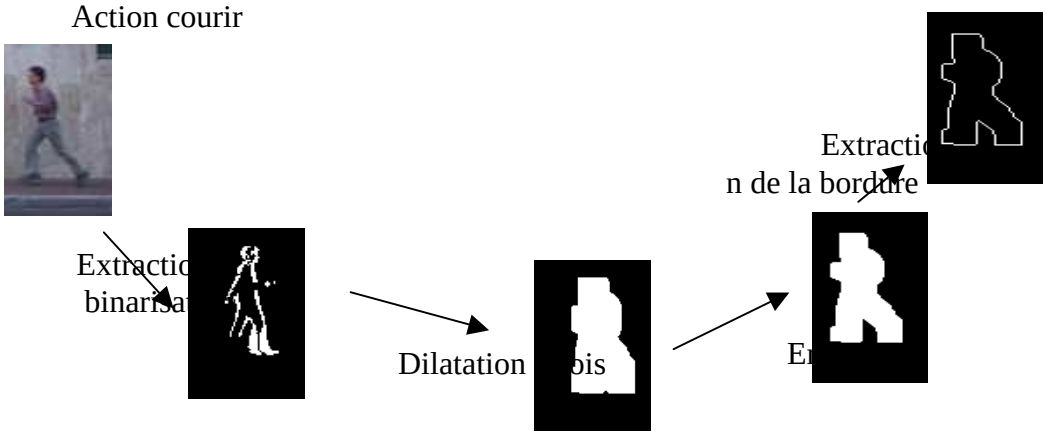


Figure 8: Extraction de la bordure par opération morphologique

Squelettisation :

C'est une manière de représenter une silhouette dans le but d'extraire les points saillant du contour. Il existe plusieurs techniques pour calculer le squelette d'une silhouette telle que la transformation de distance. Mais cette technique est très coûteuse. Nous avons extrait la silhouette (fig.8) comme proposé dans [Fujijoshi, 2004] : La silhouette est dilatée 2 fois pour éliminer les trous, puis une érosion est appliquée pour effacer toute anomalie. La figure 9 montre une extension de cette approche pour l'extraction d'une action 3D (volume) par la LPE.

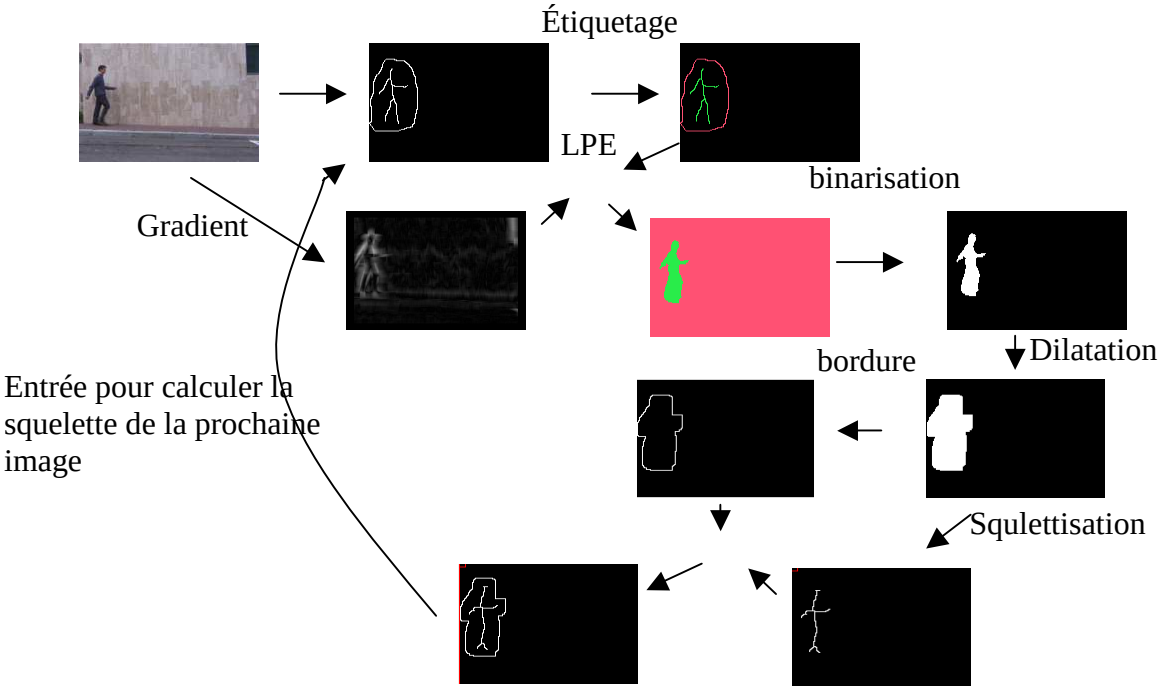


Figure 9: Technique de construction d'une squelette d'une action 3D

Dans la littérature [] il a été proposé une mesure basée sur la distance entre les maximums pour évaluer la similarité entre 2 silhouettes. Les points squelettiques sont sur le centre da la silhouette. Définissons SD comme l'ensemble qui contient les points du squelette de la silhouette détectée, et SM_i l'ensemble des points du squelette du modèle de l'action i. la mesure entre les 2 squelettes SD et SM_i est donnée par :

$$M_i = \sum_{pd \in SD} \min_{pm \in M_i} (|\ pd\ ,\ pm\ |)$$

Avec |.,.| est la distance euclidienne. L'action qui minimise cette mesure est choisie comme solution.

La projection horizontale et verticale :

Une autre manière de représenter une silhouette est ses projections horizontales et verticales [Haritaoglu, 1998 ; Panini, 2003 ; Boulay, 2005]. Une fois que nous avons la silhouette binaire de la personne nous la représentons par sa projection horizontale et verticale. La projection horizontale (verticale) sur l'axe de référence est obtenue en comptant

la quantité de pixels du mouvement correspondant à la personne détectée pour chaque rangée (colonne) de l'image. Ils projettent le modèle 3D sur une image pour chaque action de référence qui a été produite pour toutes les orientations possibles. Alors nous comparons la projection (H&V) de ces images avec la projection (H&V) de la silhouette détectée. Ils proposent une comparaison basée sur les secteurs non recouverts (équation 2 et 3) des projections (H&V) :

Ils définissent 2 rapports :

$$R_0(H) = \frac{\sum_{ir \in I_0} (H_{ir}^0 - H_{ir}^m)^2}{\sum_{ir} (H_{ir}^0)^2}$$

Ce qui représente la somme de la différence au carré des projections calculées sur l'intervalle I_0, normalisé par la somme des valeurs au carré de la projection horizontale de la personne détectée (H^d),

$$R_m(H) = \frac{\sum_{ir \in I_m} (H_{ir}^0 - H_{ir}^m)^2}{\sum_{ir} (H_{ir}^m)^2}$$

Ce qui représente la somme des différences au carrée des projections calculées sur l'intervalle I_0, normalisé par la somme des valeurs au carré de la projection horizontale du modèle produit (H^m). La distance entre la silhouette détectée S_{ild} et la silhouette modèle S_{ilm} est donnée par :

$$dist(S_{ilm}, S_{ild}) = \frac{1}{4} R_0(H) + R_m(H) + R_0(V) + R_m(V)$$

Cette distance appartient à l'intervalle [0,1] pour laquelle 0 correspond aux silhouettes semblables. Le modèle de l'action qui donne la distance minimum est choisi pour l'action da la personne étudiée.

1.3 Détection du mouvement par distribution gaussienne :

[Mokhber&al, 2005] ont utilisés les modèles adaptatifs de distributions gaussiennes multiples qui sont employés afin de modéliser des fonds complexes (multi-modaux). Chaque pixel est classé en fonction de la Gaussienne (modèle de fond) qui le représente le mieux. Ainsi, ces techniques permettent de suivre des objets en mouvement dans une scène complexe ayant plusieurs modes de mouvement. Les gaussiennes multiples peuvent permettre par exemple de détecter plusieurs objets avec des occlusions et de les différencier de leurs ombres. On peut aussi utiliser des gaussiennes sur chaque canal rouge, vert et bleu mais il devient souvent difficile de les différencier les unes des autres. D'une manière générale le point-clef de ce type de méthodes est la technique employée afin de séparer les gaussiennes. Le mélange de gaussiennes permet de conserver un historique de la variation d'intensité des pixels. A chaque instant t, toute l'information dont nous disposons sur un pixel particulier $\{x_0, y_0\}$ est son historique :

$$\{X_1,\ldots\ldots\ldots\ldots, X_t\} = \{I(x_0, y_0, i) : 1 \le i \le t$$

Où I est une séquence d'images.

L'historique courant de chaque pixel $\{X_1, \ldots, X_t\}$ est modélisée par un mélange de K distribution gaussienne. Cette probabilité d'observation de la valeur courante du pixel est donnée par :

$$P(X_t) = \sum_{i=1}^{k} \omega_{i,t} * n(X_t, u_{i,t}, \Sigma_{i,t})$$

Ou K est le nombre de distributions, $\omega_{i,t}$ est une estimation du poids (quelle portion de données est représentative pour cette gaussienne) de la i^{eme} gaussienne dans le mélange a l'instant t, $\mu_{i,t}$ est la valeur moyenne de la i^{eme} gaussienne dans le mélange à l'instant t, $\Sigma_{i,t}$ est la matrice de covariance de la i^{eme} gaussienne dans le mélange à l'instant t et ou η est la fonction de densité de la probabilité gaussienne :

$$n(X_t, u, \Sigma) = \frac{1}{(2\pi)^{\frac{1}{2}} |\Sigma|^{\frac{1}{2}}} e^{\frac{-1}{2}(X_t - u_t)^T \Sigma^{-1} (X_t - u_t)}$$

Pour chaque nouveau pixel **X**t d'une image à traiter, nous recherchons quelle est la gaussienne la plus proche. Si la distance entre cette gaussienne et le pixel courant est inférieure à un certain seuil, celui-ci est attribué au fond, sinon, il est classé comme un pixel appartenant à un objet en mouvement. Pour gérer les changements d'éclairage durant le processus d'acquisition, les pixels qui ont été attribués au fond sont utilisés pour mettre à jour l'image de référence et donc, les gaussiennes dont ils sont le plus proche avec :

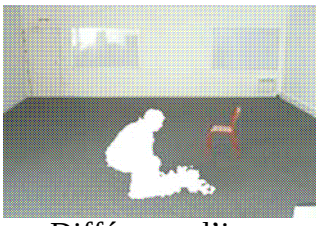
$$u_t = (1-\alpha) u_{t-1} + \alpha X_t$$

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

Où α a été empiriquement fixé à 0,1.

L'ombre est souvent détectée comme un objet en mouvement, ce qui altère fortement la forme des silhouettes détectées et perturbe donc l'algorithme de reconnaissance d'actions. Une deuxième étape est donc mise en place pour remédier à ce problème. Nous supposons, que l'ombre décroît la luminance des pixels mais n'affecte pas leur teinte. L'angle Φ entre le vecteur couleur du pixel courant $\mathbf{X}_t$ et celui du pixel de fond $\mathbf{B}_t$ correspondant (moyenne de la gaussienne la plus proche du pixel) est alors un bon paramètre pour détecter les ombres : si Φ est en dessous d'un certain seuil et si la luminance du pixel courrant est plus petite que la luminance du fond, nous considérons que le pixel est un pixel d'ombre.

A la fin du processus, seuls les pixels qui ont été détectés en mouvement par la mixture de gaussiennes et qui ne correspondent pas à un pixel d'ombre sont conservés. Plusieurs opérations de morphologie mathématique terminent cette étape et amènent, pour chaque image, à une carte binaire des pixels en mouvement.




Différence d'images Modélisation de l'ombre

Figure 10 : élimination de l'ombre


Figure 11 : détection de mouvement par distribution gaussienne

1.4 Détection du mouvement par graphe cuts :

Cette technique de détection est faite en collaboration avec maameri mourad qui travaille sur le sujet « Enveloppe visuelle d'objets dans une vidéo » proposé par Mr Chahir) . L'idée de base de *graph cuts* est de construire un graphe spécialisé correspondant à une fonction d'énergie à minimiser. Le but est de réaliser une coupe minimale sur le graphe associé qui est un problème d'optimisation combinatoire et résolu habituellement par les algorithmes de *max-flow*.

Une coupe sur un graphe **G** est définie comme un sous ensemble **C** d'arêtes de **E**(ensemble de tous les arêtes), tels qu'il y a aucun chemin de S vers T sans passer par, au moins, une des arrêtes de **C**. Une coupe divise les nœuds d'un graphe en deux partitions correspondants à une segmentation d'image en deux régions. En optimisation combinatoire le coût d'une coupe est défini comme la somme de poids d'arêtes de cette coupe. Le but est d'avoir une coupe de coût minimale(s-t mincut) qui peut correspondre à une segmentation optimale de l'image.

Ainsi, la technique de graphe cuts peut incorporer les contraintes topologique, par exemple les "contraintes dures " qui peuvent indiquer que certains germes d'image sont connus a priori comme une part d'"objets" ou du "fond".

Nous sommes dans un cadre de segmentation semi supervisé. La segmentation semi-supervisé ou semi-automatique est une manière pratique intéressante comme une alternative à la segmentation purement automatique, en considérant un compromis entre le temps d'exécution et la qualité de segmentation obtenue. Cet algorithme recalcule efficacement une segmentation optimale si certaines contraintes dures ajoutées ou supprimées, et ajuste une segmentation courante sans recalculer à nouveau la solution entière.

Séquences d'images des actions des vidéos de la base segmenter selon la technique du graphe cuts :

P = première image de la séquence vidéo

m = une des images du milieu
d = dernière image de la séquence vidéo

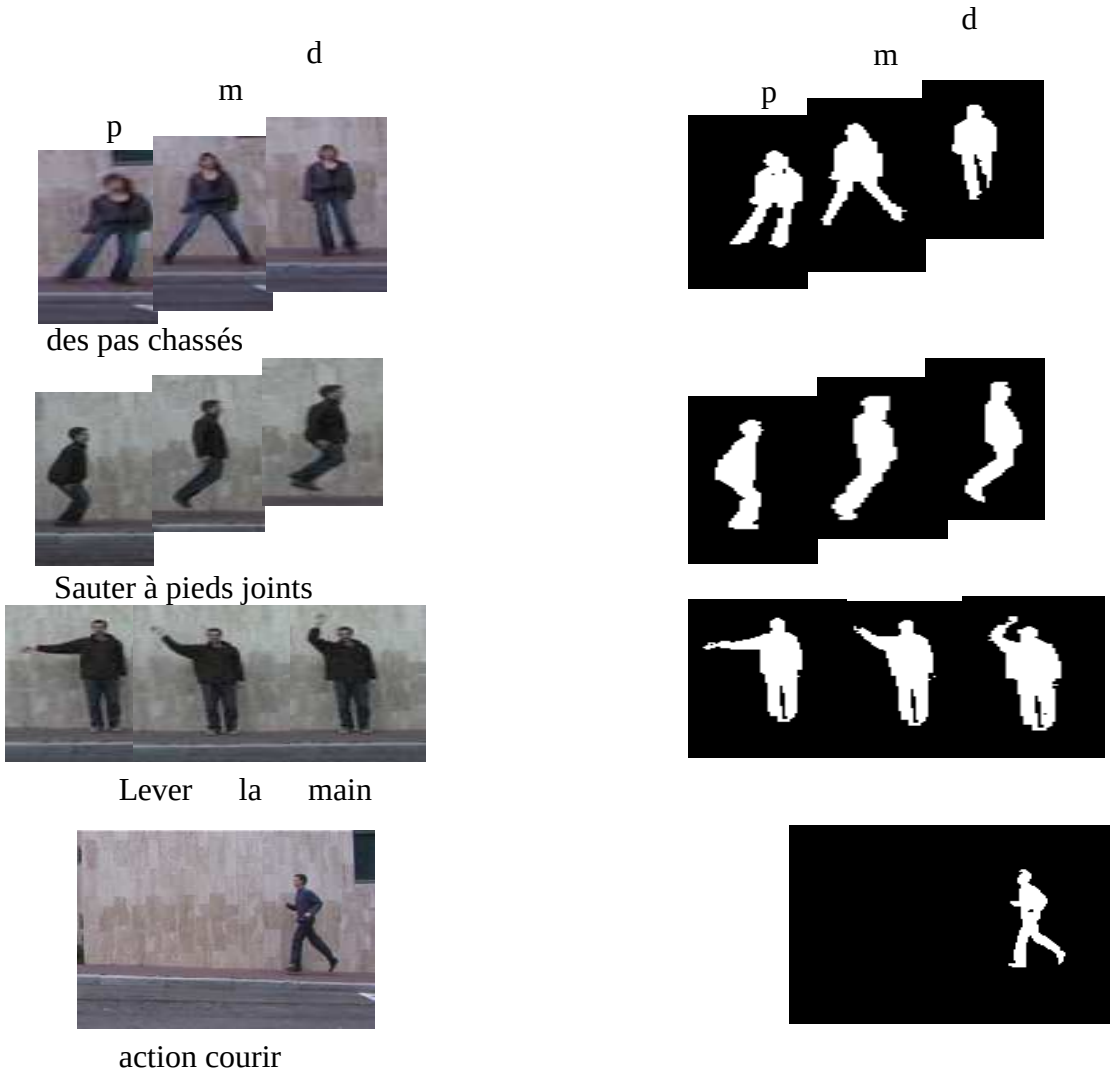


Figure 12: segmentation d'actions par graphe cuts

1.5 Reconnaissance de la forme par des moments statistiques :

Principe de la reconnaissance de formes :

Une opération de reconnaissance de formes se déroule suivant deux phases (figure 13) : une phase d'apprentissage et une phase de décision. Au cours de chacune de ces deux phases, on retrouve une phase d'extraction de paramètres (encore connue sous le nom de caractérisation). Cette phase permet d'extraire des paramètres représentatifs de l'image. Ces valeurs doivent présenter la particularité d'être invariantes à la rotation et à la translation.

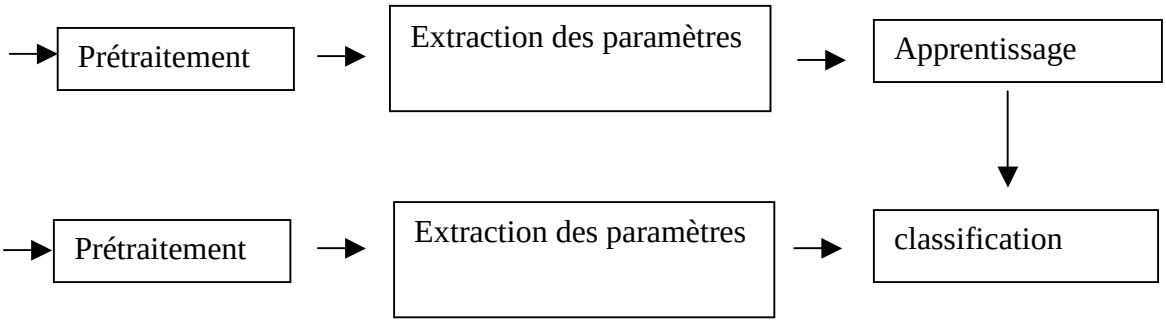


Figure 13 : principe de reconnaissance de formes

On distingue deux grandes approches comme cela est d'écrit par la figure suivante. L'approche contour qui caractérise la forme à partir de son contour sans tenir compte de la texture et l'approche globale qui étudie la forme dans son ensemble.

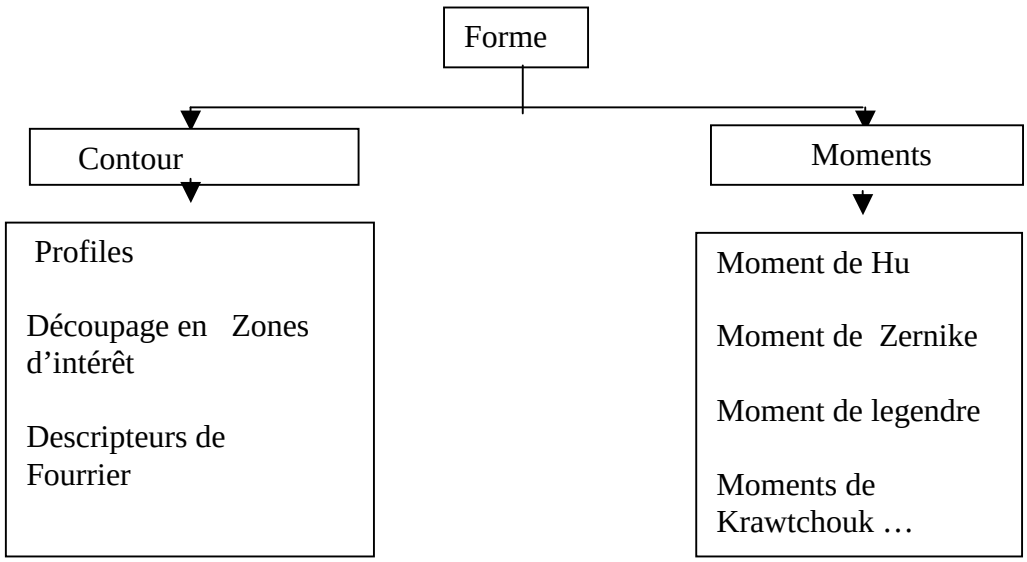


Figure 14 : différentes approches pour la phase de caractérisation

Dans ce qui suit, nous décrivons quelques types de représentation globales basées sur les moments : Les moments statistiques 2D tels que les moments de hu et Les moments géométriques 3D.

Représentation de la forme par des moments statistique 2D :

Moments de Hu :

La représentation des formes par des moments statistiques est une technique classique dans la littérature (Boblick et Pavis 2001). Ces moments sont basés sur les moments polynomiaux 2D :

$$m_{pq} = \iint x^p y^q P(x,y)\, dxdy$$

Où P = 1 si le pixel appartient à la silhouette
0 sinon

Afin de rendre les moments invariants à la translation, les moments sont centrés :

$$u_{pq} = \iint (x - \bar{x})^p (y - \bar{y})^q P(x, y) dxdy)$$

$$avec \quad \bar{x} = \frac{m_{10}}{m_{00}} \quad et \quad \bar{y} = \frac{m_{01}}{m_{00}} \quad , ou \quad m_{00} \ est \ la \ surface \ de \ l'objet$$

Pl

us loin les moments sont calculés de sorte qu'ils soient invariants à l'échelle :

$$n_{pq} = \frac{u_{pq}}{u_{00}^{\frac{p+q}{2}+1}}$$

Finalement pour ces moments invariants à la rotation, les 7 moments de hu sont calculés :

$$s_1 = n_{20} + n_{02}$$

$$s_2 = (n_{20} - n_{02})(n_{20} - n_{02}) + 4n_{11}n_{11}$$

$$s_3 = (n_{30} - 3n_{12})(n_{30} + 3n_{12}) + (n_{03} - 3n_{21})(n_{03} - 3n_{21})$$

$$s_4 = (n_{30} - n_{12})(n_{30} + n_{12}) + (n_{03} + n_{21})(n_{03} + n_{21})$$

$$s_5 = (n_{30} - 3n_{12})(n_{30} + n_{12})[(n_{30} + n_{12})(n_{30} + n_{12}) - 3(n_{03} + n_{21})(n_{03} + n_{21})] + (3n_{21} - n_{03})(n_{03} + n_{21})[3(n_{30} + n_{12})(n_{30} + n_{12}) - (n_{03} + n_{21})(n_{03} + n_{21})]$$

$$s_6 = (n_{20} - n_{02})[(n_{30} + n_{12}) - (n_{30} + n_{12}) - (n_{03} + n_{21})(n_{03} + n_{21}) + 4n_{11}(n_{30} + n_{12})(n_{03} + n_{21})]$$

$$s_7 = (3n_{21} - n_{03})(n_{30} + n_{12})[(n_{30} + n_{12})(n_{30} + n_{12}) - 3(n_{21} + n_{03})(n_{21} + n_{03})] - 3(n_{21} + n_{03})(n_{21} + n_{03})] - (n_{30} - 3n_{12})(n_{21} + n_{02})[3(n_{30} + n_{12})(n_{30} + n_{12}) - (n_{21} + n_{03})(n_{21} + n_{03})]$$

Moments de Zernike :

Les moments de Zernike sont définis par les équations : I, II, II. C'est donc le produit d'une fonction complexe : II par un polynôme radial : III. Ces moments définissent donc des valeurs complexes. Il faudra donc deux masques (réel et imaginaire) pour calculer les invariants de zernike.

$$Z_{pq} = \frac{(p+1)}{\pi(N-1)^2} \sum_{x=1}^{N} \sum_{y=1}^{N} V_{nm}^{*}(r,\theta) f(x,y) \text{..... ....} I$$

$$avec \quad V_{nm}^{*} = R_{nm}(r) \exp(im\theta) \text{..... ....} II$$

$$et \quad R_{nm}(r) = \sum_{s=0}^{\frac{(n-|m|}{2}} (-1)^{s} \frac{(n-s)!}{s!(\frac{n-2s+|m|}{2})!(\frac{n-2s+|m|}{2})!} r^{n-2s} \text{..... ..} III$$

Khotanzad a démontré que le module des moments de Zernike est invariant en rotation. Raveendran introduit des invariants en translation

composés par des combinaisons de moments de Zernike. Certains de ces

invariants sont présentés aux équations suivantes :

$$Z_{trans\ 00} = Z_{00}$$

$$Z_{trans\ 11} = Z_{11} - 2AZ_{00}$$

$$Z_{trans\ 20} = Z_{20} - 3A^{*}Z_{11} - 6AA^{*}Z_{00}$$

$$Z_{trans\ 22} = Z_{22} - 3AZ_{11} + 3A^{2}Z_{00}$$

<u>Moment de Legende</u> :

Les moments de Legendre sont définis à partir des polynômes du même nom. Ils sont définis dans le carré unité [-1,1]x[-1,1], ce qui oblige à normaliser l'objet dont on veut calculer ses moments. Le polynôme de Legendre d'ordre n est donné par:

$$\forall\, x \in [-1,1], \forall\, n \in N, P_n(x) = \frac{1}{2^n n!} \frac{d^n (x^2-1)^n}{dx^n}$$

Les polynômes de Legendre $\{P_n(x)\}$ forment une base complète et orthogonale sur le domaine de définition [-1,1]:

$$\forall\, x \in [-1,1]^2, \forall\, n \in (m,n) \in N^2, \int_{-1}^{1} P_m(x) P_n(y) dxdy = \frac{2}{2m+1} \delta_{mn}$$

δ_{mn} représente la fonction de Kronecker.

Les moments de Legendre d'ordre N sont donc donnés par:

$$\forall x \in [-1,1]^2, \forall n \in (p,q) \in N^2 / M = p+q,$$

$$L_{pq} = \frac{(2p+1)(2q+1)}{4} \int_{-1}^{1}\int_{-1}^{1} P_p(x) P_q(y) f(x,y) dxdy \qquad \text{……(**)}$$

Les moments de Legendre héritent de la propriété d'orthogonalité des polynômes de Legendre. Ainsi il n'existe plus de redondance de l'information véhiculée. A partir de l'équation**, on peut générer une infinité de moments de Legendre. Plusieurs études sur la reconnaissance des formes ont démontré que l'utilisation des moments de Legendre de bas ordre (jusqu'à l'ordre 3) est suffisante pour représenter la forme globale de l'entité donnée [Shen & Shen 1996].

Représentation d'une silhouette 3D avec des moments statistique :

A partir de toutes les images binaires obtenues, il faut extraire des caractéristiques représentatives de la séquence. On a opté pour une représentation globale des actions afin de simplifier le processus de reconnaissance et d'amener plus de robustesse lors de cette étape. Pour cela, une action est tout d'abord représentée par le volume 3D constitué par tous les points (x,y,t) détectés en mouvement. Ce volume spatio-temporel contient beaucoup d'informations dont la silhouette de la personne à chaque image et la dynamique de l'action : est ce que la forme s'agrandit au cours du temps, se déplace vers la gauche, etc. Pour caractériser ce volume, sans avoir à extraire (et séparer) les différentes informations présentes (posture, mouvement,…), ils utilisent les moments géométriques 3D.

Soit $\{x,y,t\}$ l'ensemble des points appartenant au volume binaire où x et y représentent les coordonnées spatiales et t, la coordonnée temporelle. Le moment d'ordre $(p+q+r)$ de ce volume est déterminé par :

$$A_{pqr} = E\{x^p y^q t^r\}$$

Où $E\{x\}$ représente l'espérance mathématique de x. Afin d'utiliser des caractéristiques invariantes en translation, ils utilisent les moments centrés définis par :

$$Ac_{pqr} = E\{(x-A_{100})^p (y-A_{010})^q (t-A_{001})^r\}$$

Ces moments doivent aussi être rendus invariants par rapport à l'échelle pour préserver une invariance à la distance de l'action ou à la taille des personnes. Une normalisation directe sur les différents axes, en divisant chaque composante par l'écart type correspondant n'est pas souhaitable car elle va amener une grosse perte d'informations quant à la forme des silhouettes binaires qui va s'arrondir. Aussi, une normalisation identique est effectuée sur les deux premiers axes, tandis que le troisième (le temps) sera normalisé séparément. La normalisation réalisée en conservant le ratio largeur/hauteur des silhouettes binaires est donc obtenue avec :

$$M_{pqr} = E\{\left(\frac{x-A_{100}}{Ac_{200}^{1/4} Ac_{020}^{1/4}}\right)^p \left(\frac{y-A_{010}}{Ac_{200}^{1/4} Ac_{020}^{1/4}}\right)^q \left(\frac{t-A_{001}}{Ac_{002}^{1/2}}\right)^r\}$$

Techniques de réduction de dimension

Les problèmes de réduction et d'analyse de données sont classiquement abordés par les techniques de sélection de variables et de réduction de dimension, qui visent à trouver des structures intrinsèques de dimension plus réduite cachée dans les observations dimensionnelles, dont l'analyse en composantes principales (ACP) est un exemple bien connu. Récemment, une nouvelle famille de méthodes est apparue, fondée sur des approximations globales de données Isomap, LLE, qui font l'objet de nombreuses recherches. La diffusion géométrique sur graphe généralise toutes ces méthodes et offre un cadre unifié pour la réduction, la visualisation, la catégorisation et la fusion de données de grande dimension. Nous avons récemment montré la pertinence de cette approche basée sur le processus de diffusion sur graphe dans le cadre d'applications sur la catégorisation d'expressions faciales, sur l'organisation et visualisation de grandes bases d'images et sur l'analyse d'images vectorielles représentant les textures.

1. Analyse en composante principale (ACP) :

L'ACP est une méthode mathématique d'analyse des données qui consiste à rechercher les directions de l'espace qui représentent le mieux les corrélations entre *n* variables aléatoires.

Dans la plupart des situations, on dispose de plusieurs observations sur chaque individu constituant la population d'étude. On a donc à prendre en compte *p* variables par individu, *p* étant strictement supérieur à 1. L'étude séparée de chacune de ces variables donne quelques informations mais est insuffisante car elle laisse de côté les liaisons entre elles, ce qui est pourtant souvent ce que l'on veut étudier. C'est le rôle de la statistique multifactorielle que d'analyser les données dans leur ensemble, en prenant en compte toutes les variables. L'Analyse en Composantes Principales est alors une bonne méthode pour étudier les données multidimensionnelles, lorsque toutes les variables observées sont de type numérique, de préférence dans les mêmes unités, et que l'on veut voir si il y a des liens entre ces variables. Dans la littérature, on trouve deux approches différentes de l'ACP :

- Elle peut être présentée comme la recherche d'un ensemble réduit de variables non corrélées, combinaisons linéaires des variables initiales résumant avec précision les données (approche anglo-saxonne).

- Une autre interprétation repose sur la représentation des données initiales à l'aide de nuage de points dans un espace géométrique. L'objectif est alors de trouver des sous-espaces (droite, plan,...) qui représentent au mieux le nuage initial. C'est cette dernière approche que nous aborderons par la suite.

1.1) Formulation mathématique de l'ACP :

On part d'un tableau de données rectangulaire, représentant toutes les données, en plaçant en ligne les individus et en colonnes les variables. Soit X ce tableau de données.

$$X = \begin{matrix} individus \\ \\ \\ \\ \\ \end{matrix} \begin{matrix} \\ \\ \\ \\ \\ \end{matrix} \begin{bmatrix} x_1^1 & \dots & \dots & x_1^p \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ x_1^n & \dots & \dots & x_p^n \end{bmatrix}$$

$$\text{var } iables$$

Par convention, on placera en exposant ce qui se rapporte aux individus : le premier individu est donc x^1, et en indice ce qui se rapporte aux variables : x_1 désigne la première variable. $X^1{}_1$ désigne la valeur numérique prise par la première variable, pour le premier individu. Et le tableau de données regroupe toutes les valeurs prises par tous les individus (de 1 à *n*) par *p* variables, soit encore :

$$X = (x_j^i)_{(1 \le i \le n, 1 \le j \le p)} \quad \text{et} \quad \forall (i, j), x_j^i \in R$$

1.2) Problème de l'A.C.P.:

Trouver le sous-espace vectoriel E_k de dimension *k* (*k*<*p* souvent *k*=2) tel que I_{Ek}, inertie du nuage *N* par rapport à l'espace E_k soit minimum.

Avec : $$I_{E_k} = \sum_{i=1}^{n} p_i d_M^2(x^i, E_k)$$

Et les notations suivantes :

p_i : pondération sur les individus. Chaque individu *i* est muni du poids p_i. La plupart du temps, on se place dans un cadre d'équipondération : tous les individus ont le même poids.

d_M : distance définie par la métrique *M*. En pratique, on considère deux métriques différentes : I, la métrique identité, ou la métrique D_{1/σ^2} qui réduit les variables. Réduire un tableau de données consiste à calculer l'écart type pour chacun des caractères et à exprimer toutes les cases en nombre d'écarts types (positif ou négatif). L'écart type devient ainsi une mesure unique commune à tous les caractères et les unités dans lesquels s'expriment initialement les données n'ont plus d'importance. $d_M(x^i, E_k)$ désigne la distance entre x^i et E_k soit la distance entre x^i et son projeté sur E_k.. $I_{\Delta u_1^\perp}$

On procède alors de la manière suivante :

- recherche d'un axe Δu_1 maximisant l'iner $I_{\Delta u_1^\perp}$
- …
- recherche d'un axe Δu_k, *M*-orthogonal à E_{k-1} maximisant l'inertie ;

On note *V* la matrice d'inertie du nuage *N*, qui est aussi la matrice de covariance des caractères (x_1,...,x_p).

La solution est alors obtenue en utilisant les propriétés spectrales des matrices : les vecteurs propres normés de la matrice *VM* ordonnés suivant les valeurs propres décroissantes fournissent les axes Δu_1, Δu_k, appelés axes factoriels. De plus, les inerties expliquées par ces axes sont égal : valeurs propres Δ_k. Les u_i forment une base *M*-orthonormée de E_k : les vecteurs u_i son définition normés et par ailleurs, la matrice *VM* étant symétrique, ses vecteurs propres sont orthogonaux.

1.3) Représentation des individus avec une ACP

Le problème initial était d'obtenir une représentation du nuage *N* dans des espaces de dimension réduit. On connaît maintenant les axes définissant ces espaces. Pour pouvoir obtenir les différentes représentations, il suffit de déterminer les coordonnées de la projection de tous les points du nuage sur chaque axe factoriel. Soit $c_1^i,\ldots,c_n^i$ ces *n* coordonnés pour l'axe *i*.

Le vecteur $c^i = \begin{pmatrix} c_1^i \\ \cdots \\ c_n^i \end{pmatrix}$ t appelé $i^{\text{ème}}$ composante principale.

On peut alors l'image " du nuage *N* dans un plan factoriel quelconque (u_i, u_j) grâce aux composantes principales c^i et c^j. La représentation dans le premier plan factoriel est obtenue grâce à c^1 et c^2. En utilisant conjointement la représentation du plan (u_1, u_3), on peut " voir " le nuage dans le sous-espace E_3.Le calcul des composantes principales se fait par changement de base. Il suffit de faire une projection orthogonale sur les nouveaux vecteurs de base. Ainsi, pour la $i^{\text{ème}}$ composante principale, on a :

$$\mathbf{c}^{\mathbf{i}} = (\mathbf{c}_{\mathbf{j}}^{\mathbf{i}})_{1\le j\le n} \quad \textbf{avec} \quad \mathbf{c}_{\mathbf{j}}^{\mathbf{i}} = M(u_i, x^j)$$

d'où l'expression de la composante principale : $c^i = XMu_i$

2. k plus proches voisins (k-ppv) :

La méthode des k plus proches voisins (k-ppv) est une approche non paramétrique qui ne fait appel a aucune hypothèse sur la répartition des différentes classes ou la nature des surfaces séparatrices. Le volume de la région r autour d'une forme xi est choisi de manière à contenir exactement k points de l'échantillon observe. On choisit généralement des volumes réguliers centres en xi (hyper cubes, hyper sphères). Les performances de la méthode dépendent de la valeur de k, le nombre des plus proches voisins. Elle consiste a rechercher pour un nouveau vecteur a classer le sous-ensemble des k plus proches vecteurs de l'ensemble d'apprentissage au sens d'une distance (distance euclidienne, métrique adaptative) qui détermine le type de voisinage, puis a affecter a ce vecteur la classe majoritairement représentée dans le sous-ensemble. Cette méthode est simple à implémenter avec aucun paramètre à fixer, mais néanmoins elle est très coûteuse en temps de calcul puisqu'il est nécessaire de calculer, à chaque nouvelle affectation,

3. Exploration du graphe par marches aléatoires :

Dans cette section, nous discutons de la méthode de représentation d'un ensemble de données $X = \{x_1, x_2, ..., x_N\}$ où $x_i \in R^n$ en termes de coordonnées de diffusion, et nous montrons le rapport entre le processus de diffusion sur graphe et les marches aléatoires sur l'ensemble de données.

La diffusion géométrique sur graphe offre un cadre unifié pour la réduction, la visualisation, la catégorisation et la fusion de données de grande dimension. Nous avons récemment montré la pertinence de cette approche basée sur le processus de diffusion sur graphe [Y. Chahir] dans le cadre d'applications sur la caractérisation de textures [Y. Chahir] et sur l'organisation et visualisation de grandes bases d'images et vidéos [Y. Chahir].

L'idée de base, issue de la théorie des graphes, est de représenter cette variété de données comme un graphe $G = (V, E)$ qui consiste en un ensemble fini de sommets, et un ensemble fini d'arêtes E inclues dans VxV. Deux sommet v_i et v_j sont adjacents si l'arête $(v_i, v_j) \in E$. Nous considérons des graphes connexes non orientés. Un graphe est considéré comme un graphe pondéré, si on peut lui associer une fonction de poids $w : V \times V \to R^+$ qui satisfait les conditions suivantes pour chaque sommet $v_i, v_j \in V$:

(a) $w(v_i, v_j) = w(v_j, v_i)$

(b) $w(v_i, v_j) \geq 0$

Cette fonction de poids reflète le degré de similarité entre deux sommets du graphe et décrit ainsi l'interaction du premier ordre entre les sommets du graphe. Son choix dépend généralement de l'application considérée. Ainsi, la matrice des poids obtenus, désormais appelée matrice d'affinité ou matrice de similarité, W est définie par : $(W)_{ij} = w(v_i, v_j)$ si les sommets v_i et v_j sont adjacents et $w(v_i, v_j) = 0$ dans le cas contraire. Les sommets étant liés à eux-mêmes nous avons pour tout sommet v_i $w(v_i, v_i) = 1$.

Le degré d'un sommet v_i est défini par :

$$d(v_i) = \sum_{v_j \in V} w(v_i, v_j).$$

On définit la matrice diagonale des degrés des sommets D avec :

$(D)_{ii} = D(v_i, v_i) = d(v_i)$ et

$D(v_i, v_j) = 0$ pour $v_i \neq v_j$

On définit la matrice L telle que :

$$(L)_{ij} = L(v_i, v_i) = \begin{cases} d(v_i) - w(v_i, v_i) & \text{si } v_i = v_i \\ -w(v_i, v_j) & si\, v_i et v_i \text{ sont adjacents} \\ 0 & \text{sinon} \end{cases}$$

Ainsi, le Laplacien du graphe G peut être défini par :

$\Im = D^{-1/2} L D^{-1/2}$ où $(D^{-1})_{ii} \equiv 0$ si $d(v_i) = 0$

Nous allons nous intéresser à un processus de marche aléatoire (ou de diffusion dans le graphe G). Le temps est discrétisé $t = (0,1,2,...)$. A chaque instant, un marcheur est localisé

sur un sommet et se déplace à l'instant suivant vers un sommet choisi aléatoirement et uniformément parmi les sommets voisins. La suite des sommets visités est alors une marche aléatoire, et la

Probabilité de transition du sommet v_i au sommet v_j est à chaque étape :

$$p(v_i, v_j) = \frac{w(v_i, v_j)}{d(v_i)}.$$

Ceci définit la matrice de transition P , (P_{ij}) de la chaîne de Markov correspondant à la marche aléatoire. La matrice P est stochastique, en effet : $\forall v_i, \forall v_j, 0 \le p(v_i, v_j) \le 1$ et $\sum_{v_j \in V} p(v_i, v_j) = 1$. Nous pouvons aussi écrire $P=D^{-1}W$.

Considérons $p_t(v_i, v_j)$ le noyau correspondant à la t ème puissance de P : P^t . $p_t(v_i, v_j)$ peut être interprété comme la probabilité pour un marcheur d'atteindre le sommet v_j en partant du sommet v_i en t étapes [Brandon] .

L'intérêt d'introduire cette matrice de transition est que l'exploration du graphe par la marche aléatoire qu'elle engendre permet de déterminer des propriétés topologiques du graphe [S. Ingve], reliées aux propriétés spectrales de P .

Dans le cas de graphes non pondérés, nous avons $P = D^{-1/2}(I - \Im)D^{-1/2}$ [F. Chung].

Pour notre problématique, l'ensemble d'origine $X = \{x_1, x_2, \ldots, x_N\}$ est considéré comme l'ensemble de sommets du graphe pondéré avec w(·,·). Le but étant de représenter chaque $x_i \in R^d$ par un point $y_i \in R^m$ où m <<d, de sorte que l'ensemble $Y = \{y_1, y_2, .., y_N\}$ capture toute l'information géométrique intrinsèque de l'ensemble d'origine. Pour cela, Belkin et Niyogi [1] ont montré que l'utilisation de l'opérateur de Laplace Beltrami, constitue un "bon" choix pour la prise en compte de l'information géométrique.

En particulier, l'information locale est bien préservée par l'utilisation de ces fonctions propres (eigenmaps). Il y a un lien entre le Laplacian d'un graphe et l'opérateur de Laplace Beltrami, Δ, sur les variétés.

Lafon, Keller et Coifmann ont montré que l'utilisation d'un noyau gaussien w_ε produit une représentation des données qui est fortement corrélée à la distribution des échantillons des données. Il est définit comme suit :

$$w_\varepsilon(x_i, \mathbf{x}_j) = \mathbf{exp}\left(-\left\|x_i - x_j\right\|^2/\varepsilon\right)$$

où ε est un paramètre d'échelle et $\| \ \|$désigne la distance euclidienne standard.

Nous présentons, dans le tableau 1, notre approche de catégorisation qui est basée sur la diffusion par marches aléatoires sur graphe.

Tab. 1 : Exploration par marches aléatoires

Entrée : $X = \{x_1, x_2, ..., x_N\} \subset R^d, t, \varepsilon, m$

Sortie : $Y = \{y_1, y_2, ..., y_N\} \subset R^m$

Construction de la matrice de similarité

$w_\varepsilon(x_i, \mathbf{x}_j) = \mathbf{exp}\left(-\left\|x_i - x_j\right\|^2/\varepsilon\right)$

Normalisation en utilisant l'opérateur de Laplace-Beltrami

$$\tilde{w}_\varepsilon(x_i, \mathbf{x}_j) = \frac{w_\varepsilon(x_i, \mathbf{x}_j)}{q_\varepsilon(x_i) q_\varepsilon(x_j)}$$

Matrice de transition

$$p(x_i, \mathbf{x}_j) = \frac{\tilde{w}_\varepsilon(x_i, \mathbf{x}_j)}{\sqrt{\tilde{q}_\varepsilon(x_i)\tilde{q}_\varepsilon(x_j)}}$$

avec $q_\varepsilon(x_i) = \sum_{x_k \in X} w_\varepsilon(x_i, \mathbf{x}_k)$

Diagonalisation de la matrice P

Espace de diffusion

$$x \to y = (\lambda_1^{t/2}\varphi_1(x), \cdots, \lambda_{m(}^{t/2}\varphi_{m(}(x))^T$$

La première normalisation de la matrice de similarité permet de trouver une représentation indépendante de la distribution.

La décomposition spectrale de la matrice de transition P donne un ensemble de valeurs propres $1 = |\lambda_0| \geq |\lambda_1| \geq |\lambda_2| \geq ... \geq 0$ engendrant un ensemble de vecteurs propres $\{\varphi_0, \varphi_1, \varphi_2, ..., \varphi_m\}$, solutions de:

$$P\varphi_m = \lambda^t_m \varphi_m$$

Ainsi, on peut définir la famille des distances de diffusion $\{D_t\}_{t \geq 1}$ par :

$$D_t^2(x, y) = \sum_{j \geq 0} \lambda_j^t (\varphi_j(x) - \varphi_j(y))^2 \qquad (1)$$

où le paramètre d'échelle t contrôle la sensibilité de la distance de diffusion D_t aux valeurs propres φ_j .

$D_t(x,y)$ mesure le taux de connectivité entre les données x et y par les chemins de longueur t .

Considérons la transformation suivante $\{\psi_t\}_{t \geq 1}$:

$\psi_t : R^n \to R^{m(t)}$

$$x \to \psi_t(x) = (\lambda_0^{t/2}\varphi_0(x), \lambda_1^{t/2}\varphi_1(x), \lambda_2^{t/2}\varphi_2(x), \cdots, \lambda_{m(t)}^{t/2}\varphi_{m(t)}(x))^T$$

avec φ_0 qui est un vecteur constant $\varphi_0(x) = (1, 1, ..., 1)$.

Cette transformation est communément utilisée maintenant pour l'analyse et la réduction de donnée de grande dimension [Stéphane Lafon]. Elle permet donc de passer d'un espace de mesure de dimension n à un espace de représentation homogène de dimension $m(t)$ plus réduit représentant toutes les informations ainsi que les propriétés structurales du graphe. $m(t)$ est le nombre pour lequel les valeurs propres $\{\lambda_j^{t/2}\}_{j \geq m(t)}$ sont numériquement insignifiants. On a généralement $m(t) \leq 3$. Ces informations sont principalement captées par les premiers vecteurs propres de P^t liés aux plus grandes valeurs propres. La distance de diffusion (1) peut être définie par :

$$D_t^2(x,y) = \|\psi_t(x) - \psi_t(y)\|$$

Les vecteurs propres de la matrice de transition P_t peuvent être interprétés comme la généralisation des fonctions de Fourrier sur un graphe. Ainsi les valeurs propres de faibles valeurs correspondent aux vecteurs propres de hautes fréquences, et celles de fortes valeurs correspondent aux vecteurs de basses fréquences.

Notons, que le second vecteur propre φ_1 est connu comme le vecteur de Fiedler et peut être utilisé pour ordonner l'ensemble des données X .

3.1) catégorisation des actions par un processus de marche aléatoire :

Dans le cadre d'une vidéo, il s'agit de détecter des groupes d'images proches selon une mesure de similarité donnée qui va favoriser des regroupements qu'on peut classer en plans ou en scènes. Une première application consiste à détecter les plans (ou cuts) qui généralement représentent des prises de caméra continues et une constance de luminosité. Ensuite, les plans qui partagent la même sémantique peuvent être groupés en scènes. Ici, on suppose que la sémantique des plans vidéo est suffisamment bien exprimée par les caractéristiques de bas niveau tel que les informations de couleur, ou des informations topologiques tel que le mouvement [F. Chung]. Dans le cadre d'un corpus de vidéos, il s'agit de classer des vidéos entières dans des catégories définies. Pour mettre en évidence le cadre unifié de notre approche pour la structuration, visualisation et catégorisation des ces objets à savoir : une image, un plan et une scène pour une vidéo donnée, nous avons construit un graphe complet entre les images d'une vidéo où la mesure de similarité est définie par : $w(x_i, \mathbf{x}_j) = 1$ si $x_i, \mathbf{x}_j$ appartiennent à la même classe, sinon $w(x_i, \mathbf{x}_j) = 0$. La décomposition spectrale de la matrice de transition P détermine les valeurs propres et les vecteurs propres de la matrice: $\{\lambda_1^t \psi_1, \boldsymbol{\lambda}_2^t \psi_2, .., \boldsymbol{\lambda}_i^t \psi_i ...\}$.

3.2) application pour la réduction et la réorganisation des données :

Nous présentons dans cette section plusieurs applications qui ont été mise en œuvre. La première est liée à l'organisation et la visualisation d'images où chaque élément de la base de données est représenté comme un vecteur qui représente tous les pixels de l'image. Pour construire le graphe, nous avons utiliséela mesure de similarité suivante : $w(x_i, \mathbf{x}_j) = \mathbf{exp}(-\|x_i - x_j\|^2)/\varepsilon)$ où $\| \|$ désigne la métrique euclidienne standard et ε le paramètre d'échelle . Nous construisons un graphe complet et la matrice de transition P comme décrit dans la section précédente. A partir de la décomposition spectrale de P nous

utilisons les coordonnées $(\lambda_1\psi_1, \lambda_2\psi_2)$ pour la visualisation 2D. Dans nos expériences, pour fixer le paramètre ε nous utilisons la stratégie proposée par Lafon .

$$\varepsilon = \frac{1}{N}\sum_{i=1}^{N} \min\left\{ \left\| x_i - x_j \right\|^2 / \left\| x_i - x_j \right\|^2 > 0,\ j = 1,2,...N \right.$$

Ré-Organisation et visualisation d'une même base d'actions:

Dans cet exemple, les données sont des ensembles d'images d'une personne qui marche qu'on cherche à ordonner. Chaque image couleur a une résolution de 180 x 144 pixels, c'est-à-dire un vecteur de données de dimension 25920*3. L'opérateur de diffusion utilisé est Laplace-Beltrami de pas t=1.

Figure 15: Réorganisation d'un ensemble d'images d'une action

On reconnaît dans cette figure l'efficacité du processus de diffusion pour la révélation géométrique de l'ordre d'une personne qui marche et permet de mieux capter les petites variations d'intensité entre chaque paire de frames.

Expérimentation et résultats

Une action peut être exécuté de la gauche vers la droite ou de la droite vers la gauche par des personnes différentes.

- (1) marcher (walk)
- (2) courir (run)
- (3) sauter sur les 2 pieds en se déplacent (jump)
- (4) sauter sur les 2 pieds sans se déplacer (jump in place)
- (5) toucher le sol avec la main droite et se remettre debout (bend)
- (6) lever la main gauche sans se déplacer (one-hand wave)
- (7) lever les 2 mains sans se déplacer (two-hands wave)
- (8) mouvement a pas chasse (gallop sideways)

De l'historique des actions on peut remarquer, qu'il y a des informations, telles que la durée d'une action, son rythme, et le sens d'une trajectoire qui sont importants. On peut espérer que la forme, la durée et le rythme d'une action soit capturé par le vecteur caractéristique du volume. Par ailleurs, la durée des actions est variable dans le corpus. Mais, pour une meilleure cohérence des résultats, nous avons pris la durée minimale commune de toutes les actions qui est de l'ordre de 27. La figure3.2 et la figure3.5, montrent un premier résultat de l'utilisation des marches aléatoires sur graphe. Il s'agit de la réorganisation des historiques de mouvement en triant le 1^{er} vecteur propre dans l'espace de diffusion, en utilisant la différence de pixel. Dans cet exemple, pour l'affichage nous nous sommes limités aux 4 classes parmi les 8 actions. On peut remarquer que les 8 actions, des personnes, de même nature se suivent et donc peuvent être classés à part. En fait, il s'agit des classes : « bend », « pjump » ,« wave1 » et « wave2 ».

Les actions sont classées de gauche à droite, de haut en bas. Cependant, il y a un problème, signalé par *, qui Le trie du vecteur propre nous donne une idée de la représentation et des cluster former par les données. Dans la figure suivante, on présente le résultat de tri du premier vecteur propre, selon une mesure de similarité basée sur la différence de pixels.

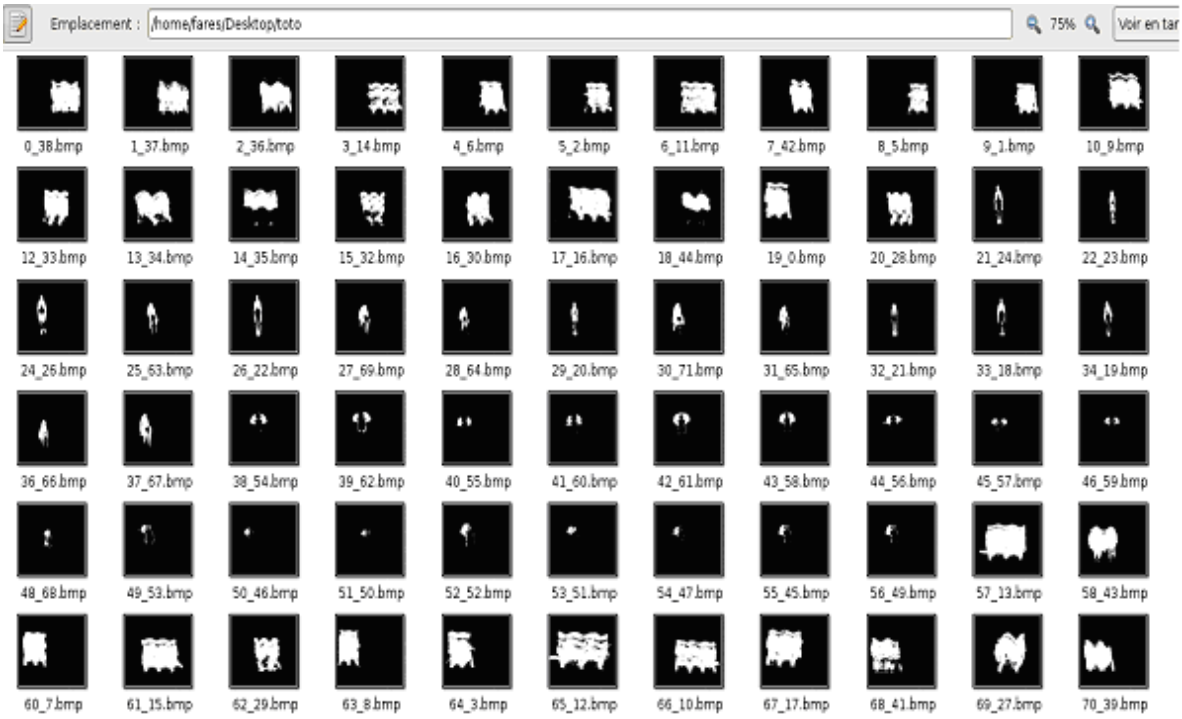


Figure 16: trie du 1er vecteur propre selon une différence de pixels

D'après le premier vecteur propre on remarque trois classes :

* Une première classe des personnes qui se déplacent de la droite vers la gauche indépendamment de leur mouvement. (images 0-20)

* Une 2eme classe des personnes qui font des mouvements sans se déplacer, qui sont d'ailleurs bien classifiés, càd que chaque action est plus au moins reconnue, on remarque juste une confusion entre le mouvement 4 et 5. Par contre le mouvement 7 (image 38 à 46) et 8 (images 46 à 56) sont reconnues à 100% .

* Une 3eme classe qui regroupe des personnes qui se déplacent de la gauche vers la droite.

Nous avons effectué le même travail (figure 17) , mais en comparant les surfaces générées par l'historique des actions.

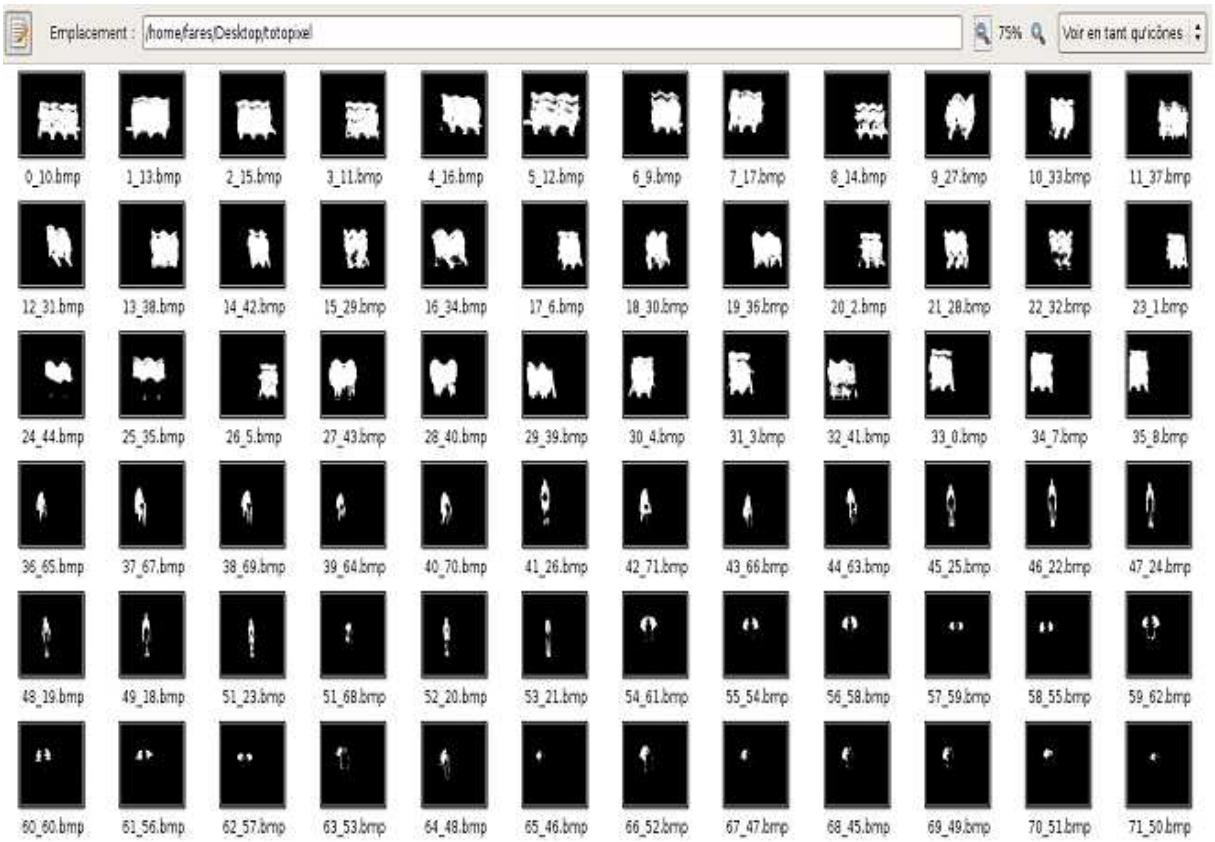


Figure 17: Réorganisation selon les surfaces des historiques

A première vue des résultats, on remarque deux classes :

* une première classe, regroupe des personnes qui se déplacent en effectuant le même mouvement.

* une 2eme classe, regroupe des personnes qui font des mouvements sans se déplacer.

En détaillant le vecteur propre, on remarque que l'action 2 (courir) a été détecté a 100% cela est du a la particularité de l'action, qui est dense. Par contre les autres actions sont confuses. En ce qui concerne les personnes qui effectue le mouvement sans se déplacer, l'action 7 et 8 sont reconnu a 100% , on remarque juste une confusion entre le mouvement 4 et 5.

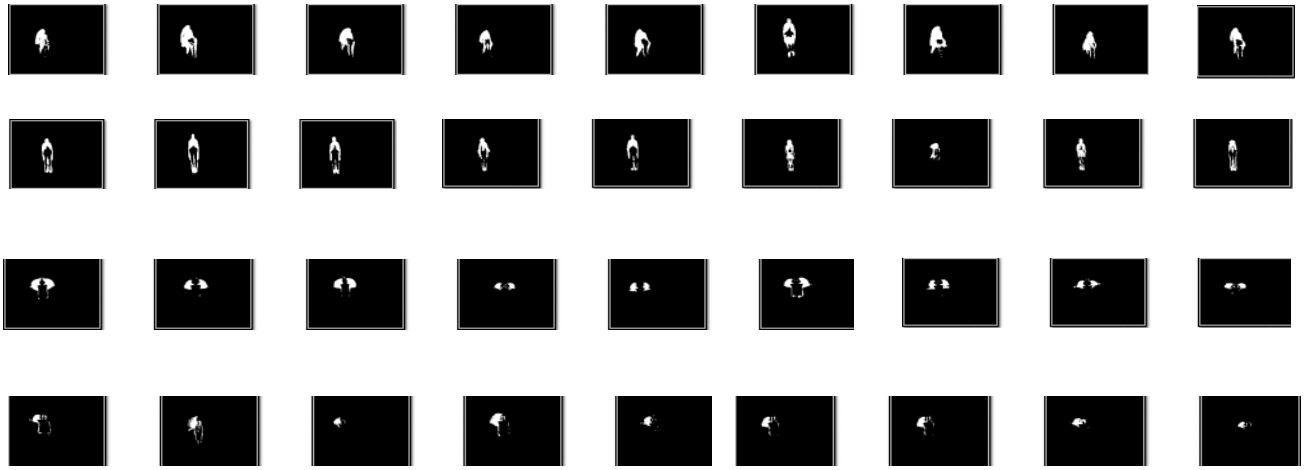


Figure 18: exemples des bonnes reconnaissances d'action de la base

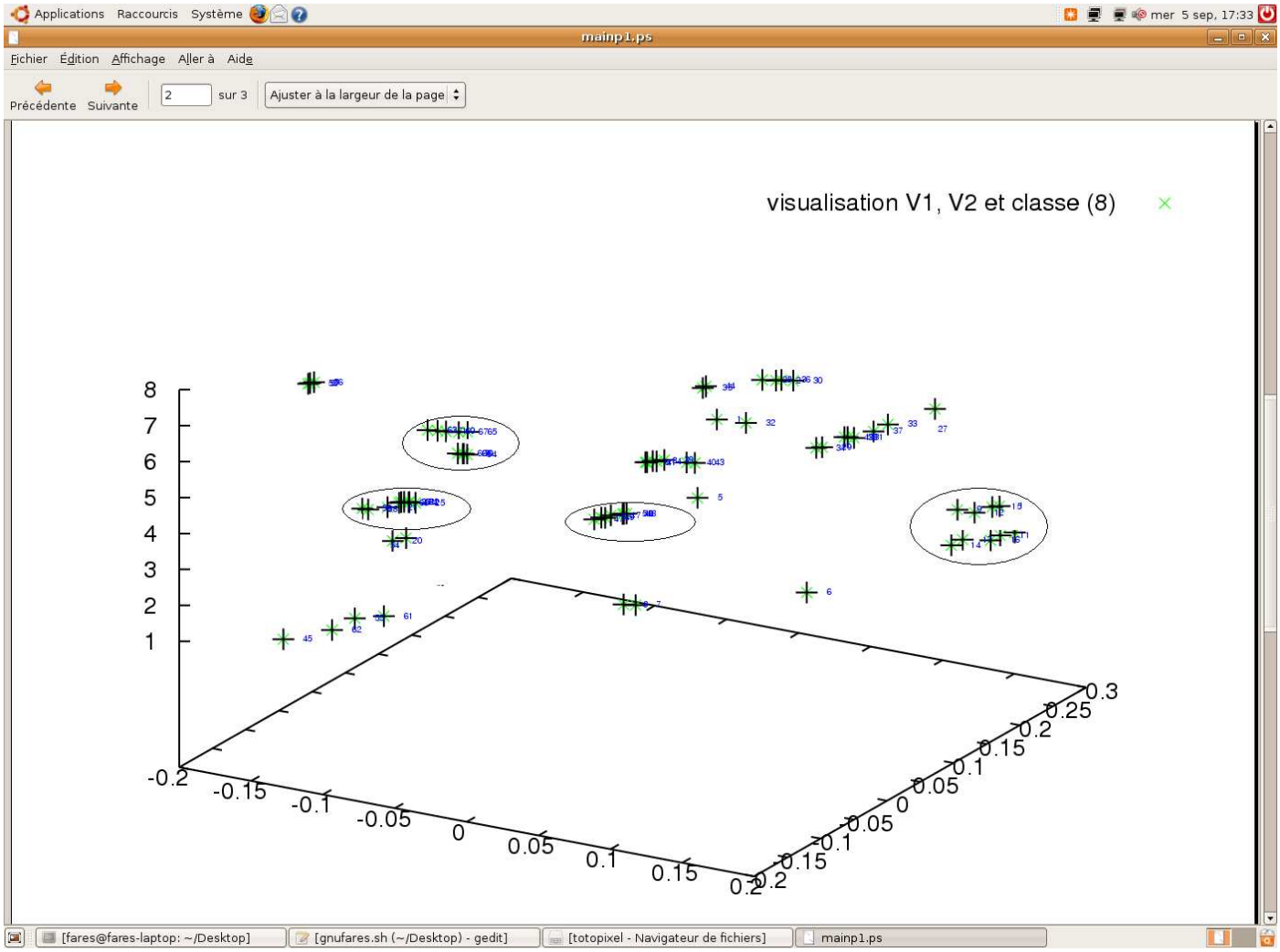


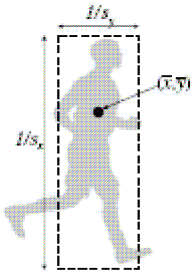
Figure 19: montre le résultat de la catégorisation des actions en utilisant la classification par k-moyennes en 8 classes, sur les deux projections: $(\lambda_1^t \quad \psi_1, \lambda_2^t \quad \psi_2)$.

C1, C2,C3,C4 sont les classes ou le taux de reconnaissance est bon qui sont respectivement bend, pjump, wave1, wave2. les autres mouvement sont confus.

Afin d'améliorer ces résultat nous avons opté pour une autre approche qui consiste a détecté les mouvements directement du volume 3D, pour cela nous
avons utilise les moments géométriques 3D.

Lors de la reconnaissance, la base est composée de 8 actions réalisées chacune par 9 personnes. La reconnaissance est réalisée en cherchant pour chaque vecteur caractéristique d'une action de la base, un vecteur plus proche dans la même base, au sens de la distance euclidienne. Ce qui revient à construire une matrice de distance entre toutes les vidéos de la base. Ensuite un kmeans est utilise (méthode de 1 plus proche voisin) afin de regrouper les distances les plus proches. Nous utilisons un vecteur de caractéristique M compose des moments d'ordre 2 et 3, soit 14 moments:

M={M200, M011, M101, M110, M300, M030, M003, M210, M201, M120, M021, M102, M012, M111}.



Les 14 moments de action courir (2) de la base :

1.31474 0.0667942 1.10797 0.0110768 0.000652416 -0.000166253 0.000220814 0.000700451 0.000547358 6.48052e-05 8.47032e-05 0.000374271 6.14959e-05 0.000244231 1.00002

Sur le tableau suivant, sont présentés les taux de reconnaissance obtenus lors de la comparaison des distances. Chaque moments d'une personne est compare avec tous les moments de la base, indiquant ainsi les 8 moments les plus proche de lui. Travail répété sur tous les moments de la base:

	1	2	3	4	5	6	7	8	9
1Walk	50	37.5	37.5	50	50	37.5	37.5	50	50
2Run	62.5	62.5	62.5	25	25	62.5	62.5	62.5	25
3Jump	25	25	50	50	50	50	25	50	50
4Pjump	100	100	88.88	100	88.88	100	100	88.88	100
5Bend	100	88.88	100	100	88.88	100	100	100	100
6Wave1	88.88	100	100	88.88	88.88	88.88	100	88.88	88.88
7Wave2	100	100	100	100	100	88.88	88.88	100	100
8Side	50	50	50	37.5	37.5	37.5	50	37.5	50

Tableau2: taux de reconnaissance obtenus par comparaison des distances

Les taux de reconnaissance moyens sur les 8 actions effectuées chacune par 9 personnes varient de 44,44 a 97,52. Nous constatons que les taux de reconnaissance moyens des personnes qui font des mouvements en ce déplaçant (action 1,2,3,8), est faible par rapport aux personnes qui font des mouvements sans se déplacer (action 4,5,6,7). On peut donc conclure que les actions effectuées par des personnes qui font des mouvements sans se déplacer sont bien reconnues. Nous avons constate aussi que le mouvement des personnes effectuant l'action de la gauche ver l'adroite ne sont pas classe de la même manière que les personnes effectuant l'action de l'adroite ver la gauche, de même mouvement, c'est ce qui explique le taux de reconnaissance faible dans les action 4,5,6,7 et provoque des confusions avec d'autre mouvements effectuer dans la même direction. Nous présentons la matrice de confusion moyenne, obtenue sur les 8 actions:

	1walk	2run	3jump	4pjump	5bend	6wave1	7wave2	8side
1Walk	44.44	0	34.72	0	0	0	0	20.84
2Run	0	50	20.32	0	0	0	0	29.69
3Jump	0	4.18	41.66	0	0	0	0	54.16
4Pjump	0	0	0	96.29	3.71	0	0	0
5Bend	0	0	0	0	97.52	0	2.48	0
6Wave1	0	0	0	0	0	95.58	4.42	0
7Wave2	0	0	0	0	2.48	0	97.52	0
8Side	0	19.45	36.11	0	0	0	0	44.44

Tableau3: matrice de confusion


L’action marcher exécutée par 2 personnes différentes dans des directions différentes.


L’action a pas chassé exécutée par 2 personnes différentes dans des directions différentes.


L’action exécutée par 2 personnes différentes dans des directions différentes.

Figure 20: même action effectuée dans des direction différentes

Par exemple, l'action marcher (1), elle est exécuté par 9 personnes différentes, 5 personnes parmi les 9 marchent de la gauche ver l'adroite, et les 4 autres de l'adroite ver la gauche.

On comparant les moments de la première personne qui marche avec les moments de toutes la base. Il a classifie comme étant plus proche de 4 personnes qui marche et ses personnes ont la même direction que lui, par contre il y a eu confusion avec 3 personnes de l'action 3 et une personne de l'action 8, qu’ont toujours la même direction du mouvement que lui. Il n’a pas reconnue les 4 personnes qui marchent de l'adroite ver la gauche. Même constat fait pour toutes les autres actions.

Conclusion et perspectives

Nous avons présenté dans ce travail le processus de détection, de reconnaissance et de représentation des données liées actions humaines tel que « sauter », « marcher »… .

Nous avons dans un premier temps présenté les méthodes de détection des personnes en action. Ensuite, nous avons résumé chaque action par une image représentative qui illustre l'action spatio-temporelle.

Dans une seconde partie, nous avons étudié l'opérateur de caractérisation volumique de forme qui sont les moments géométrique 3D, afin de caractériser chaque action.

Enfin, nous avons présenté le processus d'exploration du graphe des actions par marches aléatoires, lié aux méthodes spectrales afin de réduire la dimension de représentation des données. Ce processus de diffusion fournit une structure de représentation de données adéquate.

Pour conclure sur l'expérience que j'ai pu acquérir durant mon stage, je peux dire que j'ai appris à travailler de manière autonome. Cela m'a permis comprendre et de mettre en pratique des notions d'imageries et d'analyse des données vus en cours . J'ai aussi découvert d'autres applications d'imageries vidéos et cela m'a beaucoup intéressé.