HAL
open science

Action Representation and Recognition

Daniel Weinland

To cite this version:

Daniel Weinland. Action Representation and Recognition. Other [cs.OH]. Institut National Polytechnique de Grenoble - INPG, 2008. English. NNT : . tel-00379318

HAL Id: tel-00379318

https://theses.hal.science/tel-00379318v1

Submitted on 28 Apr 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

Numéro attribué par la bibliothèque

THÈSE

pour obtenir le grade de

DOCTEUR DE L'INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

Spécialité : Imagerie, Vision et Robotique
Ecole Doctorale : Mathématiques, Sciences et Technologie de l'Information

présentée et soutenue publiquement
par

Daniel WEINLAND

le 20 Octobre 2008

Action Representation and Recognition

Directeurs de thèse : Radu HORAUD, Edmond BOYER et Rémi RONFARD

JURY

M. James CROWLEY	Président
M. David FORSYTH	Rapporteur
M. Jean PONCE	Rapporteur
M. Radu HORAUD	Directeur de thèse
M. Edmond BOYER	Directeur de thèse
M. Rémi RONFARD	Directeur de thèse

Thèse préparée dans le laboratoire Jean Kuntzmann au sein du projet Perception, INRIA Rhône-Alpes, 655 avenue de l'Europe, 38330 Montbonnot, France.

Abstract

Recognizing human actions is an important and challenging topic in computer vision, with many important applications including video surveillance, video indexing and understanding of social interaction. From a computational perspective, actions can be defined as four-dimensional patterns, in space and in time. Such patterns can be modeled using several representations which differ from each other with respect to, among others, the visual information used, e.g. shape or appearance, the representation of dynamics, e.g. implicit or explicit, and the amount of invariance that the representation exhibits, e.g. a viewpoint invariance allowing to learn and recognize using different camera configurations.

Our goal in this thesis is to develop a set of new techniques for action recognition. In the first part we present ”Motion History Volumes”, a free-viewpoint representation for human actions based on 3D visual-hull reconstructions computed form multiple calibrated, and background-subtracted, video cameras. Results indicate that this representation can be used to learn and recognize basic human action classes, independently of gender, body size and viewpoint.

We then present in the second part an approach based on a 3D exemplar-based HMM, which addresses the problem of recognizing actions from arbitrary views, even from a single camera. We will thus no longer require a 3D reconstruction during the recognition phase, instead we will use learned 3D models to produce 2D image information, which is compared to the observations.

In the third and last part, we present a compact and efficient exemplar-based representation, which in particular does not attempt to encode the dynamics of an action through temporal dependencies. In experimental results we demonstrate that such a representation can precisely recognize actions, even with cluttered and non-background-segmented sequences.

Contents

Résumé de la thèse **1**

1. Introduction **17**
1.1. The Problem . . . 18
1.1.1. Difficulties . . . 19
1.2. A Brief Background on Action Recognition . . . 20
1.3. Overview of Approaches and Contributions . . . 23
1.3.1. Motion History Volumes for Free Viewpoint Action Recognition . . . 24
1.3.2. Action Recognition from Arbitrary Views using 3D Exemplars . . . 26
1.3.3. Action Recognition using Exemplar-based Embedding . . . 26
1.3.4. IXMAS Dataset . . . 28
1.4. Thesis Outline . . . 28

2. State of the Art **31**
2.1. Posture Representation . . . 31
2.1.1. Model Based Representations . . . 32
2.1.2. Global Representations . . . 35
2.1.3. Local Representations . . . 37
2.2. Modeling Actions . . . 39
2.2.1. State-Transition Models . . . 39
2.2.2. Space-Time Representations . . . 41
2.2.3. Dynamic Free Representations . . . 43
2.3. View Independent Representations . . . 44
2.3.1. Normalization . . . 45
2.3.2. View Invariance . . . 46
2.3.3. Exhaustive Search . . . 49
2.4. Action Recognition in the Real World . . . 52
2.4.1. Temporal Segmentation . . . 52
2.4.2. Action Taxonomies . . . 55

2.4.3. Action Spotting And Garbage Models 57

I. Action Recognition in 3D: Motion History Volumes 59

3. Motion History Volumes 61
3.1. Definitions 63
3.1.1. Motion History Images 63
3.1.2. Motion History Volumes 63
3.2. Motion Descriptors 65
3.2.1. Invariant Representation 66
3.3. Classification Using Motion Descriptors 68
3.3.1. The IXMAS Dataset 69
3.3.2. Classification Using Mahalanobis Distance and PCA 69
3.3.3. Classification Using Linear Discriminant Analysis 72
3.3.4. Motion History vs. Motion Energy and Key Frames 74
3.3.5. Using Smaller Grid Resolution 75
3.3.6. Invariance vs. Alignment 76
3.4. Conclusion 77

4. Action Segmentation using Motion History Volumes. 79
4.1. Temporal Segmentation 81
4.2. Action Taxonomies 83
4.2.1. Clustering on Primitive Actions 83
4.2.2. Clustering on Composite Actions 84
4.3. Continuous Action Recognition 84
4.4. Discussion and Conclusion 87

II. Action Recognition from Arbitrary Views: 3D Exemplar-based HMM 91

5. Markov Models for Action Recognition 93
5.1. Markov Chain 93
5.2. Hidden Markov Models 94
5.2.1. Observation Probability 95
5.2.2. Forward-Backward Algorithm 96
5.2.3. Classification using Maximum a Posteriori Estimate 96
5.2.4. Viterbi Path 97
5.2.5. Learning HMM Parameters 97
5.3. Discrete HMM and Vector Quantization 98
5.4. Continuous and Semi-Continuous HMM 98
5.5. Exemplar-based HMM 99
5.5.1. Transformed Exemplar-based HMM 100

5.5.2. Exemplar Selection . . . 101
5.5.3. Selection Discussion . . . 103

6. Action Recognition from Arbitrary Views using 3D Exemplars. 105
6.1. Motivation . . . 106
6.2. Overview of Approach . . . 107
6.3. Probabilistic Model of Actions and Views . . . 109
6.4. Learning . . . 110
6.4.1. Exemplar Selection . . . 111
6.4.2. Learning Dynamics . . . 112
6.5. Action Recognition from 2D Cues . . . 113
6.6. Experiments . . . 114
6.6.1. Learning in 3D . . . 115
6.6.2. Learning from single views . . . 115
6.6.3. Comparison with MHVs . . . 117
6.6.4. On Importance of Modeling Dynamics . . . 120
6.7. Conclusion . . . 120

III. Action Recognition without Modeling Dynamics: Exemplar-based Embedding 123

7. Action Recognition using Exemplar-based Embedding 125
7.1. Introduction . . . 125
7.2. Action Modeling . . . 126
7.2.1. Exemplar-based Embedding . . . 129
7.2.2. Classifier . . . 129
7.2.3. Image Representation and Distance Functions . . . 130
7.3. Key-Pose Selection . . . 131
7.4. Experiments . . . 132
7.4.1. Evaluation on Segmented Sequences . . . 133
7.4.2. Evaluation on Cluttered Sequences . . . 135
7.5. Conclusion and Discussion . . . 139

IV. Conclusion and Perspectives 141

8. Conclusion 143
8.1. Summary . . . 143
8.2. Conclusion . . . 144
8.3. Limitations, Future Work, and Open Issues . . . 145

Bibliography 149

Résumé de la thèse

Le but de cette thèse est de développer un ensemble de nouvelles techniques pour l'identification d'action humaine à partir d'une vidéo. L'identification d'action — c.-à-d. observant d'autres personnes, identifiant ce qu'elles font, les imitant, ou réagissant à leurs actions — est une partie élémentaire de notre vie quotidienne. En fait, cette tache est toute a fait une tache banale et sans aucun effort pour la plus part d'entre nous. Pouvons-nous installer des modèles d'ordinateur qui peuvent démontrer des capacités semblables ?

En fait, les êtres humains sont capables de reconnaître facilement les actions humaines. Nous prenons beaucoup de plaisir à regarder des personnages en pleine action, dans des filmes, en sport, et dans des événements sociaux, par exemple, dès l'enfance nous commençons à regarder et observer les plus grands (parents et autres) pour apprendre d'eux comment se comporter. Les mécanismes qui nous donnent cette capacité passionnante, sont cependant que des connaissances et des compréhensions. Par conséquent le développement des techniques, qui donnent des capacités similaires sur des ordinateurs est une tache difficile et pour nous une grande challenge.

Il existe plusieurs applications intéressantes dans le domaine de la reconnaissance et d'identification d'action (voir figure 1) : nous citons, la grande base de donnée des images et vidéos sur Internet, qui ne cesse de grandir jour après jour, les archives de la télévision et les archives des filmes, ont besoin de beaucoup de compétences pour une classification automatique en catégories homogènes. En robotique, l'identification réussie d'une action est principale pour permettre aux hommes et aux ordinateurs une interaction normale et autonome. Dans les systèmes de surveillance modernes, les observateurs peuvent à peine percevoir la vue d'ensemble de toute l'information disponible. Par conséquent nous avons besoin de systèmes automatiques, qui nous aident à sélectionner l'information importante.

FIGURE 1.: Exemple d'actions dans différents domaines d'application : (En haut) Agressions sur des images de surveillance ; interactions entre l'homme et l'ordinateur ; (En bas) des gestes dans des filmes d'action, les nouvelles, et l'enregistrement de sport. Pouvons-nous concevoir une machine, qui peut comprendre telles actions ?

Actuellement, nous n'avons pas la connaissance pour établir de tels systèmes. D'abord, nous devons comprendre comment distinguer les actions en les voyant plusieurs fois s'exécuté, de différents points de vue, par des personnes différentes, avec différents modèles et dans différents contextes. Ensuite, nous devons comprendre comment employer cette connaissance instruite accumulée pour identifier et appeler de nouvelles actions quand nous les voyons. Dans cette thèse, nous voulons établir des modèles d'ordinateur qui démontrent de telles capacités.

Dans le reste de ce chapitre, nous continuerons la description des problèmes impliqués en reconnaissance et identification d'action. Nous donnons après un résume court des techniques existantes dans ce domaine, et présentons finalement nos approches et nos contributions.

Problèmes et difficultés

D'une manière technique, le terme *action* se réfère dans ce travail à un événement 4D, exécuté par un agent dans l'espace et à un temps précis. En particulier, notre centre d'attention est sur des ordres significatifs et des mouvements de corps courtes, tels que par exemple : *donner coups de pied*, *poignarder*, *marcher*, *s'assoir par terre*, etc. De tels événements habituellement ne peuvent pas être répétés exactement, et chaque exemple d'action est unique. Cependant, notre but est d'identifier des classes d'action, même lorsqu'elle est exécutée par différentes personnes sous différents points de vue, et malgré de grandes différences dans la façon et la vitesse de

l'exécution.

En développant un système pour l'identification d'action, nous devons construire une hiérarchie complexe avec de la *vision* et de la apprentissage automatique. Au niveau le plus bas, nous devons extraire le maintien et le mouvement dispositifs des vidéos, par exemple un modèle paramétrisé de corps ou une représentation simplifiée basée sur des images de silhouette. Le traitement de plus haut niveau consiste à propager ces dispositifs sur le temps et les tracer dans des classes significatives de mouvement. Chaque étape impliquée représente ses propres difficultés.

Les difficultés

Un système de reconnaissance d'actions réaliste et performant dans la vie quotidienne doit aborder une multitude des difficultés, provenant des issues très différentes. Dans le paragraphe suivant nous allons montrer en détails les différentes difficultés.

Difficulté de vision

L'extraction de caractéristiques distinctives à partir d'une séquence d'image est un problème fondamental dans la vision. En reconnaissance d'action la caractéristique visuelle la plus importante est le corps humain et sa configuration avec le temps. Des représentations très différentes peuvent être employées, s'étendant des modèles complexes de corps aux images simples de silhouettes. Dans l'un ou l'autre cas, les issues suivantes doivent être abordées : détection de personne ; extraction des caractéristiques distinctifs de la posture. De ce fait, les propriétés telles que la manipulation de l'occlusion et le fouillis, la robustesse au bruit, les différentes illuminations et la présence de l'ombres, et la robustesse à différents types de l'habillement et de physiques, sont d'importance.

De plus, dans des configurations réalistes nous ne pouvons pas imposer des contraintes au point de vue et à l'orientation des sujets en ce qui concerne les caméras. En conséquence, une pose ou un mouvement simple peut avoir comme conséquence un nombre presque infini d'observations possibles. Une représentation appropriée doit ainsi expliquer de tels changements.

Modélisation d'actions

Une autre difficulté principale vient de la grande variabilité qu'une classe d'action peut s'exposer, en particulier si elle est exécuté par différents sujets de genre et de taille différents, et avec une vitesse et une manière différentes. Les étiquettes des mouvements qui sont sémantiquement significatifs pour nous, tels que un *coup de pied*, un *coup de poing*, ou un *signe par la main*, peuvent avoir comme conséquence une latitude large des interprétations possibles. C'est ainsi

que ce problème est considéré comme un défi particulier pour concevoir un modèle d'action, qui identifie pour chaque action ses caractéristiques. Tandis qu'il maintient une adaptabilité appropriée à toutes les variations de formes. La question est quelle est la façon dont pouvons nous trouver des données appropriées, qui expliquent toute une telle variation. Les données annotées de mouvement sont crues et difficiles à acquérir, pourtant inévitable pour apprendre les modèles réalistes des actions.

Segmentation de mouvement et spotting

Étant donnée des modèles d'action appropriées, nous avons besoin de quelques approches pour appliquer ces modèles à des données réalistes sous forme de vidéos. Typiquement, les représentations d'action sont conçues et appris en utilisant des séquences artificielles, qui contiennent des exemples d'actions simples et isolés. En plus nous pouvons apprendre seulement un ensemble fini d'actions. Comment pouvons-nous appliquer des telles représentations sur un ensemble d'actions continues et réalistes ? Nous devons par conséquent localiser des frontières d'action dans les vidéos, et nous avons besoin de capacités pour repérer des actions intéressantes et la séparer du nombre presque infini de mouvements sans signification en générale que nous rencontrions en réalité.

Primitives des mouvements et taxonomies d'action

Finalement, en reconnaissance d'action le problème le plus important résulte du manque de la science de la théorie sur l'articulation et la perception de mouvement. L'identification d'action est souvent comparée à la reconnaissance de la parole, et beaucoup d'autres techniques sont adoptées du même domaine. Cependant, les systèmes de reconnaissance de la parole sont fortement basés sur les concepts linguistiques, tels qu'une représentation symbolique discrète de la langue parlée - représentée par des phonétiques, des syllabes, et des mots -, et leur assemblée dans la structure basée sur la syntaxe et la sémantique. De même, la formation du système de reconnaissance de la parole moderne est basée sur des données annotées avec plusieurs niveaux, par exemple à un niveau de phonème, sur un niveau de mot, et sur le niveau de phrase. Des concepts semblables sont absents en identification d'action (une exception sur des tâches spécialisées telles que l'annotation de danse). En conséquence, les concepts établis et les directives pour concevoir et apprendre des systèmes d'identification d'action sont absents.

État de l'art de la reconnaissance d'action

Comme nous avons cité précédemment, l'identification d'action peut être vu comme une combinaison de la vision et de la apprentissage automatique. Par conséquent diverses approches ont

été proposées, ces approches sont basées sur différente combinaison de méthodes. Bien que, beaucoup de directions intéressantes aient été ouvertes, jusqu'au maintenant, il n'y a pas beaucoup d'informations claires sur les meilleures manières de procéder [Forsyth, 2006]. Dans le paragraphe suivant nous allons donner une vue d'ensemble courte des méthodes employées ; une discussion détaillée peut être trouvée en chapitre 2.

Représentation de posture

La tâche principale de vision en l'identification d'action est l'extraction des posture et les caractéristiques de mouvement qui forment les séquences de vidéo. Traditionnellement, il y a deux vues contrastantes sur le type de représentation des caractéristiques à utiliser :

- l'approche basée sur le **modèle tente** d'extraire des caractéristiques qui décrivent explicitement la posture et le mouvement des parties du corps. Une telle représentation sont par exemple : marquer sur des certains points membres, chiffres bâton 2D, 3D ou cinématique organisme modèles. Rejet de la motion capture techniques, à marqueur ou sans marqueurs, sont nécessaires pour extraire de telles représentations. En conséquence, ces approches ne sont applicables que dans des configurations limitées, par exemple, des productions vidéo ou des analyses de sport, mais difficile à appliquer dans d'autres scénarios

- Pour surmonter les difficultés avec les extractions du modèle de corps, des représentations simplifiées basées sur des **templates locales et des modèles globaux** ont été proposées. Les représentations les plus utilisés couramment sont basé sur le flux optique, les silhouettes, et sur des correctifs locales similaire aux descripteurs SIFT en reconnaissance d'objet.

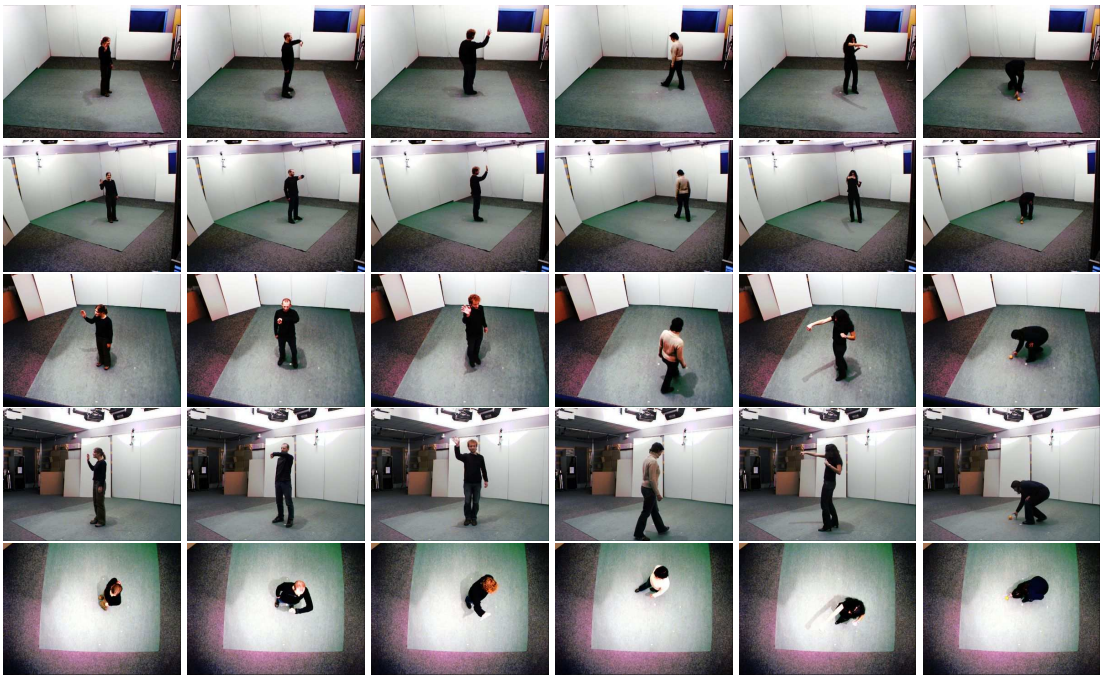
Modélisation d'action

Des techniques pour modeler des actions en termes de statistiques et la dynamique sur les caractéristiques des posture, peuvent être divisées en plusieurs vues contrastantes :

- Des actions peuvent être modelées en utilisant les techniques de reconnaissance des formes qui ont explicitement un composant temporel, c.-à-d. un modèle qui prévoit des observations à chaque instant basé sur des observations des exemples précédents dans d'autres périodes de temps, par exemple un HMM. Nous appelons de telles techniques **des modèles des états de transition**.

- Alternativement, le mouvement peut être modelé implicitement en utilisant ce que nous appelons une **représentation d'espace-temps**. Au lieu de regarder des actions en se basant sur "per-frame", des telles approches peuvent modèler des actions à une séquence de

(a) Deux ensembles de données fréquemment utilisés en reconnaissance d'action : (En haut) la base des données KTH [Schuldt et al., 2004], (En bas) l'ensemble des données de Weizmann [Blank et al., 2005]. Les vues sont approximativement fronto-parallèles, et les acteurs soit ils ont face à l'appareil-photo soit ils sont orientés parallèles aux plan de visionnement.



(b) Un exemple de séquence de notre base de données : nous enregistrons des actions avec des configurations arbitraires de multiples cameras, et les acteurs choisissent librement leur position et orientation par rapport à ces cameras.

FIGURE 2.: (a) La plus part des approches existantes assument la configuration de la vue fixe et la contrainte l'orientation des acteurs. (b) Nous travaillons aux scènes avec l'installation arbitraire de cameras.

niveau, c.-à-d. une séquence en entier a comme conséquence une représentation simple de dispositif, qui est instruite et classifiée en utilisant des techniques d'étude statiques, par exemple le plus proche voisin ou support vecteur machine (SVM).

- En conclusion, il est également possible de modeler des actions dans les **représentations dynamiques libres**, c.-à-d. à un niveau simple de frame, ou en considérant les ensembles non commandés temporels de frame, par exemple par occurrence de dispositif d'histogramme.

L'indépendance du point de vue

Des difficultés additionnelles sont présentées quand nous laissons observer des actions de différente points de vues, bien que, la majorité de travail en identification d'action adresse actuellement seulement le cas ou l'action dépend d'une vue bien déterminer, voir le schéma 2(a). Les approches Vue-indépendantes, peuvent être distinguées en se basant sur les stratégies suivantes :

- les méthodes qui se base sur **l'invariance de vue** consiste à trouver une fonction assortie entre les observations, qui est identique indépendante du point de vue observé. Une fonction si assortie peut être par exemple basée sur les contraintes epipolar ou les invariants géométriques, qui peuvent être calculés des correspondances point par point entre les paires d'images.

- Les approches basées sur la **normalisation**, tentent de détecter l'orientation réelle, et de transformer respectivement toutes les observations en base canonique, où l'assortiment a lieu. Pour détecter une telle orientation, les approches peuvent par exemple employer les parties du corps détectées, ou d'autres sélections, par exemple direction de marche.

- Les approches basées sur la **recherche exhaustive**, apprennent des vues multiples d'une action, et assortissent l'observation contre chacune d'elles. C'est probablement l'approche la plus franche pour prolonger n'importe quelle approche vue-dépendante à la vue-indépendance. D'une manière primordiale, cependant, pour noter, que la vue-indépendance demeurera restreinte exactement à l'ensemble de vues appris.

Approches et contributions

Comme cité précédemment, actuellement nous n'avons pas beaucoup d'informations claires sur l'identification d'action au sujet des meilleures manières de procéder. Il est donc important d'explorer différentes directions ; et nous faisons ainsi dans cette thèse. Nous proposons trois

nouveaux cadres pour l'identification d'action. Notre travail apporte en particulier des contributions en ce qui concerne les issues suivantes : modélisation vue-indépendante des actions, études de la posture et des primitifs de mouvement, modélisation de la dynamique des actions, segmentation temporelle des actions.

La motivation pour nos approches peut être récapitulée comme suit :

- **Modèle Libre :** Dans notre travail nous évitons l'utilisation d'un modèle paramétrisé de corps. Comme mentionné précédemment, la récupération d'un modèle de corps est un problème difficile, et actuellement limité à des configurations très contraints. Nous pensons que l'identification d'action peut fortement tirer bénéfice d'une représentation simplifiée de posture. Nous avons employé dans nos approches l'extraction des silhouettes du fond, bien que d'autres formes de représentations de dispositifs pourraient être intégrées (comme exemple, nous avons expérimenté aussi bien avec des images filtrées par contour et le chanfrein s'assortissant, pour éviter des dépendances sur la soustraction de fond).

- **L'indépendance de vue :** Notre but est de réaliser l'identification d'action d'une configuration qui ne dépend pas de la vue, en d'autre mots "vu ou filmer d'une vue arbitraire", et sans contraintes sur l'orientation relative entre les acteurs et les appareils-photo, voir le schéma 2(b). En plus, une fois appris avec une installation spécifique de caméra, nos modèles devraient être facilement transmissibles à de nouvelles installations de caméra. D'une manière primordiale, nous voulons réaliser ces demandes sans utilisation des entrées additionnelles, par exemple la présence des points de correspondances entre les observations. La clef pour remplir ces demandes dans nos approches exige l'utilisation de l'information 3D. Nous dérivons deux cadres de l'approche vue-indépendants (voir le schéma 3 et 4) : Dans notre première approche nous travaillons entièrement en 3D, en utilisant des reconstructions de "visual-hull" calculées à partir des observations de multiple vues. Notre deuxième approche essaye l'identification même d'une seule vue, encore une fois en utilisant les modèles précédemment instruits en 3D.

- **Différents modèles d'action :** Nous nous ne limitons pas à une représentation simple d'action, et expérimentons à la place avec différents modèles. Notre première approche est basée sur une représentation "espace-temps". Nous avons trouvé cette représentation dans une comparaison pour nous comporter le meilleur possible. D'une part, notre deuxième approche est basée sur un modèle d'état-transition, c.-à-d. un HMM, qui, en raison de ses caractéristiques génératives, a été adapté mieux pour produire des vues 2D arbitraires d'un modèle 3D. Dans notre dernière approche, nous avons expérimenté avec une représentation qui ne modèle pas la dynamique, et étonnant nous avons constaté qu'une telle représentation est souvent suffisante pour identifier des actions avec des résultats du

dernier cri.

- **Utilisation des primitifs de posture et de mouvement :** Nous pensons que les actions peuvent être représentées par des vocabulaires des primitifs élémentaires de mouvement et des primitifs de posture. Nous expérimentons avec différentes stratégies pour choisir quelques primitifs. Une méthode pour le choix des primitifs d'une séquence de mouvement continue est présentée dans la première partie de cette thèse. L'intérêt est la visualisation des données pour découvrir des classes de mouvement, qui n'exige aucune forme de surveillance. Dans la deuxième partie de cette thèse nous expérimentons avec le choix distinctif de clef-pose sur des séquences, cette fois en utilisant la surveillance en termes des étiquettes d'une classe donnée. Étonnant nous avons constaté qu'une fois choisi d'une telle manière, un très petit nombre de clef-pose est suffisant pour représenter une série de l'action effectuée par beaucoup de personnes.

Les approches et leurs contributions sont détaillées dans la suite.

Motion history volumes for free viewpoint action recognition

Dans notre première approche nous avons proposé l'aspect "Motion History Volumes" (MHVs) comme une représentation qui se repose sur l'indépendance de la vue d'ou la séquence de vidéo a été prise ou vue-invariable pour l'identification d'action humaine. L'idée fondamentale est de faire l'identification d'action en 3D, employant des séquences de visuel-hulls calculés à partir des appareils-photo calibrés plusieurs fois et par soustraits du fond, voir le schéma 3. Des caractéristiques de mouvement sont alors calculés en intégrant des observations multiples frame sur le temps a fin d'avoir un seul MHV simple, c.-à-d. une grille de voxels 3D qui code simultanément l'espace 3D et le temps. L'alignement et les comparaisons invariables d'orientation sont effectués efficacement en utilisant la transformée de Fourier dans des coordonnées cylindrique autour de l'axe vertical. Contrairement aux travaux existants sur l'identification par aspect de vue-invariable d'action (voir la section 2.3.2), notre représentation n'exige aucune sélection additionnelle, telle que les correspondances de point par exemple. Les contributions sont récapitulées comme suivant :

Représentation invariable de l'action en 3D : Nous présentons une nouvelle représentation invariable pour des actions observées dans l'espace du voxel 3D. La représentation est motivée par la prétention que les actions semblables diffèrent la plupart du temps par des transformations rigides composées par échelle, par translation et par rotation autour du l'axe z. On a montré que la représentation soutient u ne catégorisation significative des classes simples d'action exécutées par différents acteurs, indépendamment des tailles de point de vue, de genre et de taille du corps.

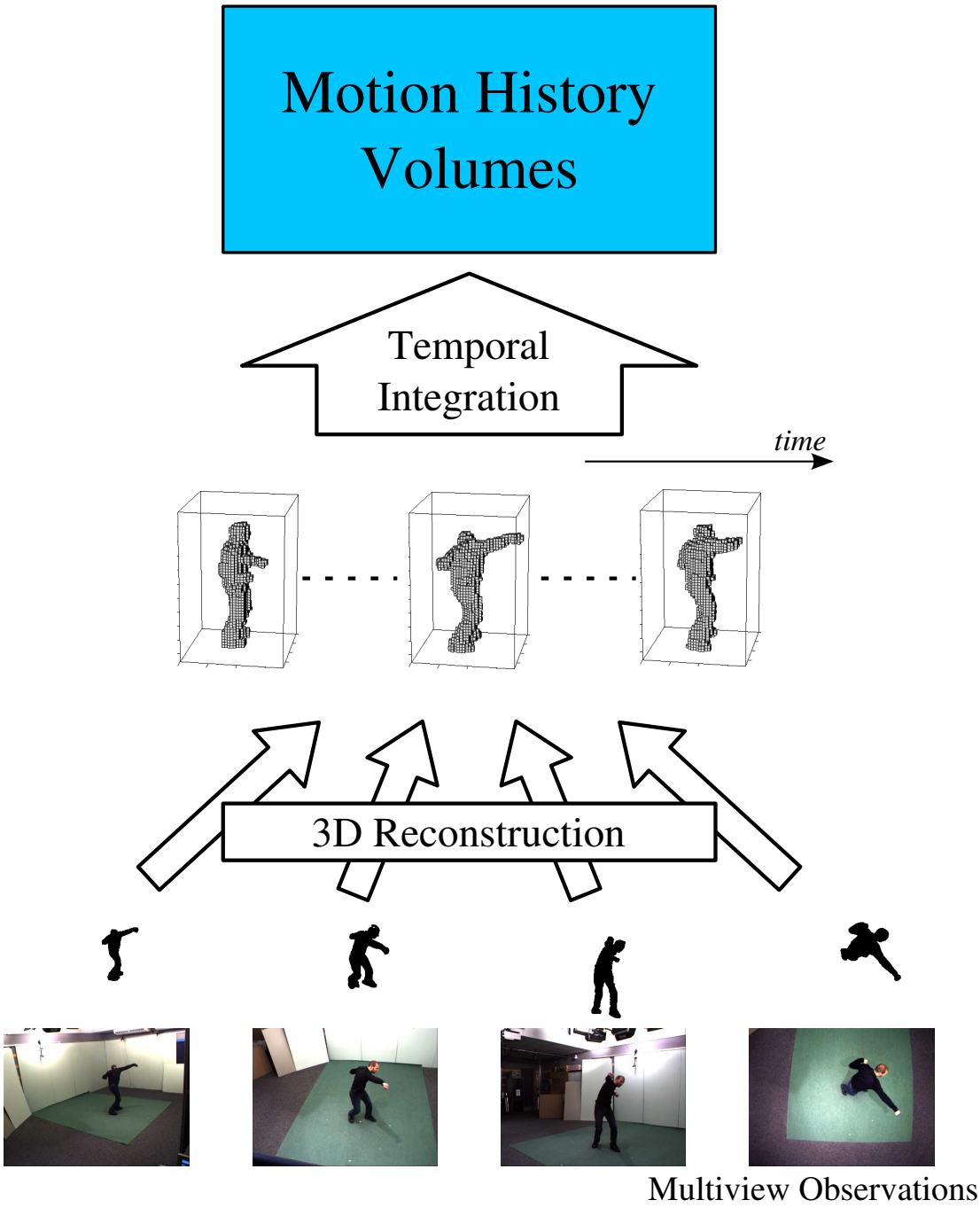


FIGURE 3.: Représentation d'action 3D "volumes d'histoire de mouvement" ("Motion History Volumes") : Nous employons des observations de multiple vue pour reconstruire des coques du visuel 3D. Les coques visuelles sont intégrées avec le temps dans des volumes d'histoire de mouvement (MHVs). Basé sur la méthode de MHVs nous apprenons et identifions des actions.

Segmentation temporelle : En se basant sur le MHVs, nous dérivons une méthode pour la segmentation temporelle des jets de mouvement. Puisque notre représentation est indépendante de point de vue, elle a comme résultats des méthodes de segmentation et de classification qui sont efficaces et robustes.

Identification d'action des vues arbitraires en utilisant des exemplaires 3D

Également, notre deuxième approche challenge l'identification d'action indépendamment de la vue. Cette fois nous atteignons ce but même lorsque une seul vue est donné, mais néanmoins sans exiger les parties du corps marquées ou n'importe quelle forme de correspondances de point entre les paires d'observations. En identifiant des actions d'une vue simple nous ne pouvons plus compter sur une reconstruction 3D pendant la phase d'identification. Cependant, la clef de notre approche pour réaliser l'identification vue-indépendante d'action est l'utilisation d'un modèle interne de l'action 3D, apprise des vues multiples, qui est alors employé pendant l'identification pour produire l'information 2D arbitraire d'image, voir le schéma 4. Cette approche apporte les contributions suivantes :

Exemplaire 3D basé sur le HMM : Le problème est formulé comme un modèle graphique de probabilité. Un exemplaire basé sur le HMM sera dérivé, qui représente l'action et la translation de vue comme deux paramètres indépendantes de processus de Markov. Le premier pour l'orientation du sujet relatif a la camera et le deuxième pour la vue indépendante, Le mouvement du corps-centré prises par l'artiste au cours des différentes étapes de l'action. Les processus aléatoires sont centrés autour d'un ensemble de postures 3D, les copies, qui représentent les différents états de mouvement, et qui sont utilisés pour générer la silhouette 2D observée

Sélection des postures dominant : Pour le choix du posture dominant, une approche de wrapper est proposée. On a montre qu'un petit ensemble de posture sélectionnée est suffisant pour représenter un grand nombre de différentes actions effectuées par différents acteurs.

Identification d'action utilisant l'encastrement base sur des exemplaires

Nos expériences précédentes avec de petits ensembles d'exemplaires ont inspiré notre dernière approche, où nous dérivons une représentation conduite a une représentation fortement simplifiée et purement de type " bottom-up" exemplaire. Nous nous présentons une représentation libre, compacte et efficace, basée sur un ensemble de distances à un ensemble d'exemplaires qui représente les clés statique distinctif de posture. En se basant sur une base de données publiques, très connue et représentes par une approche view-dependent, nous démontrons alors

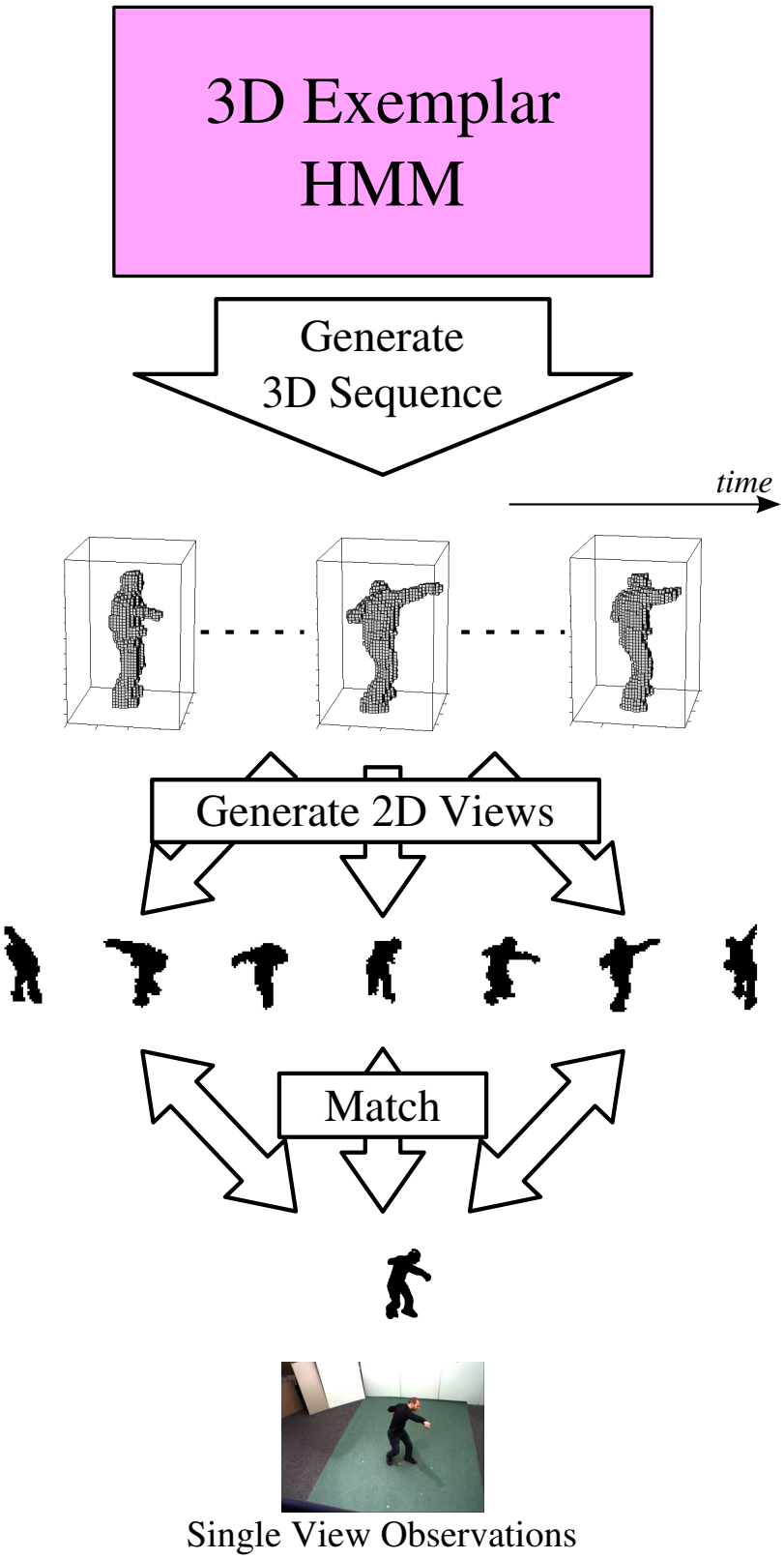


FIGURE 4.: ”Exemplaire-basé HMM” pour des reconnaissances d’action des vues arbitraires : Un modèle 3D appris est employé pour produire des séquences de visuels de la coque 3D. Les séquences 3D sont projetées en 2D et ajuster contre l’observation.

qu'une telle représentation, qui ne fait pas en particulier ne modélise aucune des relations temporelles entre les armatures, peut identifier des actions comparer aux résultats de l'état de l'art. En outre, nous expérimentons dans notre dernière approche avec des exemplaires et des images reconstruit par filtres sur les bords pour surmonter des dépendances sur la soustraction du fond. L'approche apporte les contributions suivantes :

Exemplaire incluant la représentation : Une nouvelle représentation est dérivée, qui représentent entièrement une séquence d'actions a travers une ensemble fixe d'exemplaires de distance. La représentation est équivalente à inclure des actions dans un espace défini par des distances à l'ensemble des clefs des posent des exemplaires. La représentation est peu sensible à l'ordre temporel et aux variations de d'échelle de temps, et a la vertu de l'efficacité et de la simplicité.

Modélisation libre dynamique d'action : L'approche démontre comment un ensemble de distinctif statique clef-pose est suffisant pour modeler beaucoup de différentes actions effectuées par différents acteurs. En particulier aucune modélisation des dépendances temporelles entre clef-pose n'est nécessaire pour réaliser des résultats comparables à l'état de l'art.

Identification des ordres encombrés : Nous démontrons comment la représentation de l'exemplaire proposé, en même temps que des distances assorties avancées, peut être employée pour l'identification des action a partir des actions qui contiennent beaucoup de bruit et sans soustraction du fond des séquences segmentées.

La base de données d'IXMAS

En conclusion, une contribution, qui est également devrait être mentionnée, est notre base de données (le schéma 2(b)fig :emb :3figa), que nous avons enregistré pendant nos expériences. Les ordres d'acquisition de mouvement de xmas d'Inria (IXMAS) forme un ensemble de données d'identification d'action de multi-vue de 13 jour de vie quotidienne d'actions, réalisé effectuées par 11 acteurs différents, chacun 3 fois en changeant le point de vue de la prise de la vidéo. C'est le seul base de données multi-vue qui est disponible publiquement dans le domaine de reconnaissance d'action. Depuis que nous l'avons rendu disponible publiquement, il a été téléchargé par plusieurs groupes de recherche partout dans le monde entier, et a été employé en plusieurs publications récentes de conférence, nous citons par exemple [Aganj et al., 2007, Lv and Nevatia, 2007, Zhang and Zhuang, 2007, **?**, Junejo et al., 2008, Liu and Shah, 2008, Vitaladevuni et al., 2008, Yan et al., 2008].

Profil de thèse

Un examen détaillé de la situation actuelle en reconnaissance d'action est présenté en **chapitre 2**. Par la suite la taxonomie illustrée brièvement dans la section 1.2, où nous classifions des approches selon la modélisation de l'action et la représentation et de la posture. Nous nous concentrons ensuite sur les représentations basant sur la vue-indépendant pour l'identification d'action, et finalement noua allons discuter des approches pour la segmentation automatique, découverte des primitives des mouvement, et l'identification des jets réalistes des actions.

Dans le **chapitre 3** nous présentons la méthode *motion history volumes* (MHVs) comme représentation d'action de point de vue libre. Motion history volumes sont dérivés comme extension de l'historique des mouvements des images [Bobick and Davis, 1996b]. Nous présentons alors des algorithmes pour l'alignement d'orientation et les comparaisons de MHVs, en utilisant une représentation basée sur des grandeurs de transformée de Fourier dans des coordonnées cylindrique. Les expériences sur l'ensemble de la base de données d'IXMAS prouvent que MHVs soutient la catégorisation significative des classes simples d'action qui ont ete exécuté par différents acteurs, indépendamment des tailles, de point de vue, de genre et de corps. Ce chapitre est basé sur un travail d'abord présenté à l'atelier international *IEEE International Workshop on modeling People and Human Interaction (PHI), 2005*, [Weinland et al., 2005], et a été mis à jour pour une version de journal dans *Computer Vision and Image Understanding (CVIU), 2006*, [Weinland et al., 2006b].

Nous présentons dans le **chapitre 4** une méthode pour la segmentation temporelle automatique en utilisant MHVs. A travers des expériences nous nous allons montrer comment la segmentation et la classification basées sur MHVs peuvent être employées pour découvrir automatiquement des taxonomies des primitifs de mouvement des ordres non étiquetés des mouvements. Plus loin nous donnons des résultats de classification et de détection sur les streams continus des actions. Ce chapitre est basé sur le travail publié dans la conférence *IEEE Vision and Pattern Recognition (CVPR), 2006*, [Weinland et al., 2006a]. Des partie ont été pris du papier de journal [Weinland et al., 2006b].

Dans le **chapitre 5** nous donnons une introduction courte concernant le HMMs et leur prolongation a l'exemplaire-basé HMMs. Des algorithmes pour l'étude de paramètre de HMM et le choix d'exemplaire sont donnés. Nous alors discuter particulièrement notre choix pour la sélection d'exemplaire, qui est basé sur une approche de wrapper [Kohavi and John, 1997], en comparaison avec d'autres méthodes sélectionnées.

Nous détaillons dans le **chapitre 6** notre approche pour l'identification d'action des vues arbitraires en utilisant une exemplaire-basée de HMM. Nous testons l'approche sur de diverses installations d'appareil-photo de l'ensemble de base de données d'IXMAS, et donnons finalement une comparaison aux résultats réalisés avec la représentation de MHV. Ce travail a été

présenté la première fois dans la *IEEE International Conference on Computer Vision (ICCV), 2007*, [Weinland et al., 2007].

Dans le **chapitre 7** nous présentons notre approche en utilisant l'exemplaire base sur l'encastrement. La représentation d'encastrement est détaillée et différents sélections pour la représentation d'image et des fonctions d'ajustement sont discutés. Nous avons performé des expériences sur l'ensemble de données disponible publiquement, et les nous l'avons comparé aux résultat de l'état de l'art. Ce chapitre est basé sur le travail présenté dans la conférence *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008*, [Weinland and Boyer, 2008].

Finalement, en **chapitre 8** nous allons conclure notre travail, nous discuterons les problèmes rencontrés, et nous donnerons des directions pour les travaux futurs.

CHAPTER 1

Introduction

The goal of this thesis is to develop a set of new techniques for action recognition from video. Action recognition — *i.e.* looking at other people, recognizing what they do, imitating it, or reacting to it — is an elementary part of our daily life. Actually, for most of us humans it is quite a common and effortless task. Can we build computer models that demonstrate similar capabilities?

In fact, humans easily do action recognition. We enjoy looking at other people in action; in movies, sport, and social events, for instance. From childhood on we love watching other people, and learning from them. The mechanisms, which enables us this fascinating ability, are however only little known and understand. Consequently, developing techniques, which give similar abilities to computers, is a difficult and challenging tasks.

There are numerous important applications of action recognition (see also Figure 1.1): for instance, the ever-growing amount of image and video information in the internet, TV and movie archives, needs capabilities for automatic indexing and categorization. In robotics, successful action recognition is key to allow humans and computers a natural and autonomous interaction. In modern surveillance systems, human observers can barely overview all available information. Hence we need automatic systems, which help us in selecting the important information.

Currently, we do not have the knowledge to build such systems. First, we need to understand how to learn actions by seeing them performed multiple times, from different viewpoints, by different people, with different styles and in different contexts. Second, we need to understand how to use that accumulated learned knowledge to recognize and name new actions when we see them. In this thesis, we want to build computer models that demonstrate such capabilities.

Figure 1.1.: Sample actions in different application domains: (Top) aggressions in surveillance footage; interactions between human and computer; (bottom) events in feature film, news, and sport recordings. Can we design a machine, which can understand such actions?

In the remainder of this chapter, we will continue outlining the problems involved in action recognition. We then give a short review of existing techniques, and finally introduce our approaches and contributions.

1.1. The Problem

Technical speaking, the term *action* refers in this work to a 4D event, performed by an agent in space and in time. In particular, our focus is on meaningful, short sequences of body motions, such as for instance: *kicking*, *punching*, *walking*, *siting down*, *etc*. Such events usually cannot be repeated exactly, and each action instance is unique. Yet, our aim is to recognize action classes, even when performed by different agents under different viewpoints, and in spite of large differences in manner and speed.

When developing a system for action recognition, we have to construct a complex hierarchy of *vision* and *learning* modules. On the low level, we need to extract posture and motion features from videos, *e.g*. a parameterized body model or a simplified representation based on silhouette images. The higher level processing consists of propagating those features over time and mapping them into meaningful motion classes. Each step involved represents its own difficulties

1.1.1. Difficulties

An realistic action recognition system has to address a multitude of difficulties, originating from very different issues.

Vision Issues

Extracting discriminative features from image sequences is a fundamental problem in vision. In action recognition the visual feature of interest is the human body and its configuration over time. Very different representations can be used, ranging from complex body models to simple silhouettes images. In either case, the following issues need to be addressed: person detection and location; extraction of discriminative posture features. Thereby properties such as handling of occlusion and background clutter, robustness to noise, different illumination and shadows, and robustness to different types of clothing and physiques, are of importance.

Further, in realistic settings we can not impose constraints on viewpoint and orientation of the subjects with respect to the cameras. Consequently, a single pose or motion can result in an almost infinite number of possible observations. An appropriate representation needs thus to account for such changes.

Action Modeling

Another major difficulty comes from the large variability that an action class can exhibit, in particular if performed by different subjects of different gender and size, and with different speed and style. Motions labels which are semantically meaningful to us, such as *kick*, *punch*, or *wave*, can result in a wide latitude of possible interpretations. It is thus a particular challenge to design an action model, which identifies for each action the characteristic attitudes, while maintaining appropriate adaptability to all forms of variations.

Directly related is the question of how can we find appropriate training data, which accounts for all such variation. Annotated motion data is raw and difficult to acquire, yet inevitable for learning realistic models of actions.

Motion Segmentation and Spotting

Given appropriate action models, we need approaches to apply these to realistic video data. Typically, action representations are designed and learned using artificial sequences, which contain single, isolated action instances. Further we can only learn a finite vocabulary of actions. How can we apply such representations to realistic continuous streams of action utterances? We consequently need to locate action boundaries in video streams, and we need capabilities to

spot interesting actions out of the almost infinite number of meaningless motions that we will encounter in reality.

Motion Primitives and Action Taxonomies

Finally, an important issue in action recognition results from the lack of a theoretic science on motion articulation and perception. Action recognition is often compared to speech recognition, and many techniques are adopted from this field. However, speech recognition systems are heavily based on linguistic concepts, such as a discrete symbolic representation of the spoken language — represented through phonemes, syllables, and words —, and their assembly into structure based on syntax and semantics. Similar, the training of modern speech recognition system is based on data annotated on multiple levels, *e.g.* on a phoneme level, on a word level, and on a the sentence level. Similar concepts are missing in action recognition (an exception are specialized tasks such as dancing annotation). Consequently, established concepts and guidelines for designing and learning action recognition systems are missing.

1.2. A Brief Background on Action Recognition

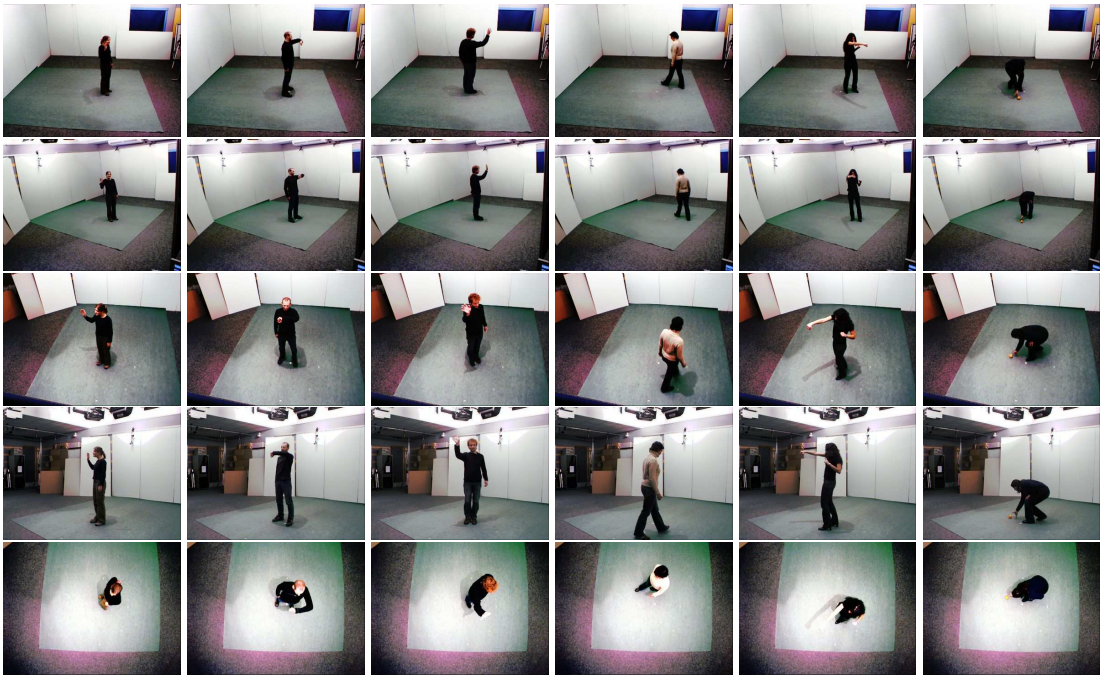
As mentioned earlier, action recognition can be viewed as a combination of vision and pattern modeling algorithms. Hence various approaches have been proposed based on different combination of such methods. Although, many interesting directions have been opened, up to now, there is very little clear information about best ways to proceed [Forsyth, 2006]. In the following we give a short overview of the methods used; an detailed discussion can be found in Chapter 2.

Posture representation

The principle vision task in action recognition is the extraction of posture and motion features form the video sequences. Traditionally, there are two contrasting views on the kind of feature representation to use:

- **Model based** approaches attempt to extract features which explicitly describe position and motion of body parts. Such representations are for instance: marker points on certain limbs, 2D stick figures, or 3D kinematic body models. Motion capture techniques, marker-based or marker-free, are necessary to extract such representations. Consequently, these approaches are only applicable in constrained settings, *e.g.* video productions or sports analysis, but difficult to apply in other scenarios.

(a) Two frequently used datasets in action recognition: (top) KTH-dataset [Schuldt et al., 2004], (bottom) Weizmann-dataset [Blank et al., 2005]. The views are approximately fronto-parallel, and actors either face the camera or are oriented parallel to the viewing plane.



(b) Sample sequences from our dataset: we record actions with multiple arbitrary camera configurations, and the actors freely choose their position and orientation with respect to these cameras.

Figure 1.2.: (a) Most existing approaches assume fixed view setup and constraint the orientation of the actors. (b) We work on scenes with arbitrary camera setup.

- To overcome the difficulties with body model extraction, simplified representations based on **local and global templates** have been proposed. Commonly used representations are based on optical flow, silhouettes, and local patches similar to SIFT descriptors in object recognition.

Action modeling

Techniques for modeling actions in terms of statistics and dynamics over posture features, similar can be divided into several contrasting views:

- Actions can be modeled using pattern recognition techniques that explicitly have a temporal component, *i.e*. a model which predicts an observations at each time instance based on observations from the preceding time instances, *e.g*. a HMM. We name such techniques **state-transition models**.

- Alternatively, motion can be modeled implicitly using so called **space-time representations**. Instead of looking at actions at a ”per frame” basis, such approaches model actions on a sequence level, *i.e*. a whole sequence results in a single feature representation, which is learned and classified using static machine learning techniques, *e.g*. *nearest neighbor* or *support vector machine*.

- Finally, it is also possible is to model actions in **dynamic free representations**, *i.e*. either on a single frames level, or by considering temporal unordered sets of frames, *e.g*. by histogramming feature occurrence.

View Independence

Additional difficulties are introduced when we allow to observe actions form different and changing views, although, the majority of work in action recognition currently addresses only the view-dependent case, see Figure 1.2(a). View-independent approaches, can be distinguished based on the following strategies:

- **View invariance** is the idea of finding a matching function between observations, which is the same independent of the observed viewpoint. Such a matching function can be for instance based on *epipolar constraints* or *geometrical invariants*, which can be computed from given point-to-point matches between pairs of images.

- Approaches based on **normalization**, attempt to detect the actual orientation, and respectively transform all observations into a canonical coordinate frame, where the matching takes place. To detect such orientation, approaches can for instance use detected body parts, or other cues, *e.g*. walking direction.

- Approaches based on **exhaustive search**, learn multiple views of an action, and match the observation against each of them. This is probably the most straightforward approach to extend any view-dependent approach to view-independence. Importantly, however, to notice, that view-independence will remain restricted to exactly the set of learned views.

1.3. Overview of Approaches and Contributions

As mentioned earlier, currently we have very little clear information in action recognition about the best ways to proceed. It is therefore important to explore different directions; and we do so in this thesis. We propose three new frameworks for action recognition. Our work makes in particular contributions with respect to the following issues: view-independent modeling of actions, learning of pose and motion primitives, modeling of action dynamics, temporal segmentation of actions.

The motivation for our approaches can be summarized as following:

- **Model free:** In our work we avoid the use of a parameterized body model. As mentioned previously, recovering a body model is a difficult problem, and currently restricted to very constrained settings. We think that action recognition can strongly benefit from a simplified posture representation. We used in our approaches background subtracted silhouettes, although other forms of features representations could be integrated (as an example, we experimented as well with edge filtered images and chamfer matching, to avoid dependencies on background subtraction).

- **View independent:** Our aim is to achieve action recognition from arbitrary view configuration, and without constraints on the relative orientation between the actors and cameras, see Figure 1.2(b). Further, once learned with a specific camera setup, our models should be easily transferable to new camera setups. Importantly, we want to achieve this demands without the use of additional cues, *e.g.* given point correspondences between observations. Key to meet this demands in our approaches is the used of 3D information. We derive two view-independent frameworks (Figure 1.3 and 1.4): In our first approach we work entirely in 3D, using visual-hull reconstructions computed from multiple views observations. Our second approach attempts recognition even from a single view, yet again using previously learned 3D models.

- **Different action models:** We do not restrict ourself to a single action representation, and experiment instead with different models. Our first approach is based on a so called *space-time* representation. Interestingly we found this representation in a comparison to perform best. On the other hand, our second approach is based on a state-transition

model, *i.e*. a HMM, which, because of its generative characteristics, was best adapted to generate arbitrary 2D views from a 3D model. In our last approach, we experimented with a representation which does not model dynamics, and surprisingly we found that such a representation is often sufficient to recognize actions with state-of-the-art results.

- **Use of posture and motion primitives:** We think actions can be represented through vocabularies of elementary motion and posture primitives. We experiment with different strategies to select such primitives. A method for selection of motion primitives from continues motion sequences is presented in the first part of this thesis. The interest is a purely visual-data driven discovery of motion classes, which does not require any form of supervision. In the second part of this thesis we experiment with discriminative selection of key-poses from sequences, this time using supervision in terms of given class labels. Surprisingly we found that when selected in such a way, very small number of key-poses are sufficient to represent a variety of action performed by many different people.

The approaches and their contributions are detailed as follows.

1.3.1. Motion History Volumes for Free Viewpoint Action Recognition

In our first approach we propose Motion History Volumes (MHVs) as a view-invariant representation for human action recognition. The basic idea is to do action recognition in 3D, using visual-hulls sequences computed from multiple calibrated and background subtracted cameras, see Figure 1.3. Motion features are then computed by integrating multiple frame observations over time into a single MHV, *i.e*. a 3D voxel-grid which simultaneously encodes 3D space and time. Orientation invariant alignment and comparisons are performed efficiently using Fourier transforms in cylindrical coordinates around the vertical axis. In contrary to most existing works on view-invariant action recognition (see Section 2.3.2), our representation does not require any additional cues, such as given point correspondences for instance. The contributions are summarized as following:

3D Invariant Action Representation: We present a new invariant representation for actions observed in 3D voxel space. The representation is motivated by the assumption that similar actions mostly differ by rigid transformations composed of scale, translation, and rotation around the z-axis. It is shown that the representation supports meaningful categorization of simple action classes performed by different actors, irrespective of viewpoint, gender and body sizes.

Temporal Segmentation: Based on MHVs we derive a method for temporal segmentation of motion streams. Because our representation is independent of viewpoint, it results in

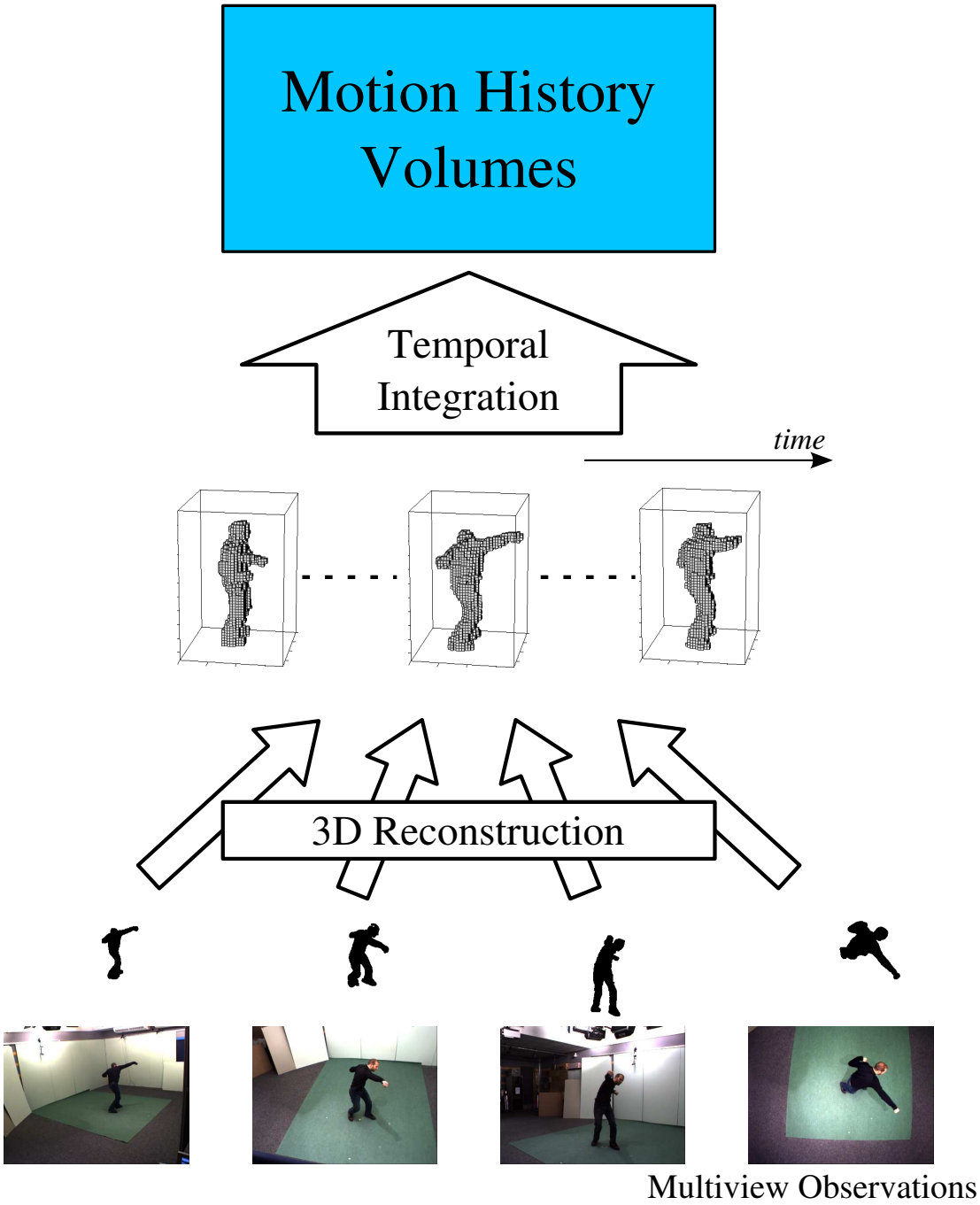


Figure 1.3.: 3D action representation "Motion History Volumes": We use multiple view observation to reconstruct 3D visual hulls. The visual hulls are integrated over time into Motion History Volumes (MHVs). Based on MHVs we learn and recognize actions.

segmentation and classification methods which are surprisingly efficient and robust.

Unsupervised Discovery of Motion Primitives: Based on the above contributions, we are able to segment action streams and cluster those segments into a hierarchy of primitive action classes. Our method can be used as the first step in a semi-supervised action recognition system that will automatically break down training examples of people performing sequences of actions into primitive actions that can be discriminatingly classified and assembled into high-level recognizers.

1.3.2. Action Recognition from Arbitrary Views using 3D Exemplars

Also our second approach attempts action recognition independent of view. This time we achieve this goal even when only a single view is given, but nevertheless without requiring labeled body parts or any form of point correspondences between pairs of observations. When recognizing actions from a single view we can not longer rely on a 3D reconstruction during the recognition phase. Yet, key of our approach to achieve view-independent action recognition is the use of an internal 3D action model, learned from multiple views, which is then used during recognition to produce arbitrary 2D image information, see Figure 1.4. The approach makes the following contributions:

3D Exemplar-based HMM: The problem is formulated as a probabilistic graphical model. An exemplar-based HMM is derived, which represents action and view-transform as two independent Markov processes, one for the orientation of the subject relative to the camera, and the other for the view-independent, body-centered motion states taken by the performer during the various stages of the action. The random processes are centered around a set of discriminative 3D key-poses, the exemplars, which represent the different motion states, and which are used to generate the 2D silhouette observations.

Discriminative Key-Pose Selection: For selection of the key-poses, a *wrapper approach* is proposed. It is shown, that a very small set of such discriminatively selected key-poses is sufficient to represent a large number of different actions performed by different actors.

1.3.3. Action Recognition using Exemplar-based Embedding

Our previous experiments with small sets of exemplars inspired our last approach, where we derive a strongly simplified and purely bottom-up driven exemplar representation. We present a dynamic free, compact and efficient action representation, based on a set of distances to a

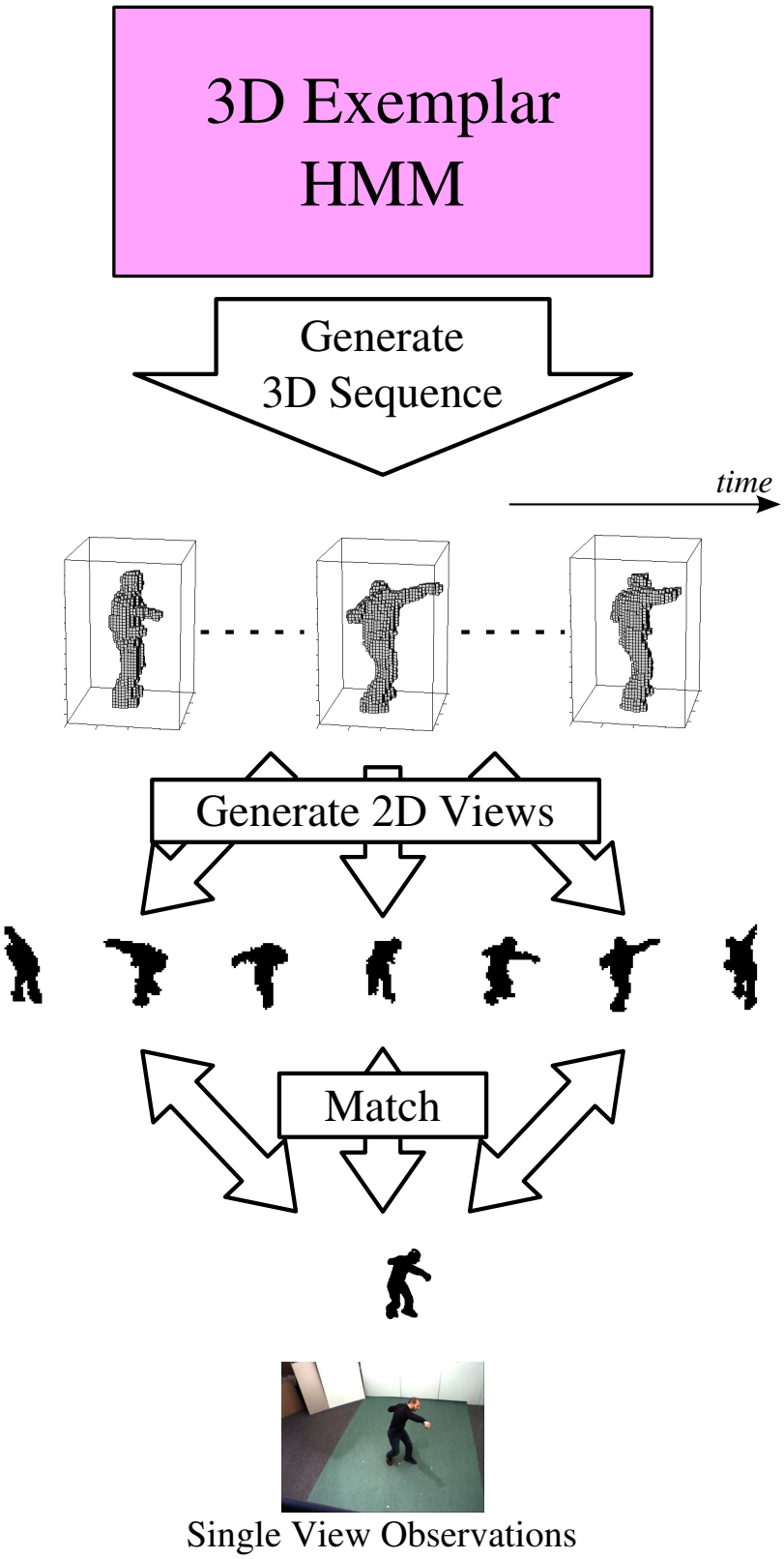


Figure 1.4.: ”Exemplar-based HMM” for action recognitions from arbitrary views: A learned 3D model is used to generate 3D visual hull sequences. The 3D sequences are projected into 2D and matched against the observation.

set of discriminative static key-pose exemplars. On a public available and well known view-dependent dataset, we then demonstrate that such a representation, which in particular does not attempt to model any temporal relations between frames, can recognize actions with state-of-the-art results. Also, we experiment in our last approach with exemplars and edge-filters images to overcome dependencies on background-subtraction. The approach makes the following contributions:

Exemplar Embedding Representation: A new representation is derived, which represent an entirely action sequence trough a fixed set of exemplar distances. The representation is equivalent to embedding actions into a space defined by distances to key-pose exemplars. The representation is insensitive to temporal order and variations in time-scale, and has the virtue of efficiency and simplicity.

Dynamic free Action Modeling: The approach demonstrates how a set of discriminative static key-poses is sufficient to model many different actions performed by different actors. In particular no modeling of temporal dependencies between key-poses is necessary to achieve state-of-the-art results.

Recognition from Cluttered Sequences: We demonstrate how the proposed exemplar representation, in conjunction with advanced matching distances, can be used for recognition from cluttered and non-background segmented sequences.

1.3.4. IXMAS Dataset

Finally, a contribution, which is also worth mentioning, is our dataset (Figure 1.2(b)), which we recorded during the course of our experiments. The Inria xmas motion acquisition sequences (IXMAS) forms a multi-view action recognition dataset of 13 daily-live actions, performed by 11 different actors, each 3 times with changing viewpoint. It is the only publicly available multiview action recognition dataset. Since we made it publicly available, it has been downloaded from research groups all over the world, and was used in several recent peer conference publications, *e.g.* [Aganj et al., 2007, Lv and Nevatia, 2007, Zhang and Zhuang, 2007, Junejo et al., 2008, Liu and Shah, 2008, Vitaladevuni et al., 2008, Yan et al., 2008].

1.4. Thesis Outline

A detailed review of the state-of-the-art in action recognition is presented in **Chapter 2**. We thereby follow the taxonomy briefly sketched in Section 1.2, where we classify approaches ac-

cording to posture representation and action modeling. We then focus on view-independent representations for action recognition, and finally discuss approaches for automatic segmentation, motion primitives discovery, and recognition from realistic streams of actions.

In **Chapter 3** we present motion history volumes (MHVs) as a free-viewpoint action representation. Motion history volumes are derived as extension to motion history images [Bobick and Davis, 1996b]. We then present algorithms for orientation invariant alignment and comparisons of MHVs, using a representation based on Fourier magnitudes in cylindrical coordinates. Experiments on the IXMAS dataset show that MHVs supports meaningful categorization of simple action classes performed by different actors, irrespective of viewpoint, gender and body sizes. This chapter is based on work first presented at the *IEEE International Workshop on modeling People and Human Interaction (PHI), 2005*, [Weinland et al., 2005], and was revised for a journal version in *Computer Vision and Image Understanding (CVIU), 2006*, [Weinland et al., 2006b].

In **Chapter 4** we present a method for automatic temporal segmentation using MHVs. In experiments we show how segmentation and classification based on MHVs can be used to automatically discover taxonomies of motion primitives from unlabeled sequences of motions. Further we show classification and detection results on continuous streams of actions. This chapter is based on work published in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2006*, [Weinland et al., 2006a]. Parts where also taken from the Journal paper [Weinland et al., 2006b].

In **Chapter 5** we give a short introduction into HMMs and their extension to exemplar-based HMMs. Algorithms for HMM parameter learning and exemplar selection are given. We then specially discuss our choice for exemplar selection, which is based on a wrapper approach [Kohavi and John, 1997], in comparison with other selection methods.

In **Chapter 6** we detail our approach for action recognition from arbitrary views using an exemplar-based HMM. We experiment the approach on various camera setups from the IXMAS dataset, and finally give a comparison to results achieved with the MHV representation. This work was first presented in *IEEE International Conference on Computer Vision (ICCV), 2007*, [Weinland et al., 2007].

In **Chapter 7** we present our approach for exemplar based embedding. The embedding representation is detailed, and different choices for image representation and matching function are discussed. We perform experiments on a publicly available dataset, and compare them with the state-of-the-art. This chapter is based on work presented in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008*, [Weinland and Boyer, 2008].

Finally, in **Chapter 8** we conclude our work, discuss issues, and give directions for future work.

CHAPTER 2

State of the Art

Action recognition has become an active research topic in computer vision over the last years, and a considerable amount of literature exists on several aspects of it. As stated earlier, actions recognition approaches typically consist of a combination of vision and machine learning techniques. The vision part is to extract representative posture features from the image or video signal; the machine learning part is to model action typical distributions over such features, in posture-space and in time. Although boundaries between the two parts sometimes overlap, we nevertheless found such a classification of approaches, based on posture representation and action representation, most adequate to review current state of the art approaches; and we will use this taxonomy in the following. Another important point, and one of the main contributions of our work, is view-independent modeling of actions. Therefore we will review different view representation used for action recognition in a separate section. Also learning and modeling of motion primitives, and their automatic identification in sequences is of interest for this thesis and will be reviewed in a separate section. For more broad reviews of tracking, motion capture, and recognition techniques we refer to the surveys [Cedras and Shah, 1994, 1995, Aggarwal and Cai, 1999, Gavrila, 1999, Moeslund and Granum, 2001, Aggarwal and Park, 2004, Forsyth et al., 2005, Moeslund et al., 2006].

2.1. Posture Representation

Various ways to represent human posture for action recognition have been suggested. We separate between *model based representation*, *i.e*. approaches based on parametric body models that

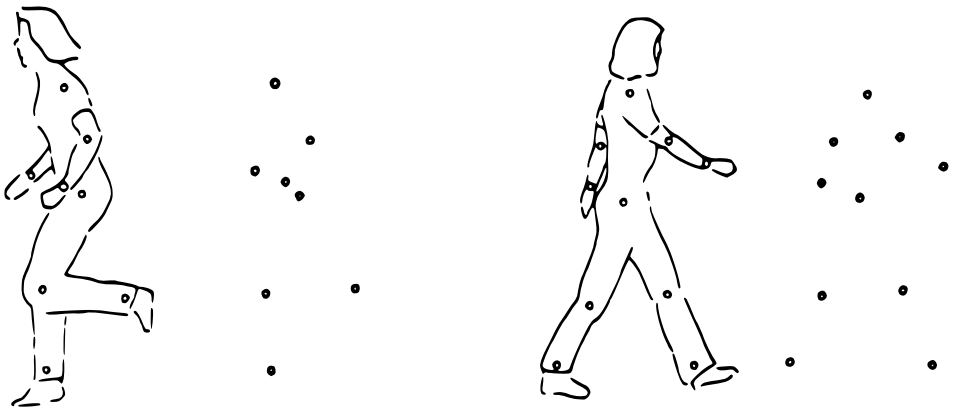


Figure 2.1.: Illustration of moving light displays, taken from [Johansson, 1973]. Johansson showed that humans can recognize actions merely from the motion of a few light displays attached to the human body.

use techniques from motion capture (MOCAP), and approaches that do not use a body model. For the latter we distinguish further between approaches based on *global representations* and approaches based on *local representations*.

2.1.1. Model Based Representations

Model based approaches represent posture in form of a parametric model of the human body. In his seminal psychophysical work on visual interpretation of biological motion, Johansson [1973] showed that humans can recognize actions merely form the motion of a few moving light displays (MLD) attached to the human body (Figure 2.1). Over several decades these experiments inspired approaches in action recognition, which used similar representations based on motion of landmark points on the human body. The experiments were also origin of the vexed controversy on whether humans actually recognize actions directly from 2D motion patterns, or whether they first compute a 3D reconstruction from the motion of the patters — and accordingly the question: how to proceed in action recognition? Other works [Cutting and Kozlowski, 1977, Kozlowski and Cutting, 1977] showed, that humans can even identify gender and identity from MLDs. Later, Sumi [1984] found that upside-down recordings of MLDs are usually not recognized by humans, what was interpreted as the presence of a strong prior model in humans perception [Goddard, 1989], which expects people walking upright and which can not adapt to strong transformations.

Consequently, two different paradigms exist: ”recognition by reconstruction” and ”direct recognition”. *Recognition by reconstruction approaches* use motion capture techniques to estimate a 3D model of the human body, typical represented as a kinematic joint model. See Figure

2.2 for some examples. In their seminal theoretical work on representation of three dimensional shapes, Marr and Nishihara [1978] proposed a body model consisting of a hierarchy of cylindrical primitives. Such a model was later adopted by several top-down approaches, *e.g.* [Hogg, 1983, Rohr, 1994]. ”Top down” means here, that a 3D model is used in a generative framework, *i.e.* 3D body models are sampled from the search space of joint configurations, projected into 2D, and matched against the observation. A more general body model based on super-quadrics was used in the multiview approach of Gavrila and Davis [1995]. Even more flexible the model used in [Green and Guan, 2004], which approximates body parts in 3D through a textured spline model. In [Ramanan and Forsyth, 2003, Ikizler et al., 2007] a bottom-up approach is proposed, which first tracks body parts in 2D, using rectangular appearance patches, and then lifts the tracked 2D configuration into 3D. Marker-based MOCAP techniques were also used for action recognition, for instance Campbell and Bobick [1995] compute a joint model from 14 marker points attached to a ballet dancer’s body. Instead of recovering kinematic joint configurations, several approaches directly work on the trajectories of 3D anatomical landmarks, *e.g.* [Campbell et al., 1996, Brand et al., 1997, Wilson and Bobick, 1999].

Direct recognition approaches directly work on 2D models of the human body, without lifting these into 3D. Common 2D representations are stick figures and 2D anatomical landmarks, similar to Johansson’s MLDs. Goddard [1992] works on interpretation of MLDs. Guo et al. [1994] recover a stick figure from the skeleton of a person’s silhouette. Niyogi and Adelson [1994] detect stick figure motions in the space-time volume spanned by an image sequence of a walking person. Alternatively, coarse 2D body representations based on blobs and patches can be used, *e.g.* [Starner and Pentland, 1995, Bregler, 1997, Brand et al., 1997, Yacoob and Black, 1998].

Independent of the model used (2D or 3D) the difficulties with finding body parts and estimating a parametric body model are evident. Commercial MOCAP systems use markers attached to the actors to recover the body pose, or require heavy user interaction, but there are only few applications in action recognition that allow for such constraints, *e.g.* film production and annotation of MOCAP data for animation [Arikan et al., 2003]. Markerless MOCAP is typically based on highly non-convex optimizations, which are doomed to such issues as false initialization, local extrema, and non-recovery from failure. Recent developments in MOCAP [Rosales and Sclaroff, 2000, Sminchisescu and Triggs, 2001, Agarwal and Triggs, 2006] use strong prior learning to reduce such issues by assuming particular types of activities, walking or running for instance, and thus by imposing constraints on the type of possible body configuration. Hence, these methods strongly reduce the search space of possible poses considered, which, however, limits their application to action recognition. Moreover, works that combine MOCAP with action models [Zhao and Nevatia, 2002, Peursum et al., 2007], indicate that successful MOCAP may need good action models, rather than the other way around.

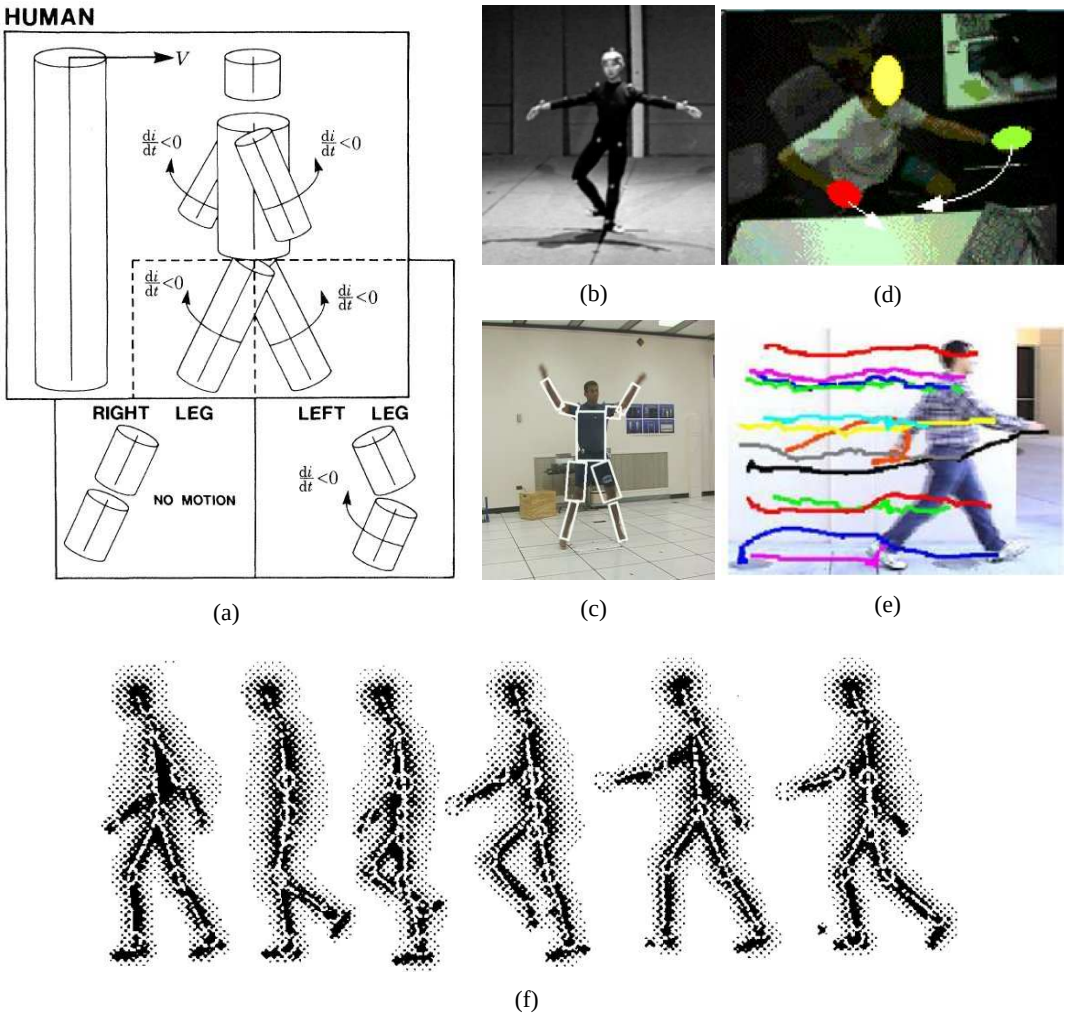


Figure 2.2.: Model based posture representations: (a) hierarchical 3D model based on cylindrical primitives [Marr and Vaina, 1982]; (b) ballet dancer with markers attached to body [Campbell and Bobick, 1995]; (c) body model based on rectangular patches [Ramanan and Forsyth, 2003]; (d) blob model [Brand et al., 1997]; (e) 2D marker trajectories [Yilmaz and Shah, 2005b]; (f) stick figure [Guo et al., 1994].

2.1.2. Global Representations

The difficulties with estimating exact body models drove the development for other representations. Global representations, also sometimes called *holistic* representations, do not model body parts, and instead only encode global body structure, mostly in form of fixed size templates. See Figure 2.3 for some examples. Such representations are strong simplifications compared to parametric body models. As our goal is, however, *not* the exact recovery of joint configurations, but to provide useful features for a higher level activity reasoning, action recognition can strongly benefit from such simplified representations, which avoid the difficulties of MOCAP approaches.

In their seminal work Bobick and Davis [1996a] motivated the use of templates for action recognition in an experiment. In the spirit of Johansson [1973] (see Section 2.1.1) they showed sequences of extremely low dimensional and blurred action recordings to an audience. Unlike with Johansson's MLDs, the blurred sequences did not carry sufficient cues for reconstructing and fitting human body parts, yet most of the people where able to recognize the actions. Bobick and Davis [1996a] concluded, that often simplified body representations, such as blurred templates, contain sufficient cues for recognition.

Among the various global representations used by the various action recognition approaches, silhouettes and contours are most frequently. Yamato et al. [1992] divide a silhouette image into a set of non overlapping bins, and compute the ratio of black and white pixels within each bin. A similar representation is also used in [Wang and Suter, 2007]. Bobick and Davis [1996b] integrate silhouettes over time in so called *motion history images* (MHI) and *motion energy images* (MEI),see Figure 2.5. In [Masoud and Papanikolopoulos, 2003] a similar representation is derived based on an *infinite impulse response filter*. Meng et al. [2007] proposes the use of a hierarchical MHI. Rittscher and Blake [1999] use a spline contour to track the outline of a person. Ogale et al. [2004] uses phase correlation to match silhouette images. Blank et al. [2005] and Yilmaz and Shah [2005a] both work on the volume spanned by silhouette images over time. Lv and Nevatia [2007] matches silhouettes using shape context descriptors.

Silhouette representation are insensitive to color, texture, and contrast changes, but nevertheless provide sufficient discriminative information for pose classification. On the downside, silhouette base representations fail in detecting self occlusions, and depend on a robust background segmentation. Consequently, few attempts have been made to apply above methods in uncontrolled setting, *e.g*. outdoor scenes, where exact background segmentation is difficult. The chamfer distance has been used to match silhouettes of humans in cluttered scenes [Gavrila and Philomin, 1999, Toyama and Blake, 2001, Elgammal et al., 2003], but few of these techniques have been extended to action recognition in uncontrolled setting.

Instead of depending on a background model, one can segment body structure from back-

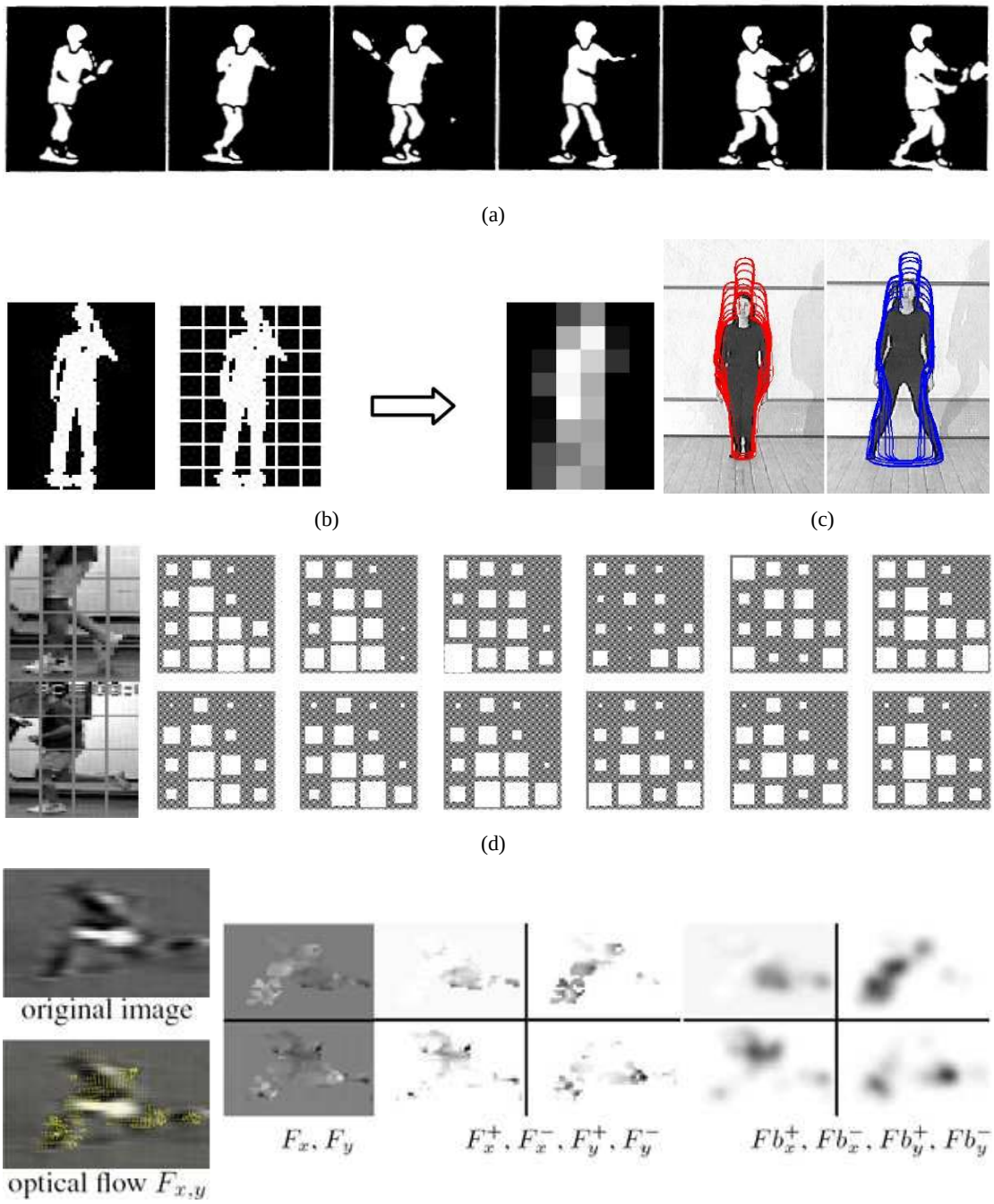


Figure 2.3.: Global posture representations: (a) Silhouettes of tennis strokes [Yamato et al., 1992]; (b) silhouettes pixels accumulated in regular grid [Wang and Suter, 2007]; (c) spline contours [Rittscher and Blake, 1999]; (d) optical flow magnitude accumulated in regular grid [Polana and Nelson, 1994]; (e) optical flow split into directional components, then blurred [Efros et al., 2003].

ground by identifying only parts that are moving. Several works use optical flow to that aim. Polana and Nelson [1992] compute several statistics based on the direction and magnitude of the normal flow. In [Polana and Nelson, 1994] flow magnitude is accumulated in a grid of non-overlapping bins. Cutler and Turk [1998] segment an optical flow field into motion blobs. Efros et al. [2003] split the optical flow field into four different scalar fields (corresponding to the negative and positive, horizontal and vertical component of the flow), which are separately matched. This representation was also used in [Robertson and Reid, 2005, Wang et al., 2007]. Also gradient fields in XYT direction [Zelnik-Manor and Irani, 2001] can be used to identify structure based on motion.

Flow based representations are typically computed over a small time window, and contain thus besides posture cues as well motion information. While not depending on a background subtraction, these approaches depend strongly on the flow computation, and consequently inherit all issues that come with these estimations, such as sensitivity to noise, color, and texture variations.

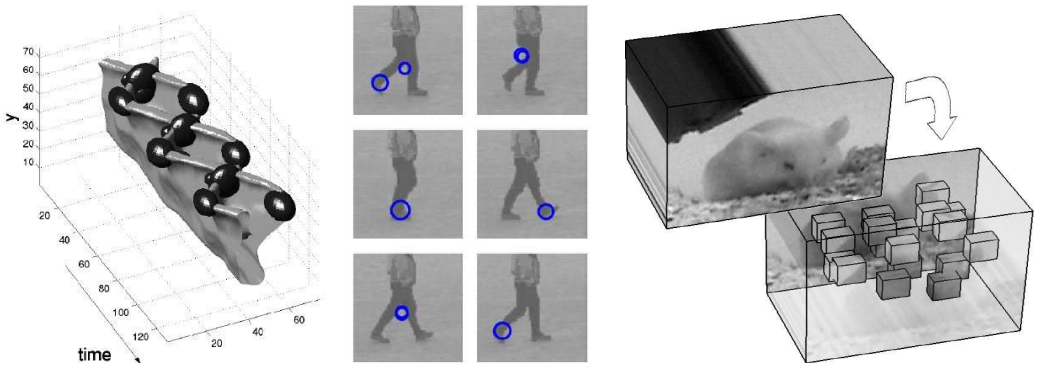
It is also possible to simply match images without any foreground and background substraction, as in the seminal work of [Darrell and Pentland, 1993], where images of hand gestures are directly correlated. This work assumes however a static black background.

Templates result in strong simplifications compared to parametric body models. Templates are, however, difficult adapted to variations in view and pose. It is thus important to account for such variation, either through a large number of different template instances, or by using suitable features and matching functions that are insensitive to view and pose transformations. While holistic approaches have been applied to scenarios of many different kind, they are sometimes advocated as being especially useful for distant-views and coarse representations, *e.g.* ”30 pixel tall” [Efros et al., 2003].

2.1.3. Local Representations

A compromise between global static templates and the highly parameterizable body models, is to decompose a global observation into smaller informative regions and to describe the regions by a set of local templates. Unlike with model based representations, the resulting interest regions are, however, *not* linked to certain body parts. Instead basic image statistics, such as high variation in tempo-spatial gradient, or homogeneity criteria, are used to locate the regions in images and videos.

Recently, so called *space-time interest points* [Laptev and Lindeberg, 2003, Dollar et al., 2005] became popular, driven by the success of interest points and local descriptors [Forstner and Gulch, 1987, Harris and Stephens, 1988, Schmid et al., 2000, Lowe, 2004] in object recognition and image classification. Such image classification approaches are typically based on



(a) (b)

Figure 2.4.: Local posture representations: (a) Space-time interest points in [Laptev and Lindeberg, 2003] are computed at points of high spatiotemporal variation (”spatiotemporal corners”). (b) Spatio-temporal features in [Dollar et al., 2005] are designed to be more responsive then the former space-time interest points.

bottom up strategies, which first detect small interest regions in the image, mostly at corner like structures, and then assign each region to a set of preselected ”vocabulary-features”. Image classification reduces then to computations on histograms, which accumulate the occurrence of such vocabulary-features. Similar interest detectors have been proposed by Laptev and Lindeberg [2003] and later by Dollar et al. [2005], see Figure 2.4, to locate informative regions in the space-time volumes spanned by video sequences. Typically, for each detected location a compact vector descriptions [Laptev and Lindeberg, 2003, Dollar et al., 2005, Scovanner et al., 2007] of the surrounding space-time cuboid is formed and assigned to a set of preselected vocabulary-features. Generally there are many analogies between these approaches and the use of so called SIFT-descriptors [Lowe, 2004] in image classification.

A huge advantage of interest point based approaches is, that no background subtraction is required for the computation of the space-time features. On the downside, the detected features are mostly unordered and of variable size, and consequently modeling geometrical and temporal structure is difficult with space-time features. Many approaches stick therefore with so called ”bags of features” representations, which, as previously mentioned, describes sequences simply through histograms of feature occurrences, without modeling any geometrical structure between the feature locations [Laptev and Lindeberg, 2003, Schuldt et al., 2004, Dollar et al., 2005, Jhuang et al., 2007, Niebles et al., 2006, Nowozin et al., 2007, Scovanner et al., 2007]. Bags of feature modeling became prominent in image classification for categorization of objects classes, such as bicycles, cars, and chairs for example. In these settings, discarding global structural information can be even advantageous, as it results in proper insensitivity to intra-class variations and view transformations. It is however not clear whether such insensitivity

to structural information is advantageous for action recognition task, where we are concerned with a single object category, *i.e*. the human body, yet with exact knowledge about its structural configuration. Recent approaches [Niebles and Fei-Fei, 2007, Wong et al., 2007] use graphical models to add structural information to local feature representations in action recognition. These attempts are however in a very early state, and currently do not lead to significant improvements in recognition results.

Besides SIFT like features, other forms of local patches have been used. Boiman and Irani [2005] try to compose newly observed space-time regions using patches extracted from previous frames of a sequence. This approach needs no supervision and can detect irregularities in videos as well as in single images. Unfortunately, it depends on an extensive combinatorial search over all possible space-time patch combinations. Another possibility to identify patches is to over-segment the space-time volume, and to explain segments through prior manually selected space-time patches [Ke et al., 2007]. In this work pictorial structures [Fischler and Elschlager, 1973, Felzenszwalb and Huttenlocher, 2000] are used to model geometric relations between the patches.

2.2. Modeling Actions

As mentioned earlier, from a machine learning view, actions consist of distributions over posture configurations, this over time. In the previous section we discussed different vision based posture representations. In this section we discuss how to model statistics over such posture representations, and in particular how to represent the temporal component of such models. We therefore distinguish between three kinds of approaches: *state-transition models*, *i.e*. approaches that explicitly model temporal evolution of posture configurations, typically through a set of finite states and temporal transitions between these states; *space-time representations*, *i.e*. approaches that implicitly model posture and time by learning static classifiers over complete sequence examples; and *Dynamics-free representations*, *i.e*. approaches that do not model temporal relations between postures.

2.2.1. State-Transition Models

We can represent dynamics over postures explicitly using a dynamical model, *i.e*. an approximation of the true dynamic system. The choice of dynamic model is generally independent from the posture representation, *i.e*. joint models or templates. A common way to approximate the dynamical system over postures is to group postures into similar configurations, *i.e*. states, and to learn temporal transition functions between these states. We name this kind of dynamics modeling *state-transition models*.

Among the versatile state-transition models used for action recognition the most prominent is certainly the *hidden Markov model* (HMM) [Rabiner, 1990]. The HMM came in particular to fame because of its great success in the speech and natural language processing community. It is a basic probabilistic finite state machine with a single state variable, whose labels typically represents different posture clusters. State transitions follow the Markovian assumption, *i.e*. the state at time t only depends on its directly preceding state at time $t-1$. The first work on action recognition using HMMs was probably [Yamato et al., 1992], where a discrete HMM is used to represent sequences over a set of vector quantized silhouette features of tennis footage. Starner and Pentland [1995] use a continuous HMM for recognition of American sign language. Bregler [1997] learns a kind of *switching-state HMM* over a set of autoregressive models, each approximating linear motions of blobs in the video frame. Other approaches using HMMs are [Wilson and Bobick, 1995, Brand, 1999, Wang et al., 2001, Green and Guan, 2004, Ogale et al., 2004, Lv and Nevatia, 2006], for instance.

Various extensions to the more general class of *dynamic Bayesian networks* (DBN) [Ghahramani, 1998] have been proposed to overcome limitations of a HMM. Brand et al. [1997] learn coupled HMMs to model interactions between several state variables. They use a two state coupled HMM to recognize interactions between left and right hand motions during Tai Chi exercises. Park and Aggarwal [2003] use a complex DBN to model interactions between two persons, such as *hugging*, *handshaking*, and *punching* for instance. Peursum et al. [2005] model interactions between people and objects in their work using Bayesian networks. Nguyen et al. [2005] propose to use hierarchical HMMs for activity recognition. Jojic et al. [2000] and Toyama and Blake [2001] extend HMMs with separate latent states for posture and view.

In recent work, Sminchisescu et al. [2005] propose to use Conditional Random Fields (CRF) instead of HMMs. They argue that CRFs can better model dependencies between features and observations overs time, because they do not depend on strong simplifying assumptions such as the HMMs. Further, CRFs are advocated for being discriminative, compared to the generative HMMs. Modelling sub-structures within actions is, however, *not* as straightforward with a CRF, as it is *e.g*. with a HMM. Therefore the basic CRF framework, proposed in [Sminchisescu et al., 2005], can only model dynamics between separate actions instances, but not within action classes. More recent works in [Wang et al., 2006a, Morency et al., 2007, Wang and Suter, 2007] overcome this problem by using additional layers of latent variables.

Other dynamic models that have been used for action recognition are: auto regressive models [Bregler, 1997, Rittscher and Blake, 1999, Bissacco et al., 2001], grammars [Ivanov and Bobick, 2000, Ogale et al., 2005], state-space approaches [Bobick and Wilson, 1997], time delayed neural networks [Yang and Ahuja, 1999], and techniques from natural language processing [Kojima et al., 2002, Thurau, 2007], among others.

Ali et al. [2007] criticize in their work the previously discussed dynamic models for making

assumption about the dynamical process, such as linearity, number of states, and Markovian assumption, *etc*., and they argue that these models only approximate the true non-linear physical process of human motion. They propose to use a chaotic system, to discover the true type of inherent dynamics.

It is also important to mention, that state space models are especially advantageous due to their high degree of modularity. Sequential models can serve as smaller vocabulary units to build larger networks of complex actions [Ikizler et al., 2007], and similarly, complex models can be used to segment sequences into smaller units [Brand and Kettnaker, 2000, Green and Guan, 2004, Peursum et al., 2004], as discussed more in detail in Section 2.4.

2.2.2. Space-Time Representations

Instead of representing posture and dynamics explicitly and separately in a layered model, space-time representations implicitly encode dynamics by directly learning the appearance of complete sequences. As for the state-transition models, space-time representations can generally use any posture representations, *i.e*. joint models or global/local templates.

Typically, space-time approaches directly represent dynamics through example sequences, either by stacking framewise features into a single feature vector, or by extracting features from the n-dimensional *space-time volume* spanned by a sequence over time. See Figure 2.5 for some examples. For instance, Blank et al. [2005] and Yilmaz and Shah [2005a] build space-time volumes by stacking multiple silhouette frames into a single volumetric representation, and extract a set of local and global features from this volumes to represent actions. In their seminal work on space-time templates, Bobick and Davis [1996b] build MHIs by mapping successive frames of silhouette sequences into a single image, and extract Hu-moments [Hu, 1962] from this representation. MHIs are generally similar to a depth map computed from a space-time volume. Polana and Nelson [1993] and Guo et al. [1994] extract Fourier coefficients from a sequence as compact descriptor.

In comparison to state-transition models, space-time representations can *not* explicitly model variations in time, speed, and action style. Such variations are instead represented through large sets of example sequences, and in combination with advanced classification techniques. As most space time representations result in fixed size vector representations, they are easily used with static classification techniques. Often simple nearest-neighbor assignment or naive Bayes classification are used in experiments, but also more advanced classification techniques have been proposed, such as Neural Networks in [Guo et al., 1994], Support Vector Machines in [Meng et al., 2007], and Adaboost in [Ke et al., 2005, Smith et al., 2005, Laptev and Pérez, 2007, Nowozin et al., 2007].

To deal with variable lengths representations, simple length based normalization or the more

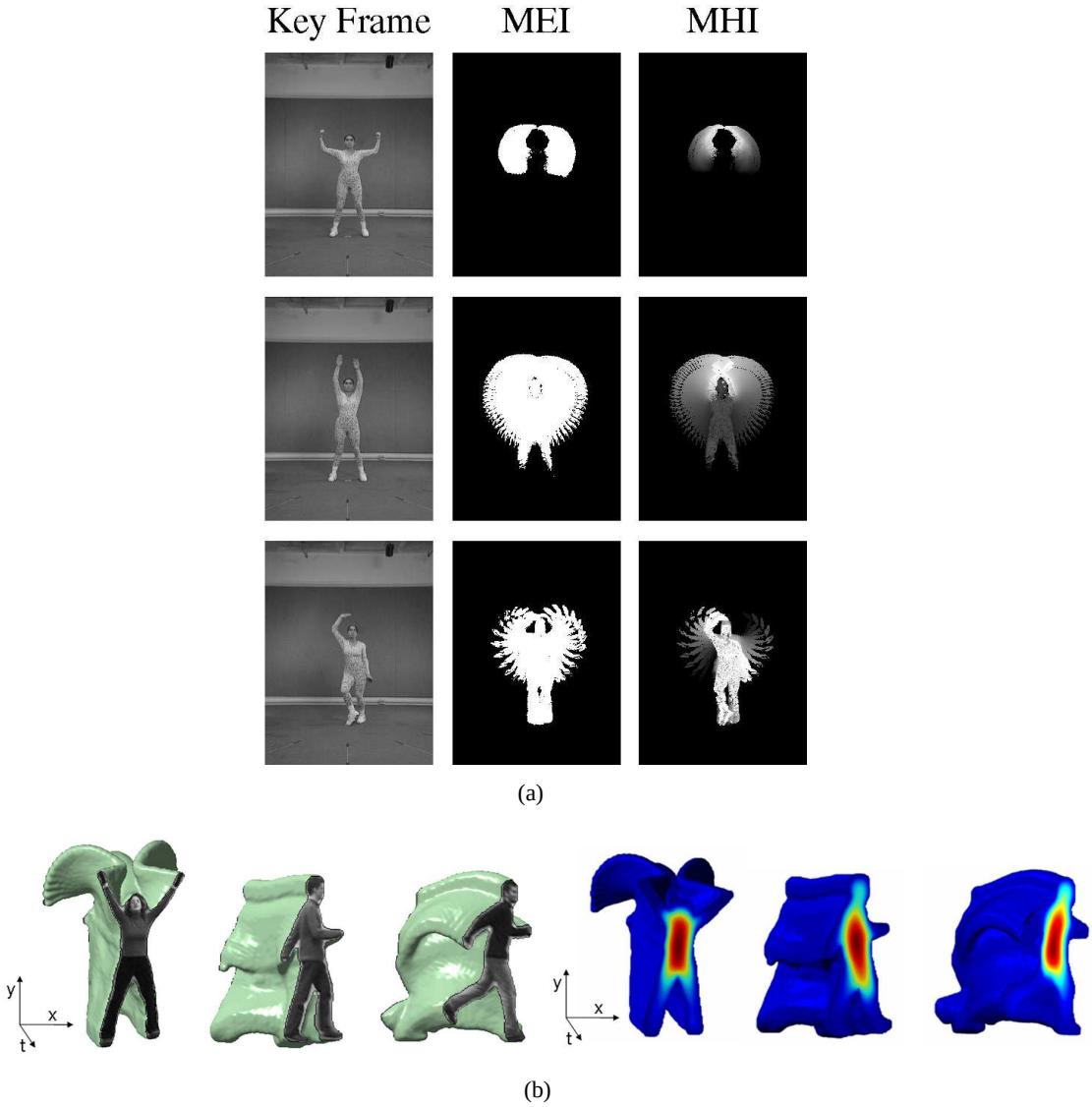


Figure 2.5.: Space-time representations: (a) motion energy images (MEI) and motion history images (MHI) [Bobick and Davis, 2001]; (b) space-time shapes [Blank et al., 2005].

advanced dynamic time warping (DTW) [Sakoe and Chiba, 1978] has been used. Darrell and Pentland [1993] correlate observations frames against a set of learned pose templates, and match the resulting sequences of correlation scores using DTW. Niyogi and Adelson [1994] and Gavrila and Davis [1995] use DTW to match sequences of joint model configurations. Veeraraghavan et al. [2006] propose a DTW method for action recognition that allows better modeling of variations within model sequences. Note that DTW can be also interpreted as a state-transition model, where each frame of the model sequence represents a sperate state, and the best sequences of the observation frames through these states is found. Under this consideration it is also important to note, that the dynamic programming algorithm used for DTW is general a non-probabilistic form of the the Viterbi path algorithm used for HMMs.

It is also important to mention, that several of the previously discussed local and global representations, *e.g*. spatiotemporal features and optical flow (Section 2.1.3 and 2.1.2), are computed over small time windows (typically 2-4 frames). These representations contain consequently as well space-time information, however, only in a very small time interval. To model complex action these descriptors need a further modeling of dynamics using the techniques discussed in this chapter.

2.2.3. Dynamic Free Representations

Several approaches do not aim to model dynamics, either by using single frames to recognize actions, or by using time independent measures, such as frequency of feature occurrence over time. While certainly not practical with all kind of actions, these approaches often argue that humans can recognize many actions from a single image — hence recognition methods should be able to do so as well. The benefit of such representations is reduced complexity and insensitive to temporal variations such as exact length and speed of actions. Nevertheless, motion contains important cues for action recognition, as demonstrated in Johansson [1973]. These approaches need thus powerful static matching techniques to compensate for the lack of motion information.

Carlsson and Sullivan [2001] introduced the use of single *key-frames*, *i.e*. characteristic frames of an action, to recognize forehand and backhand strokes in tennis recordings. The term key-frame comes originally from animation and filmmaking, where key frames define start and ending point of a smooth motion. Matching in [Carlsson and Sullivan, 2001] is based on an advanced point to point matching between edge filtered images, to measure the deformation of a edge template with respect to the image observation.

Time independent representations are also important for action/event recognition from single imagery, *i.e*. photographs. Li and Fei-Fei [2007] present an approach that combines different visual cues in a generative model to recognize sport events in static imagery, *e.g*. *badminton*,

snowboarding, *sailing*, *etc*. Another approach for unsupervised discovery of action classes from single images is proposed in [Wang et al., 2006b].

Besides single static images, sequences can be also encoded without taking temporal relations into account. Histogram techniques, *i.e*. the so called *bags of features* approaches, have been used to represent sequences simply base on the frequency of feature occurrence *e.g*. [Schuldt et al., 2004, Dollar et al., 2005, Scovanner et al., 2007, Wang et al., 2007]. The biologically motivated system of Jhuang et al. [2007] uses a different technique with feature vectors computed as maximum match responses to a set of prototypes.

Similarly, in our work on exemplar-based embedding (Section 7) we derive a new representation, which does not take temporal relations into account. In particular we found that time independent modeling of actions can often compete with much more complicated dynamical frameworks, while providing the virtues of simplicity and efficiency.

As with space-time representations, time independent representations typically result in fixed size vectors, and are therefore often used in combination with advanced classification techniques, such as SVMs in [Schuldt et al., 2004, Jhuang et al., 2007] and Adaboost in [Nowozin et al., 2007].

2.3. View Independent Representations

As mentioned earlier (Section 2.1.1), fundamental considerations on the model representation, *i.e*. wether to use a 2D or 3D representation, have a long history in action recognition. Early psychophysical experiments [Johansson, 1973] where asking whether humans use structure from motion reconstruction to recognize actions, or whether they recognize actions directly from 2D motion patterns. Besides action recognition, paradigms on reconstructive vision vs. purposive vision [Aloimonos, 1990] where generally popular in the vision community.

Approaches demonstrating general qualities of either direction (2D or 3D model) have been proposed. Following the initial success of those approaches, new challenges, such as learning larger number of action classes and robustness under more realistic settings, gained importance. Within this scope, a very important demand is independence to viewpoint, which wasn't address by most of the early approaches. It is our opinion, that such considerations bring the issue on how to represent posture, *i.e*. in 2D or 3D, into a interesting new perspective. In the following discussion we will therefore in particular focus on the different view representations used by action recognition approaches, *i.e*. 2D, multi-view, or 3D.

We take our taxonomy for view-independent action recognition from work on shape matching: Kazhdan [2004] names three strategies for view-independent matching: normalization, invariance, and exhaustive search. *Normalization* maps observations from different views into a common canonical coordinate frame; matching is then performed under this canonical setting.

Invariance uses features that do not depend on view transformations, such that the resulting match is the same for any view transformation. *Exhaustive search* takes all possible view transformations into account, and searches for the optimal match within these. All these strategies apply as well to action recognition, with the additional difficulty that the viewpoints themselves may change over time. In the following we discuss approaches based on these strategies, and further, as mentioned previously, separate between view representations in 2D, multi-view, or 3D.

2.3.1. Normalization

In normalization, each observation is mapped to a common canonical coordinate fame. Therefore normalization approaches generally first estimate cues that indicate the transformation from the canonical frame to the current state of the observation, and then correct the observation with respect to the estimated transformation. Matching then takes place after the observations have been normalized.

Normalization in 2D

Normalization is used by many approaches as a preprocessing step to remove global scale and translation variations. In particular global representation, *e.g*. silhouette base approaches (Section 2.1.2), often extract a rectangular region of interest (ROI) around the silhouette, and scale and translate this region to a unit frame. This normalization removes global variations in body size, as well as some scale and translation variations resulting from perspective changes.

Normalization with respect to out-of-plane transformations, *e.g*. a camera rotation, is not trivial given a single 2D observation. Nevertheless, Rogez et al. [2006] propose a method, which estimates the 3D orientation of a person from its walking direction in 2D, using knowledge about the ground homography and camera calibration. Assuming only horizontal rotation of the body in 3D, the 2D silhouette of the person is perspectively corrected onto a fronto parallel view and matched against a set of canonical silhouettes.

Normalization in 3D

Although it strongly limits the application of action recognition approaches, walking direction as orientation cue was, as well, used by several 3D based approaches to compute a reference frame for normalization. Bodor et al. [2003] use multiple views to compute a 3D voxel reconstruction of a walking person. The walking direction is then used to back-project the person silhouette on a view orthogonal to the walking direction, and action recognition is performed on the resulting silhouettes. Also Cuzzolin et al. [2004] align voxel grids of human bodies us-

ing their waking direction. After normalization, they perform action recognition on velocities of body part estimates. Roh et al. [2006] extend MHIs [Bobick and Davis, 1996b] to disparity maps. An estimated global flow direction is used in this work to align the 2.5D MHIs. Also several joint model based approaches use an estimated walking direction to estimate the initial model, *e.g.* [Zhao and Nevatia, 2002, Peursum et al., 2007].

Given a 3D joint body model, an orientation independent joint representation can be compute based on the global body structures. Often the torso is used as reference object to normalize all joints with respect to its orientation. It is further possible to represent each body part with and individual coordinate frames. For example, Gavrila and Davis [1995] compute individual reference frames for the torso, arms, and hips.

In summary, normalization approaches are based on the estimation of the body orientation. Consequently, all following phases depend on the robustness of this step. Miss alignments, because of noise or intraclass variations, are likely to affect all following phases of the approach.

2.3.2. View Invariance

View-invariant approaches do not attempt to estimate view transformations between model and observation. Instead view-invariant approaches search for features and matching functions that are independent (*i.e.* do not change) with respect to the class of view transformations considered.

View Invariance in 2D

A simple form of view-invariance is based on histogramming. Instead of representing image features in a fixed grid, only the frequency of feature occurrences is stored. Such an representation has been used for instance by Zelnik-Manor and Irani [2001] to represent distributions of space-time gradients. This representation, however, only provides invariance to transformations in the image plane.

The availability of point correspondences, *e.g.* in form of anatomical landmarks, was frequently used for view-invariant matching between pairs of observations, see Figure 2.6 for some examples. For instance, an epipolar geometry can be estimated from a subset of point correspondences, and then used to constrain the set of all point correspondences, and respectively a matching cost over changing views can be computed without requiring a full 3D reconstruction. *I.e.* given point matches (x_i, x'_i), $i = 1, \ldots, n \geq 8$ in pairs of images I, I', the fundamental matrix F, which holds the relation $x_i F x'_i = 0$, can be estimated. This relation holds however only if all point pairs come from the same rigid object. Hence the resulting residual $\sum_i |x_i F x'_i|^2$ can be used as matching cost [Syeda-Mahmood et al., 2001, Gritai et al., 2004, Sheikh and Shah, 2005, Yilmaz and Shah, 2005a,b]. Similar, matrix factorization and rank constraints, as in structure from motion estimation [Tomasi and Kanade, 1992], can be used to validate whether point

correspondences in two images came from the same single rigid object [Seitz and Dyer, 1997, Rao et al., 2002, 2003a,b].

Geometric invariants, *i.e.* measures that do not change under a geometric transformation, can also been used for invariant matching of landmark points. These invariants can be computed from 5 points that lie in a plane. Parameswaran and Chellappa [2003, 2005, 2006] detect joint configuration during walking cycles that fulfill the condition of 5 landmarks lying in a common plane, and use these to compute geometric invariants.

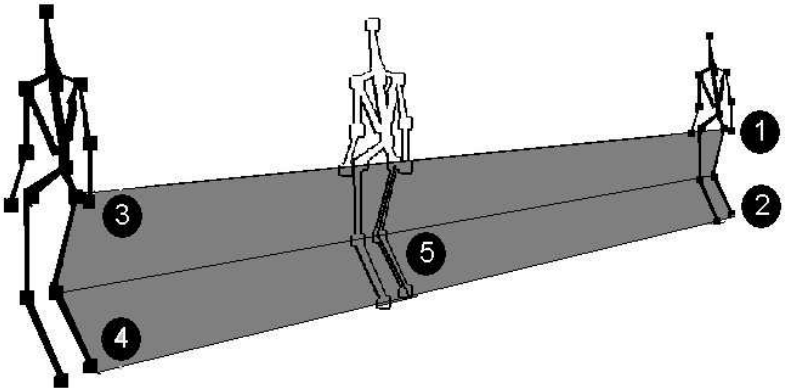
All these approaches are based on the assumption that point correspondences are available in pars of images. With exception in [Rao et al., 2002, 2003a,b, Yilmaz and Shah, 2005a], the problem of how to compute such points is not addressed by these approaches. Another drawback of these approaches is, that the matching can only be computed between pairs of image observations, and it is difficult to extend the matching to more general class representations. Further note, that although these approaches are single view 2D, computing the fundamental matrix and structure from motion factorizations are already first steps towards a 3D reconstruction.

View Invariance in 3D

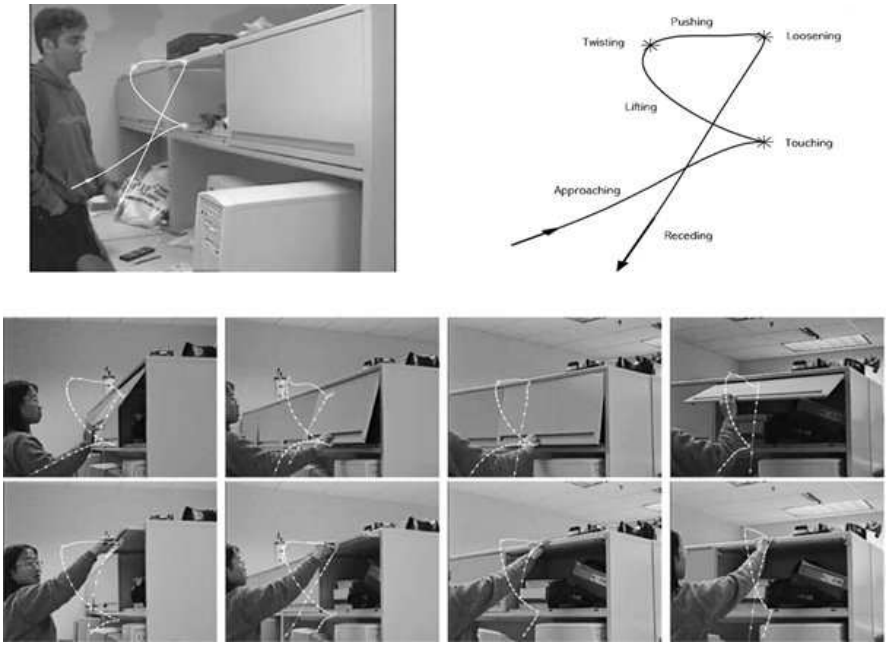
Based on 3D body part trajectories Campbell et al. [1996] investigates 10 different view-invariant representations in their work. These include shift invariant velocities (dx, dy, dz) in cartesian coordinates, and shift and horizontal rotation invariant velocities $(dr, d\theta, dz)$ in polar coordinate. In evaluation on 18 Tai Chi gestures, the polar coordinate representation has best overall recognition rates.

There are few other view-invariant approaches in 3D, and especially few approaches that do not depend on a joint body model or point correspondences. An exception is the work of Cohen and Li [2003], which proposes a view invariant pose representation based on a voxel reconstruction. Cylindrical 3D histograms (Figure 2.7) similar to the 2D shape context descriptor [Belongie and Malik, 2000], are used as invariant measure of the the voxel's distribution in this work. The same descriptor was later used by Pierobon et al. [2006] for action recognition. This representation, however, only applies to *binary* voxel grids.

One of the contributions of our work is a view-invariant representation based on Fourier coefficients in cylindrical coordinates, that applies to any *multi-valued* voxel representations, *e.g.* in our work a 3D extension of MHIs. Fourier descriptors are well known as invariant shape representations, *e.g.* [Granlund, 1972, Zahn and Roskies, 1972, Otterloo, 1991]. Similar, the closely related spherical harmonics (broadly speaking: Fourier components on a sphere) have been proposed for invariant shape retrieval by Kazhdan et al. [2003]. To our knowledge, no previous work exist on using invariant fourier descriptors for action recognition.



(a)



(b)

Figure 2.6.: View invariant action recognition: (a) geometrical invariants can be computed from 5 point that lie in a plane [Parameswaran and Chellappa, 2003]; (b) View-invariant matching of hand trajectories [Rao et al., 2003b]. Point matches between different observations are computed from discontinuities in motion trajectories.

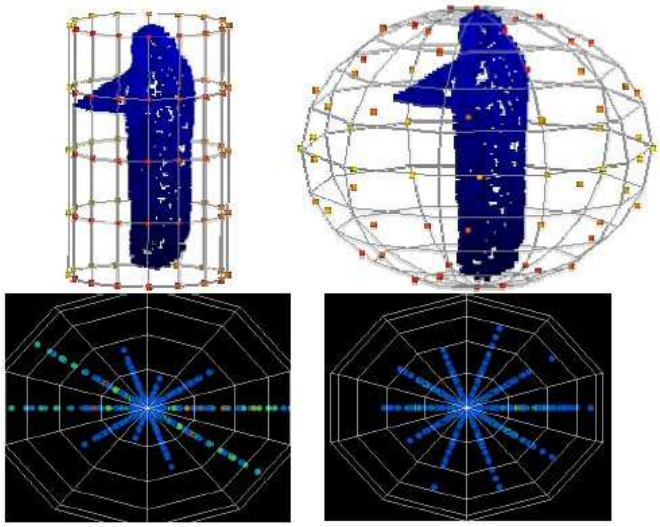


Figure 2.7.: View invariant visual hull distributions [Cohen and Li, 2003]: (Left) invariant to rotation along vertical axis using a cylindrical surface. (Right) 3D rotation invariance using a sphere.

In parallel to our work, another view invariant MHI representation has been propose. Canton-Ferrer et al. [2006] proposes to compute MHVs from voxel reconstructions, similar to our work. However, they chose a different invariant representation based on 3D moments [Lo and Don, 1989].

2.3.3. Exhaustive Search

Instead of deciding on a single transformation, as it is typical for normalization methods, or discarding all transformation dependent information, as with invariant methods, one can search over all possible transformations considered. At first sight, an exhaustive search may seem heavy on computational resources. Yet, reasonable assumptions, such as restrictions to certain classes of transformations, advanced search strategies, and propagation of findings over time, can drastically reduce the search space. Moreover, with the steadily increasing performance of modern computer systems, the computational expense of such methods is about to become fairly manageable.

Exhaustive Search using Multiple 2D Views

Several approaches use a fixed set of cameras installed around the actor, and simultaneously record the actions from this multiple views. During recognition, an observation is then matched against each recorded view and the best matching pair is identified. In their work on MHIs,

Bobick and Davis [2001] record actions with 7 cameras, each with an offset of 30° in the horizontal plane around the actors. During recognition two cameras with 90° offset are used, and matched against all pairs of recorded views with the same 90° offset. An action is then labeled with respect to the best average match of two cameras. Similar Ogale et al. [2004, 2005] use 8 prerecorded views, and a single view during recognition. Their work uses a single HMM to mode temporal relations between prototype silhouettes and view changes over time. Also Ahmad and Lee [2006] use 8 prerecorded views. In their work individual HMMs for each view are learned without transitions between close views. Consequently, smooth view changes, *e.g.* a person slowly turning around while performing an action, can not be recognized.

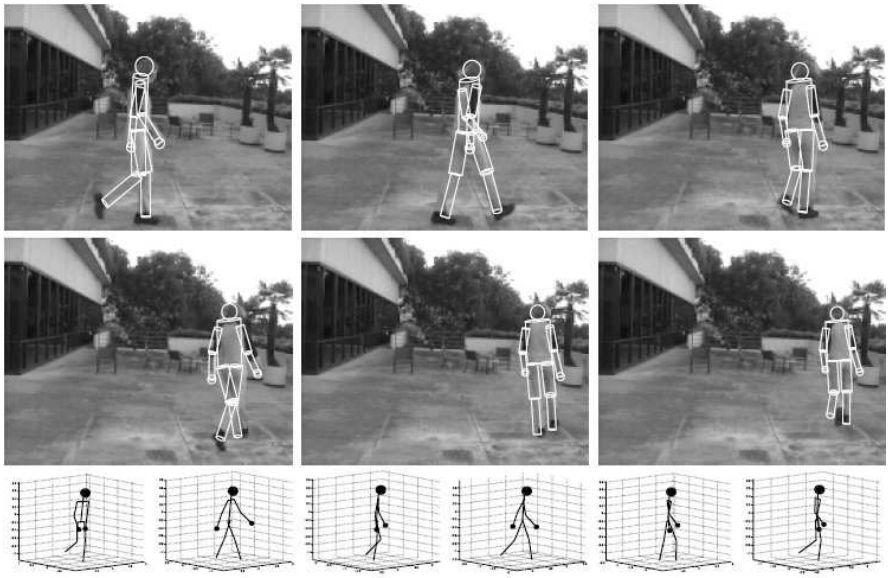
All these approaches only work with a limited set of prerecorded views; any camera configuration that was not explicitly recorded during learning can not be modeled.

Exhaustive Search using a 3D Generative Model

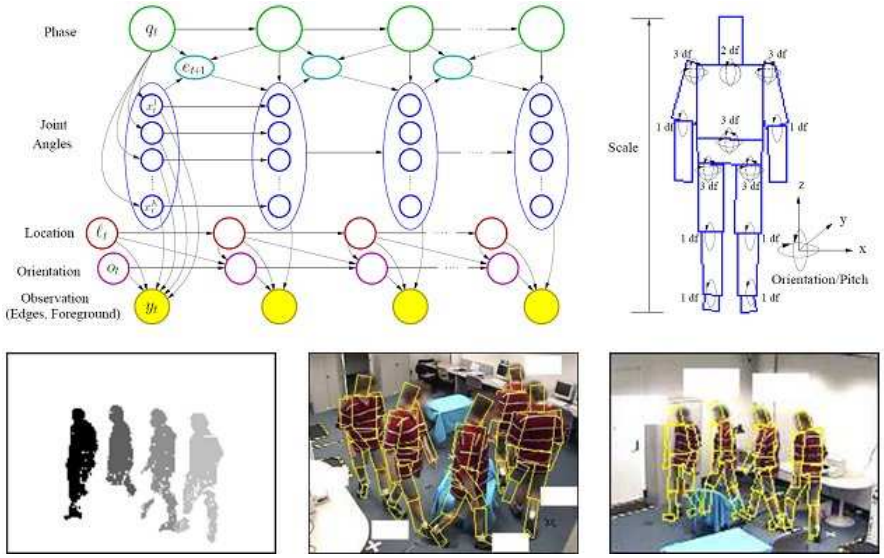
To achieve more flexibility with respect to changes in camera setup, an internal model based on a 3D representation can be used. From such a 3D representation, and given camera parameters, any possible 2D view observation can be rendered. Generative approaches are frequently used in MOCAP, where parameterized 3D models of the human body are projected into 2D. These models have explicit variables for global 3D position and orientation, that are estimated simultaneously with the remaining joint parameters, *e.g.* [Deutscher et al., 2000, Sidenbladh et al., 2000, Peursum et al., 2007], see also Figure 2.8.

Interestingly, similar methods haven not been proposed for action recognition, and in particular not without using a representations based on a joint model. An exception is the key-pose approach by Lv and Nevatia [2007], that has been developed in parallel to our work. In this work a small set of synthetic 3D key-poses is rendered from a modeling software. Actions are then matched against the poses, which are projected into 2D with respect to all possible transformations. Dynamics over poses and changes in view transformations are modeled in a dynamic network, and the best pose-view sequence is found via a dynamic programming search. Interestingly, this approach which was developed in parallel to our work on exemplar-based HMMs (Section 6), shares the idea of projecting a set of learned 3D exemplars/keyposes into 2D to infer actions from arbitrary view. However, the work that we will present in this thesis, uses a probabilistic formulation instead of the deterministic linked action graph introduced in [Lv and Nevatia, 2007]], allowing therefore to naturally handle uncertainties inherent to actions performed by different people and with different styles.

In another context, Frey and Jojic [2000], Jojic et al. [2000], Toyama and Blake [2001] propose HMMs with additional capabilities to model similarity transformations in the viewing plane. Such HMMs allow to compute transformation independent observation probabilities by



(a)



(b)

Figure 2.8.: Generative MOCAP: (a) Tracking a person walking in cycle [Sidenbladh et al., 2000]. (b) Factor-state hierarchical HMM body model and tracking results using the generative approach in [Peursum et al., 2007].

marginalizing over all possible transformations. Computing marginals is indeed a form of exhaustive search, except that no deterministic decision are made. Instead a probability taking all possible search results into account is computed. The exemplar-based HMMs proposed in this thesis is an extension of such a framework for view independent action recognition. To our knowledge, such a framework hasn't been previously used to model perspective transformations in action recognition.

2.4. Action Recognition in the Real World

Most of the previously discussed action representations where designed for classification of single, isolated action units. In this section we discuss some of the techniques that are necessary to apply such representations to continuous streams of actions, as it is necessary for real world applications: *Temporal segmentation* is necessary to cut streams of motions into single action instances that are consistent to the set of initial training sequences used to learn the models. Closely related are the questions: how to choose such initial segmentations; and is there something like an elementary vocabulary of primitive motions in action articulation and perception? To address this questions we will discuss several works that aim at building *action taxonomies*. Another problem that arises on realistic sequences is, that we very unlikely can learn models for all possible motions that may appear. We need thus techniques to *spot* actions, *i.e.* a technique that identifies important actions within utterances of unimportant actions. Further, we need models for these unimportant actions, so called *filler* or *garbage* models.

2.4.1. Temporal Segmentation

We categorize methods for temporal segmentation into three classes: boundary detection, sliding window, and dynamic programming approaches. *Boundary detection* methods explicitly search for features in the motion sequences that characterize start and end points of actions. Such boundaries are for example discontinuities or extrema in acceleration, velocities, and curvature. *Sliding window* based methods correlate prior learned action models against the observation stream, and detect actions by searching for peaks in the correlation score. *Dynamic programming* approaches use state-transition representation of actions that explicitly model transitions between the different action models. Dynamic programming techniques, such as the Viterbi path, are then used to label a observation by identifying the best sequence of that observation through the set of action states. The approaches are described in detail in the following.

Boundary Detection

As mentioned earlier, motion boundaries are typically defined as discontinuities and extrema in acceleration, velocities, or curvature of the observed motions. The choice of boundaries thus implicitly results in a basic motion taxonomy.

Marr and Vaina [1982] discussed in their seminal theoretical work the problem of segmenting the 3D movement of shapes and suggest the use of rest states, *i.e.* local minima, of the 3D motion of human limbs as natural transitions between primitive movements. Similar, Rubin and Richards [1985] define in their work two elementary kinds of motion boundaries: *starts and stops* and *dynamic boundaries*. *Starts and stops*, are boundaries that occur whenever a motion changes from a moving state into a rest state, and vice versa. Starts and stops are thus analog to the rest states defined by Marr and Vaina [1982]. Furthermore, *dynamic boundaries* result from discontinuities, such as steps and impulses, in force applied to the object in action, and fall thus generally within the reference frames defined by starts and stops.

Computational approaches for motion boundary detection are that of Rui and Anandan [2000]. They perform an SVD decomposition of a long sequence of optical flow images and detect discontinuities in the trajectories of selected SVD components to segment video into motion patterns. Similar, Ogale et al. [2004] cluster action sequences by detecting minima and maxima of optical flow inside body silhouettes. Instead of factorizing flow into different components, their method only uses the average global flow magnitude. Base on 2D trajectories of hand gestures, Wang et al. [2001] search for local minima of velocities and local maxima of change in direction. Similar Rao et al. [2002] examine trajectories of hand motions, see Figure 2.6(b). Kahol et al. [2003] use a hierarchical body model and detect local minima in force, momentum, and kinetic energy of the joints. They compare their segmentation results on professional choreographed dances.

In our work on motion history volumes (Chapter 4), we propose a segmentation of actions into primitives based on 3D motion velocity minima. In contrary to previous work, this representation does not depend on flow commutations or availability of joint trajectories, and is directly computed from the global 3D MHV representation. To the best of our knowledge, no previous work has attempted to perform motion segmentation from *volumetric* reconstructions.

An advantage of boundary detection methods, is that segmentation does not depend on prior learned action models. As we will see in our work on MHVs, such independent segmentation can be used to generates action taxonomies purely from visual cues.

Sliding Window

Sliding window approaches segment video streams based on prior learned action models. Based on an empirically chosen length and stepsister, a sequence is divided into multiple overlapping

subsequences. Each subsequence is matched against all learned actions models, and peaks in the resulting correlation scores are assumed as possible action positions. The segmentation depends thus strongly on the sequences used to train the models, what is in contrary to boundary detection, where the action models result from the choice of boundary segmentation criteria.

A sliding window approach can be used with any of the previously discussed spatial and temporal representations. The space-time approaches [Zelnik-Manor and Irani, 2001, Zhong et al., 2004, Feng and Cham, 2005, Ke et al., 2005, 2007] use a sliding window. There are as well approaches that align sequences based on DTW [Darrell and Pentland, 1993, Morguet and Lang, 1998, Alon et al., 2005], and HMM based representations [Bobick and Ivanov, 1998, Wilson and Bobick, 1999].

Compared to boundary detection methods, sliding window methods are more expensive, as they involve per frame evaluation of all models. To achieve robustness against duration of an action, often multiple window length are used, resulting in an additional multitude of evaluations. Also, as already mentioned, sliding window method can only segment known actions, in contrary to the boundary detection methods that segment independent of action models.

Dynamic Programming

In the previous section we discussed some state-transition models, such as HMMs, that used a sliding window to segment actions. In these setups isolated models for each action are evaluated independently for each windowed observation, and labeling follows a maximum a-posteriori rule. Another possibility to segment sequence using state-transition models is to build a single network from combination of all individual action models. Such networks can be build for instance by joining all models in a common start and end node and by adding a loop-back transition between these two nodes. It is also further possible to allow for more complex transitions between actions, *e.g*. actions may share states and transitions between actions may be adjusted individually to reflect realistic probabilities of actions following each other. Such complex structure are similar to HMMs networks used in continuous speech recognition. Segmentation and labeling of a complex action sequence is then computed as a minimum-cost path trough the network using dynamic programming techniques, *e.g*. the Viterbi path for HMMs [Rabiner, 1990]. The works [Brand and Kettnaker, 2000, Green and Guan, 2004, Peursum et al., 2004, Lv and Nevatia, 2006] use such networks for action recognition based on HMMs. Similar Sminchisescu et al. [2005], Morency et al. [2007] use CRFs. The work of Rittscher and Blake [1999] uses autoregressive models to represent actions, and a condensation filter to switch between these models.

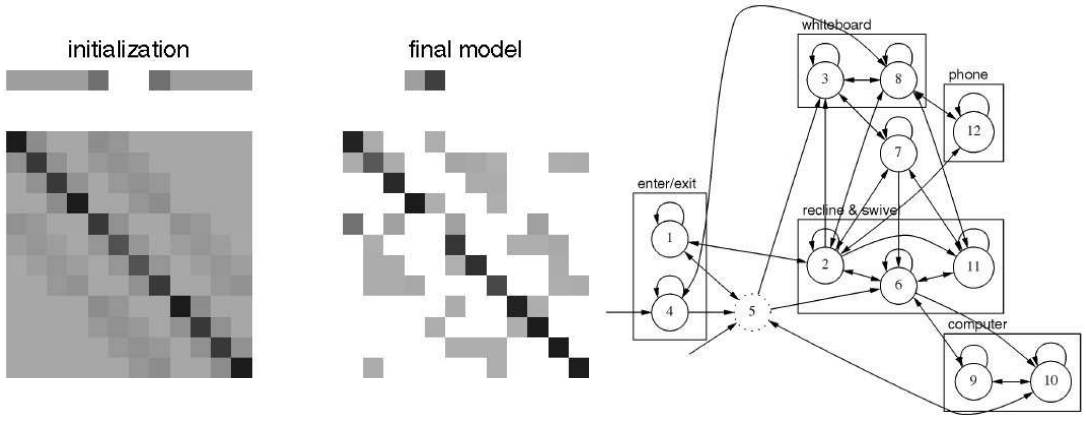
One problem with these aparchies is, however, that a meaningful learning of the complex networks structures requires an enormous amount of training data, especially when transitions

between actions are learned from real data. In speech recognition such data is available in form of text-documents, word-transcriptions, and phonetically labeled sequences. Similar data does, however, currently not exist for action recognition, and therefore transitions between actions are often set manually, or strong assumption, such as uniform transition probabilities, are made. Under those assumptions the segmentation quality of these models is equivalent to sliding window approaches over the individual models. Nevertheless, the dynamic programming technique can be advantageous, as it efficiently avoids the exhaustive search over all subsequences that is inherent to the sliding window approaches.

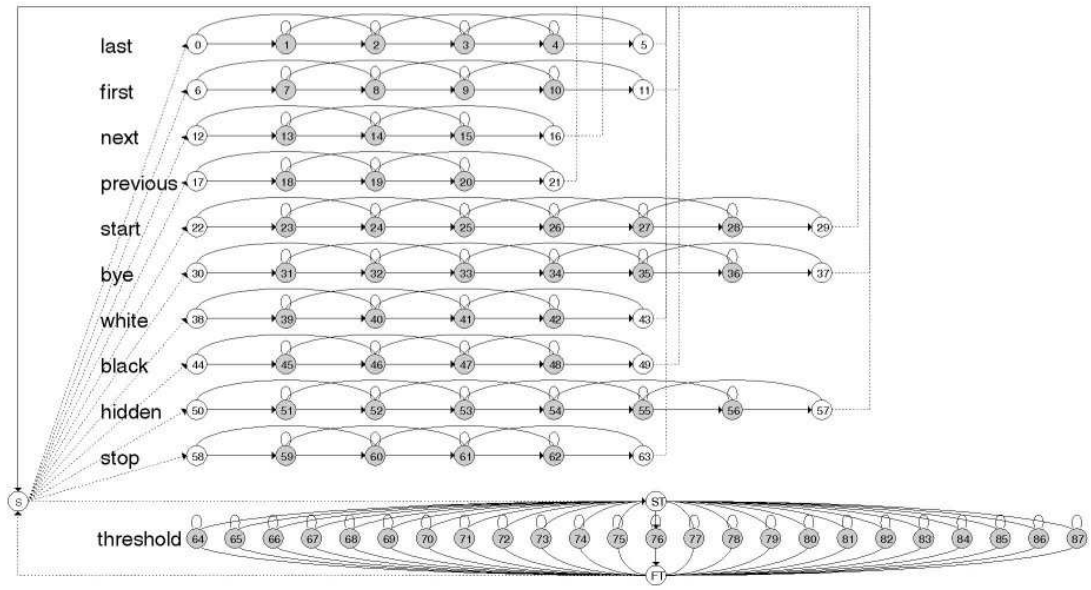
2.4.2. Action Taxonomies

As mentioned earlier, many models in action recognition are derived from techniques in speech recognition. In speech, the elementary ingredients to build such models, *i.e.* definitions of speech primitives, vocabularies, and grammars are well defined. Similar commonly accepted taxonomies do not exist for motions and actions. Independent proposals for motion primitives have been made. Bregler [1997] introduces *movemes* as the complement to *phonemes* in speech. *Movemes* are basic building blocks of actions words, that can be approximated with a linear system. Similar, Green and Guan [2004] chose the name *dyneme*. In their work they build HMM networks based on an empirical chosen alphabets of 35 *dynemes*. Another work which addresses building motor primitives in joint space is that of Guerra-Filho and Aloimonos [2007]. In this work, primitives of kinetic origin are named *kinetemes*. In computer graphics, Rose et al. [1998] introduce the concept of *verbs and adverbs* to interpolate new motions from example motions. In the work of Arikan et al. [2003], the user can define a set of motion primitives, which is then used to synthesize new composite motions.

Besides manually defining action taxonomies, several approaches attempt a purely data driven discovery of motion primitives. Brand and Kettnaker [2000], see Figure 2.9(a), start from a fully connected HMM to represent a continuous action sequence. An entropy based minimization is then used to discover independent structures within the HMM by pruning most of the transitions. They evaluate their method on single blob trajectories of office activity and outdoor traffic. Wang et al. [2001] segment hand gestures using boundary detection. For each segment a separate HMM is learned and a distance defined between pairs of HMM allows them to hierarchical cluster these HMMs. Zelnik-Manor and Irani [2001] compute normalized cuts on correlation matrixes of action sequences to cluster these sequences. Vasilescu [2002] propose a SVD techniques for 3rd order tensors, to factorize $\mathbb{R}^{\text{people}\times\text{action}\times\text{time}}$ joint-coordinate tensors into *motion signatures*. Jenkins and Mataric [2002] use a spatio-temporal extension of the ISOMAP embedding method [Tenenbaum et al., 2000] to discover motor-primitive from MOCAP data.



(a)



(b)

Figure 2.9.: Primitive discovery and garbage modeling using HMMs: (a) Images from [Brand and Kettnaker, 2000]. Transition probabilities and initial state probabilities of a fully connected HMM, (Left) before and (Middle) after structure discovery using entropic estimation. (Right) resulting graphical model (b) HMM network with additional garbage-model (threshold model). The garbage model is build as an ergodic fully connected HMM based on copies of all states in the original models from [Lee and Kim, 1999].

The work on MHVs we present in this thesis provides a similar unsupervised discovery of motion primitives. In contrary to previous work, our approach is purely based on visual-cues, and does not require extraction of a body model or point trajectories.

2.4.3. Action Spotting And Garbage Models

When working with realistic sequences we will be confronted with an almost infinite number of possible actions. Many of these actions will be meaningless motion utterances that are not critical for the application context. Nevertheless, it is critical that we detect them as meaningless. There are many issues here. One is the size of the vocabulary of human actions. In speech recognition also, there are problems dealing with large vocabularies. But to be fair, we are not there yet in action recognition. The first thing is to recognize that we can only deal with a small vocabulary of actions. Thus, we can either limit ourselves to situations with small vocabularies, such as sign language, or take care of actions that cannot be recognized because they were not learned. This is similar to the distinction between *open world* vs *closed world* assumptions is AI.

In analogy to speech recognition, we can distinguish in action recognition between *word-spotting approaches*, *i.e.* approaches that *detect* a small set of important actions within a large corpus of unknown actions, and *continuous recognition*, *i.e.* approaches that *classify* observations, under the assumption that all observed actions belong to a fixed vocabulary of known and previously learned actions. Most current approaches in action recognition only address the latter classification scenario. Typically a small fixed set of actions models ($\approx$ 3-10) is learned and each observation is assigned to one of these models. In speech, similar approaches are meanwhile able to cover vocabularies as large as complete language models. Although attempts to learn action vocabularies exist (see Section 2.4.2), it is arguably whether these will ever be as complete as speech vocabularies. Especially if we consider that there is an almost infinite number of meaningless actions. Moreover, real world applications of action recognition, *e.g.* surveillance, often depend on detection of a few important actions, *e.g.* aggression. Spotting approaches, that distinguish between key-actions and unimportant actions, are thus an interesting alternative to continuous action recognition.

Some classification approaches can be easily extended to spotting by thresholding model distances or similarity scores. Thresholding was used for example in [Darrell and Pentland, 1993, Alon et al., 2005, Ke et al., 2005, 2007]. For approaches that allow for variable length observations, however, appropriate normalization of the matching scores has to be taken into account. Typical examples are HMMs, where the observation likelihood continuously decreases with increasing length of a sequence, and consequently a fixed threshold can not be applied. An approach that takes such a specialized normalization into account is [Morguet and Lang, 1998].

An alternative to length based normalization is to explicitly model the non action class using a so called *filler* or *garbage* models [Rohlicek et al., 1989, Rose and Paul, 1990]. There are few works in action recognition that attempt to learn such a model. An exception is the work of Lee and Kim [1999] on gesture recognition. Here HMMs are learned for each action and an additional garbage model is build as an ergodic fully connected HMM, based on copies of all states in the original models, see Figure 2.9(b).

Part I

Action Recognition in 3D: Motion History Volumes

This part introduces Motion History Volumes (MHV) as view-invariant space-time representation for human actions. MHVs are computed in 3D, based on visual hull reconstructions from multiple calibrated and background-subtracted video cameras.

In chapter 3, we present algorithms for computing, aligning and comparing MHVs of different actions performed by different people in a variety of viewpoints. Alignment and comparisons are performed efficiently using Fourier transforms in cylindrical coordinates around the vertical axis. Results indicate that this representation can be used to learn and recognize basic human action classes, independently of gender, body size and viewpoint.

In chapter 4, we present a new method, based on MHVs, for segmenting actions into primitives and classifying them into a hierarchy of action classes. Because our representation is independent of viewpoint, it results in segmentation and classification methods which are surprisingly efficient and robust. Our new method can be used as the first step in a semi-supervised action recognition system that will automatically break down training examples of people performing sequences of actions into primitive actions that can be discriminatingly classified and assembled into high-level recognizers.

CHAPTER 3

Motion History Volumes

In this chapter we introduce a new motion descriptor, the *motion history volume* (MHV), which fuses action cues, as seen from different viewpoints and over short time periods, into a single three dimensional space-time representation. MHVs are derived as a 3D extension of *motion history images* (MHI), which were originally proposed by Bobick and Davis [1996b]. We use therefore multiple cameras and shape from silhouette techniques.

Based on MHVs we investigate how to build models of human actions that can support categorization and recognition of simple action classes, independently of viewpoint, actor gender and body sizes. The key to our approach is the assumption that we need only consider variations in viewpoints around the central vertical axis of the human body. Accordingly, we propose a view invariant representation based on Fourier analysis of MHVs in a cylindrical coordinate system.

Figure 3.1 explains our method for comparing two action sequences. We separately compute their visual hulls and accumulate them into motion history volumes. We transform the MHVs into cylindrical coordinates around their vertical axes, and extract view-invariant features in Fourier space.

The chapter is organized as follows. First, we recall Davis and Bobick's definition of motion templates and extend it to three dimensions in Section 3.1. We present efficient descriptors for matching and aligning MHVs in Section 3.2. We present classification results in Section 3.3 and conclude in Section 3.4.

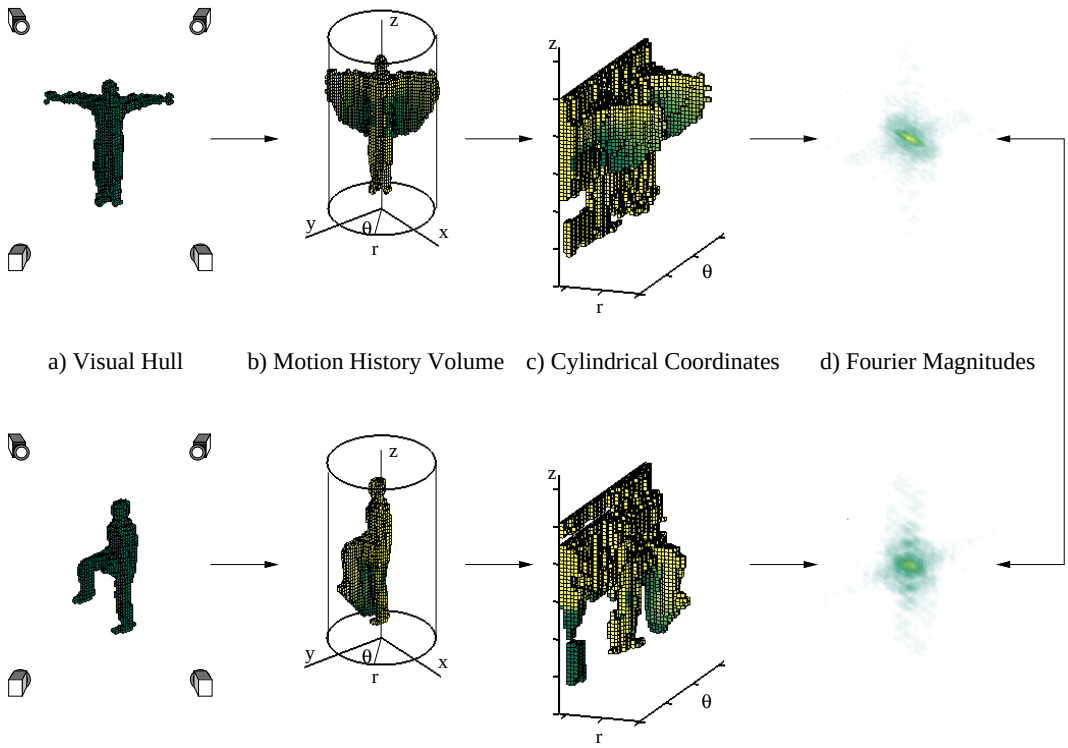


Figure 3.1.: The two actions are recorded by multiple cameras, spatially integrated into their visual hulls (a), and temporally integrated into motion history volumes (b)(c). Invariant motion descriptors in Fourier space (d) are used for comparing the two actions.

3.1. Definitions

In this section, we first recall 2D motion templates as introduced by Bobick and Davis [1996b] to describe temporal actions. We then propose their generalization to 3D in order to remove the viewpoint dependence in an optimal fashion using calibrated cameras.

3.1.1. Motion History Images

Motion Energy Images (MEI) and Motion History Images (MHI) [Bobick and Davis, 1996b] were introduced to capture motion information in images. They encode, respectively, where motion occurred, and the history of motion occurrences, in the image. Pixel values are therefore binary values (MEI) encoding motion occurrence at a pixel, or multiple-values (MHI) encoding how recently motion occurred at a pixel. More formally, consider the binary-valued function $D(x, y, t)$, $D = 1$ indicating motion at time t and location (x, y), then the MHI function is defined by:

$$h_\tau(x, y, t) = \begin{cases} \tau & \text{if } D(x, y, t) = 1 \\ \max(0, h_\tau(x, y, t-1) - 1) & \text{otherwise,} \end{cases} \tag{3.1}$$

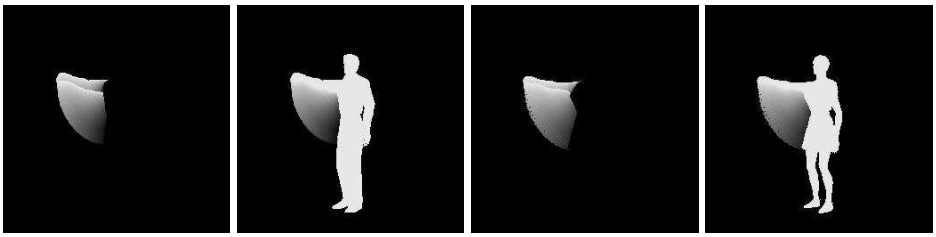
where τ is the maximum duration a motion is stored. Intuitively, the MHI correspond to the depth map computed along the time axis of the space-time volume spanned by function D. The associated MEI can easily be computed by thresholding $h > 0$.

The above motion templates are based on motion, i.e. $D(x, y, t)$ is a motion indicating function, however Bobick and Davis also suggest to compute templates based on occupancy, replacing $D(x, y, t)$ by the silhouette occupancy function. They argue that including the complete body makes templates more robust to incidental motions that occur during an action. Our experiments confirm that and show that occupancy provides robust cues for recognition, even if occupancy encodes not only motion but also shapes which may add difficulties when comparing movements, as illustrated in Figure 3.2.

In [Bobick and Davis, 2001] invariance to in-plane rotations and scaling is achieved via Hu moments [Hu, 1962], which are only a crude representation. View invariance with respect to out-of-plane rotation requires then many images from different viewpoints and it becomes unclear how to decide which action is seen. We instead compute a single rotation invariant motion history representation in 3D, as explained in the next sections.

3.1.2. Motion History Volumes

In this section, we extend 2D motion templates to 3D. The choice of a 3D representation has several advantages over a single, or multiple, 2D view representation:



(a) (b) (c) (d)

Figure 3.2.: Motion versus occupancy. Using motion only in image (a), we can roughly gather that someone is lifting one arm. Using the whole silhouette instead, in (b), makes it clear that the right arm is lifted. However the same movement executed by a woman, in (c), compares favorably with the man's action in (a), whereas the whole bodies comparisons between (b) and (d) is less evident.

- A 3D representation is a natural way to fuse multiple images information. Such representation is more informative than simple sets of 2D images since additional calibration information is taken into account.
- A 3D representation is more robust to the object's positions relative to the cameras as it replaces a possibly complex matching between learned views and the actual observations by a 3D alignment (see next section).
- A 3D representation allows different camera configurations.
- Finally, as we will se in the remainder of this chapter, a 3D representation allows for view-invariant matching of actions, without requiring further input cues. This is hence in contrary to the previously discussed 2D invariant approaches (Section 2.3.2), which additionally required given point correspondences between pairs of observations.

Motion templates extend easily to 3D by considering the occupancy function in 3D $D(x, y, z, t)$, where $D = 1$ if (x, y, z) is occupied at time t and $D = 0$ otherwise, and by considering voxels instead of pixels:

$$v_\tau(x, y, z, t) = \begin{cases} \tau & \text{if } D(x, y, z, t) = 1 \\ \max(0, h_\tau(x, y, z, t-1) - 1) & \text{otherwise.} \end{cases} \tag{3.2}$$

In the rest of this chapter, we will assume templates to be normalized and segmented with respect to the duration of an action:

$$v(x, y, z) = v_{\tau = t_{\max} - t_{\min}}(x, y, z, t_{\max})/(t_{\max} - t_{\min}), \tag{3.3}$$

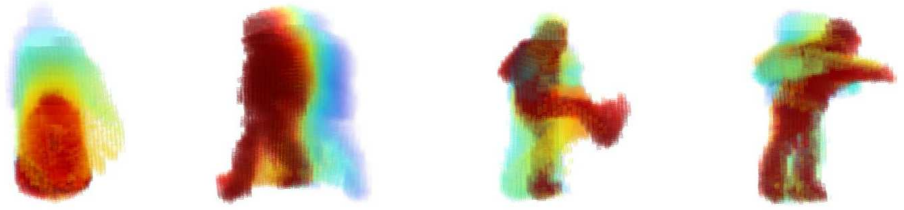


Figure 3.3.: Motion history volume examples: (Left to right) *sit down*, *walk*, *kick*, *punch*. Color values encode time of last occupancy.

where $t_{\min}$ and $t_{\max}$ are start and end time of an action. Hence, motions loose dependencies on absolute speed and result all in the same length. In Chapter 4 we will present an automatic method to detect action boundaries using a motion energy based segmentation.

The input occupancy function $D(x, y, z, t)$ is estimated using silhouettes and thus, corresponds to the visual hull [Laurentini, 1994]. Visual hulls present several advantages, they are easy to compute and they yield robust 3D representations. Note however that, as for 2D motion templates, different body proportions may still result in very different templates. Figure 3.3 shows examples for motion history volumes.

3.2. Motion Descriptors

Our objective is to compare body motions that are free in locations, orientations and sizes. This is not the case of motion templates, as defined in the previous section, since they encode space occupancy. The location and scale dependencies can be removed by centering, with respect to the center of mass, and scale normalizing, with respect to a unit variance, motion templates, as usual in shape matching. For the rotation, and following Bobick and Davis [1996b] who used the Hu Moments [Hu, 1962] as rotation invariant descriptors, we could consider their simple 3D extensions by Sadjadi and Hall [1980]. However, our experiments with these descriptors, based on first and second order moments, were unsuccessful in discriminating detailed actions. In addition, using higher order moments as in [Lo and Don, 1989] is not easy in practice. Moreover, several works tend to show that moments are inappropriate feature descriptors, especially in the presence of noise, e.g. [Shen and Ip, 1999]. In contrast, several works [Grace and Spann, 1991, Heesch and Rueger, 2002, Poppe and Poel, 2006] demonstrated better results using Fourier based features. Fourier based features are robust to noise and irregularities, and present the nice property to separate coarse global and fine local features in low and high frequency components. Moreover, they can be efficiently computed using fast Fourier-transforms (FFT). Our approach is therefore based on these features.

Invariance of the Fourier transform follows from the *Fourier shift theorem*: a function $f_0(x)$ and its translated counterpart $f_t(x) = f_0(x - x_0)$ only differ by a phase modulation after Fourier transformation:

$$F_t(k) = F_0(k)e^{-j2\pi k x_0}. \tag{3.4}$$

Hence, Fourier magnitudes $|F_t(k)|$ are shift invariant signal representations. The invariance property translates easily onto rotation by choosing coordinate systems that map rotation onto translation. Popular example is the Fourier-Mellin transform, e.g. [Chen et al., 1994], that uses log-polar coordinates for translation, scale, and rotation invariant image registration. Work in shape matching by Kazhdan et al. [2003] proposes magnitudes of Fourier spherical harmonics as rotation invariant shape descriptors.

In a similar way, we use Fourier-magnitudes and cylindrical coordinates, centered on bodies, to express motion templates in a way invariant to locations and rotations around the z-axis. The overall choice is motivated by the assumption that similar actions only differ by rigid transformations composed of scale, translation, and rotation around the z-axis. Of course, this does not account for all similar actions of any body, but it appears to be reasonable in most situations. Furthermore, by restricting the Fourier-space representation to the lower frequencies, we also implicitly allow for additional degrees of freedom in object appearances and action executions. The following section details our implementation.

3.2.1. Invariant Representation

We express the motion templates in a cylindrical coordinate-system:

$$v(\sqrt{x^2 + y^2}, \tan^{-1}\left(\frac{y}{x}\right), z) \rightarrow v(r, \theta, z).$$

Thus rotations around the z-axis results in cyclical translation shifts:

$$v(x\cos\theta_0 + y\sin\theta_0, -x\sin\theta_0 + y\cos\theta_0, z) \rightarrow v(r, \theta + \theta_0, z).$$

We center and scale-normalize the templates. In detail, if v is the volumetric cylindrical representation of a motion template, we assume all voxels that represent a time step, i.e. for which $v(r, \theta, z) > 0$, to be part of a point cloud. We compute the mean μ and variances σ_r and σ_z in z- and r-direction. The template is then shifted, so that $\mu = 0$, and scale normalized so that $\sigma_z = \sigma_r = 1$.

We choose to normalize in z and r direction, instead of a principal component based normalization, focusing on the main directions human differ on. This method may fail aligning e.g. a person spreading its hand with a person dropping its hand, but gives good results for people

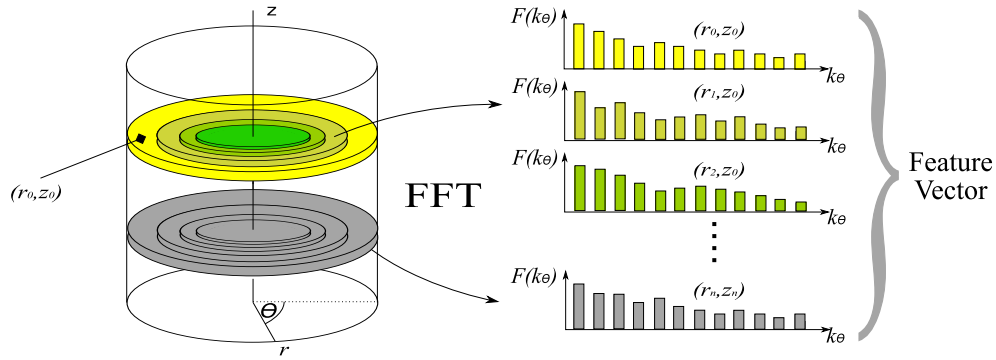


Figure 3.4.: 1D-Fourier transform in cylindrical coordinates. Fourier transforms over θ are computed for couples of values (r, z). Concatenation of the Fourier magnitudes for all r and z forms the final feature vector.

performing similar actions, which is more important.

The absolute values $|V(r, k_\theta, z)|$ of the 1D Fourier-transform

$$V(r, k_\theta, z) = \int_{-\pi}^{\pi} v(r, \theta, z) e^{-j2\pi k_\theta \theta} d\theta, \tag{3.5}$$

for each value of r and z, are invariant to rotation along θ.

See Figure 3.4 for an illustration of the 1D-Fourier transform. Note that various combinations of the Fourier transform could be used here. For the 1D Fourier-transform the spatial order along z and r remains unaffected. One could say, a maximum of information in these directions is preserved. Further, to gain the properties of the Fourier transform (*e.g.* robustness to noise, separation in fine and coarse features) for all dimensions, an additional 2D Fourier-transform can be applied to $f(r, k_\theta, z)$ for r and z:

$$\hat{V}(\omega_r, k_\theta, \omega_z) = \iint_{-\infty}^{\infty} |V(r, k_\theta, z)| e^{-j2\pi(\omega_r r + \omega_z z)} dr dz. \tag{3.6}$$

An important property of the 1D-Fourier magnitudes is its *trivial ambiguity* with respect to the reversal of the signal. Consequently, motions that are symmetric to the z-axis (e.g. move left arm - move right arm) result in the same motion descriptors. This can be considered either as a loss in information or as a useful feature halving the space of symmetric motions. However, our practical experience shows that most high level descriptions of human actions do not depend on this separation.

In cases where it is important to resolve left/right ambiguities a slightly different descriptor

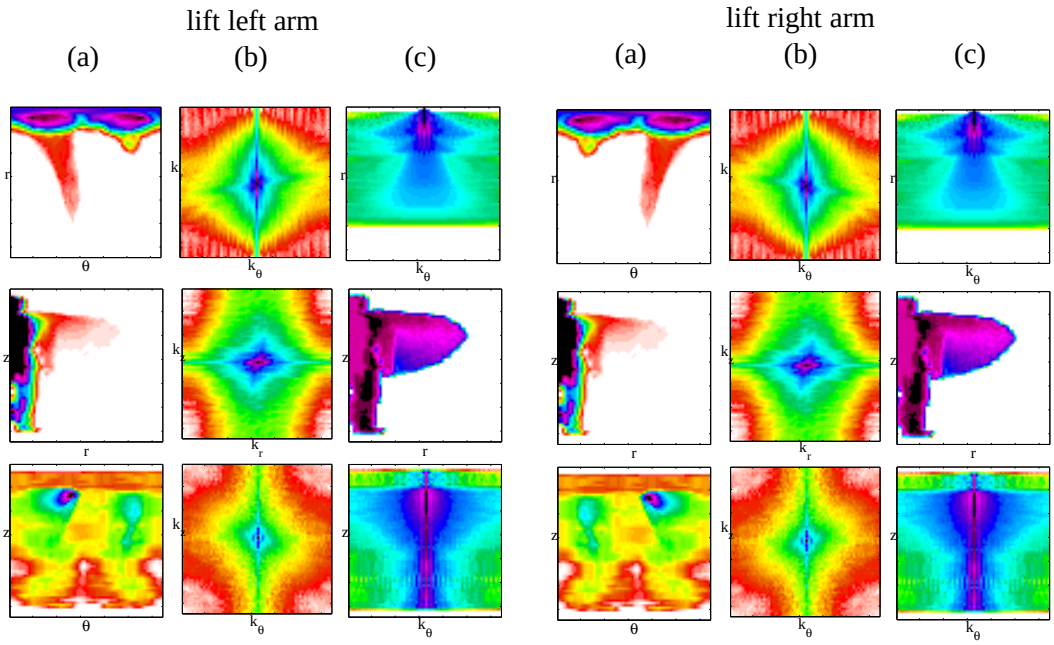


Figure 3.5.: Volume and spectra of sample motions: (a) cylindrical representation in $(\theta, r), (r, z), (\theta, z)$ averaged over the third dimension for visualization purposes; (b) corresponding 3D-Fourier Spectra; (c) 1D-Fourier spectra. Note that the 3D descriptor treats both motions differently (i.e. top and bottom row (b)), while the 1D descriptors treats them the same.

can be used. One such descriptor is the magnitude $|V(k_r, k_\theta, k_z)|$ of the 3D-Fourier transform

$$V(k_r, k_\theta, k_z) = \int_{-\infty}^{\infty}\int_{-\pi}^{\pi}\int_{-\infty}^{\infty} v(r,\theta,z)e^{-j2\pi(k_r r + k_\theta \theta + k_z z)} dr d\theta dz, \tag{3.7}$$

applied to the motion template v. This descriptor is only symmetric with respect to an inversion of all variables, i.e. humans standing upside-down, which does not happen very often in practice. While our previous work [Weinland et al., 2005] used that descriptor (3.7) with success, the results were anyway inferior to those obtained with (3.5) and an invariance to left right symmetry proved to be beneficial in many classification tasks. A visualization of both descriptors is shown in Figure 3.5.

3.3. Classification Using Motion Descriptors

We have tested the presented descriptors and evaluated how discriminant they are with different actions, different bodies or different orientations. Our initial results [Weinland et al., 2005]

using a small dataset of only two persons already indicated the high potential of the descriptor. Here we presents results on an extended dataset, the so called *IXMAS* dataset. The dataset is introduced in the next section, followed by classification results using dimensional reduction combined with Mahalanobis distance and linear discriminant analysis (LDA).

3.3.1. The IXMAS Dataset

Early in our thesis, we created the IXMAS (INRIA Xmas Motion Acquisition Sequences) data set for training, testing and evaluation of our algorithms. Since there was no publicly available data set of that kind at the time, we decided to make it available to other researchers as well[1]. The data set contains 11 actions, see Figure 3.6 for instance, each performed 3 times by 10 actors (5 males / 5 females). To demonstrate the view-invariance, the actors freely change their orientation for each acquisition and no further indications on how to perform the actions beside the labels were given, as illustrated in Figure 3.7.

The acquisition was achieved using 5 standard Firewire cameras. Figure 3.8 shows example views from the camera setup used during the acquisition. From the video we extract silhouettes using a standard background subtraction technique modeling each pixel as a Gaussian in RGB space. Then visual hulls are computed as discrete grids using a voxel carving method, where we carve each voxel that does not project into all of the silhouettes images. This method was, however, mostly chosen because of its simplicity, and there are no special requirements for the visual hull computation used.

For experiments in this this chapter, we compute MHVs in cylindrical coordinates from the visual hulls, as previously described. We use a discrete cylindrical coordinate representation with resolution $64 \times 64 \times 64$, if not otherwise mentioned. Temporal segmentation was performed manually, such that each action is represented through a single motion template. Note, that a fully automatic segmentation method is presented in Chapter 4.

3.3.2. Classification Using Mahalanobis Distance and PCA

In this section, we describe our experiments for classifying the actions in the IXMAS data set using MHVs. This includes dimension reduction (MHV initially has $64 \times 64 \times 64 = 262144$ dimensions) and the choice of discriminant functions.

In initial experiments on a small dataset and with different distance measures (i.e. Euclidean distance, simplified Mahalanobis distance, and Mahalanobis distance + PCA, see also [Weinland et al., 2005]), the combination of a principal component analysis (PCA) dimensional reduction plus Mahalanobis distance based normalization showed best results. Because of the small number of training samples that we had at this time, we only used one pooled covariance

[1] The data is available on the Perception website `http://perception.inrialpes.fr` in the "Data" section.

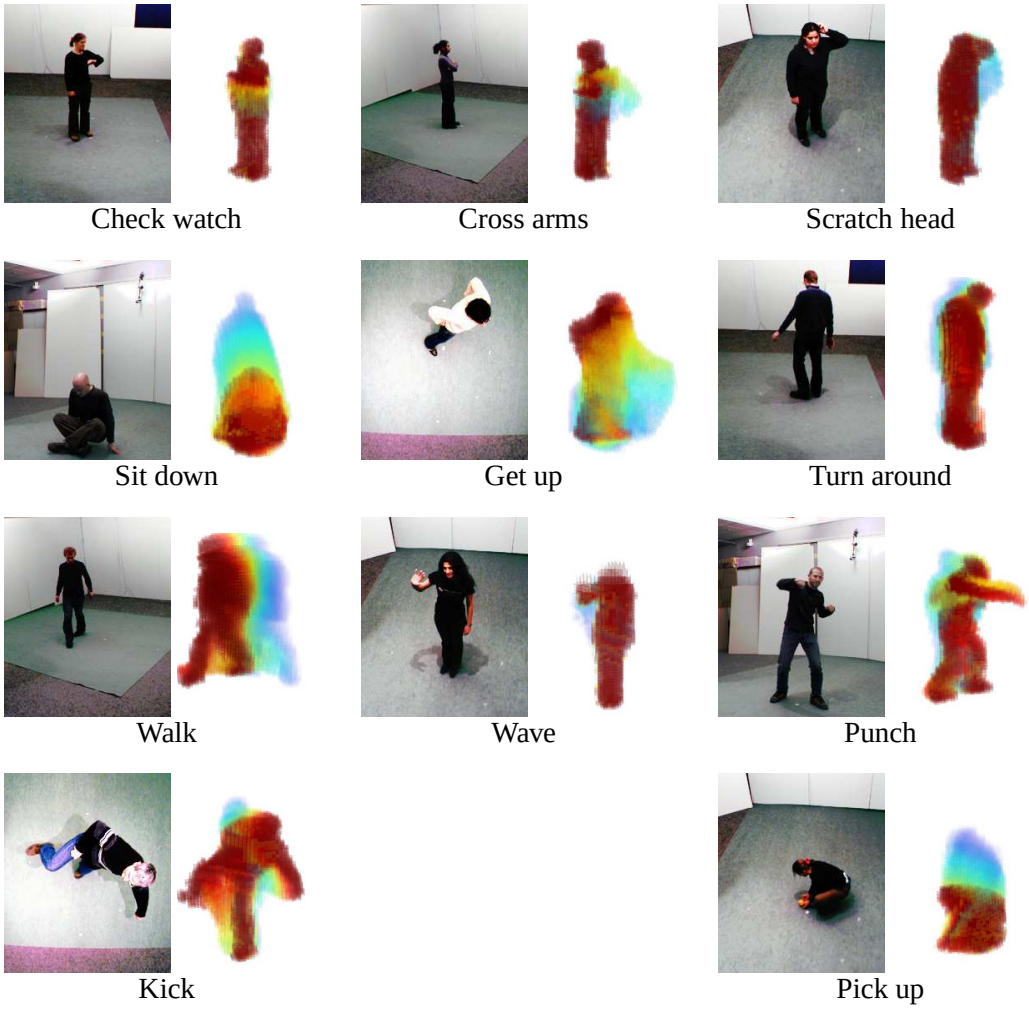


Figure 3.6.: 11 actions, performed by 10 actors.

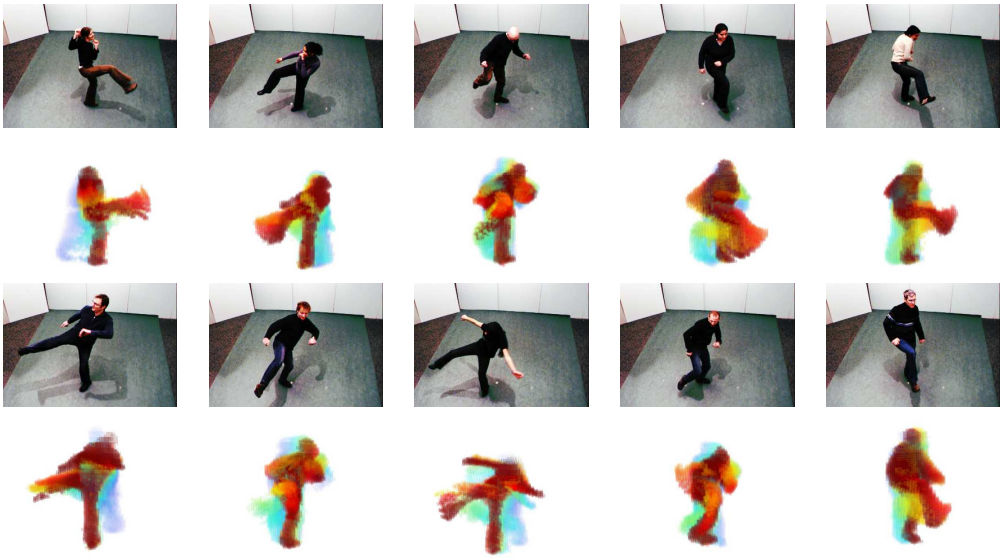


Figure 3.7.: Sample action “kick” performed by 10 actors.

Figure 3.8.: Example views of 5 cameras used during acquisition.

matrix computed from all samples from all classes. Interestingly, we found that the method extends well to larger datasets and even competes with linear discriminant analysis (LDA), as will be shown in the next section.

In PCA, data points are projected onto a subspace that is chosen to yield the reconstruction with minimum squared error. It is well known [Webb, 2002] that this subspace is spanned by the largest eigenvectors of the data's covariance Σ, and corresponds to the directions of maximum variance within the data. Further, by normalization with respect to the variance, an equally weighting of all components is achieved, similar to the classical use of Mahalanobis distances in classification, but here computed for one pooled covariance matrix.

Every action class in the data-set is represented by the mean value of the descriptors over the available population in the action training set. Any new action is then classified according to a Mahalanobis distance associated to a PCA based dimensional reduction of the data vectors. One pooled covariance matrix Σ based on the training samples of all classes $\mathbf{x}_i \in \mathbb{R}^d$, $i = 1, \ldots, n$ was computed:

$$\Sigma = \frac{1}{n} \sum_{i}^{n} (\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^\top, \tag{3.8}$$

where $\mathbf{m}$ represents the mean value over all training samples.

The Mahalanobis distance between feature vector $\mathbf{x}$ and a class mean $\mathbf{m}_i$ representing one action is:

$$d(\mathbf{m}_i, \mathbf{x}) = (\mathbf{x} - \mathbf{m}_i)^\top V \Lambda^{-1} V^\top (\mathbf{x} - \mathbf{m}_i),$$

with Λ containing the k largest eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_k$, $k \leq n - 1$, and V the corresponding eigenvectors of Σ. Thus feature vectors are reduced to k principal components.

Following this principle, and reducing the initial descriptor (equation (3.5)) to $k = 296$ components (the maximum number of possible components, given that we have 297 training sequences per experiment) an average classification rate of 93.33% was obtained with leave-one-out cross validation, where we successively used 9 of the actors to learn the motions and the 10th for testing. Note that in the original input space, as well as for a simple PCA reduction without covariance normalization the average rate is only 73.03%. Detailed results are given in Table 3.1.

3.3.3. Classification Using Linear Discriminant Analysis

For further data reduction, class specific knowledge becomes important in learning low dimensional representations. Instead of relying on the eigen-decomposition of one pooled covariance matrix, we use here a combination of PCA and Fisher linear discriminant analysis (LDA), see e.g. Swets and Weng [Swets and Weng, 1996], for automatic feature selection from high dimensional data.

Action	PCA (%)	Mahalanobis (%)	LDA(%)
Check watch	46.66	86.66	83.33
Cross arms	83.33	100.00	100.00
Scratch head	46.66	93.33	93.33
Sit down	93.33	93.33	93.33
Get up	83.33	93.33	90.00
Turn around	93.33	96.66	96.66
Walk	100.00	100.00	100.00
Wave hand	53.33	80.00	90.00
Punch	53.33	96.66	93.33
Kick	83.33	96.66	93.33
Pick up	66.66	90.00	83.33
Average rate	73.03	93.33	92.42

Table 3.1.: IXMAS data classification results. Results on PCA, PCA + Mahalanobis distance based normalization using one pooled covariance, and LDA are presented.

First PCA is applied, $Y = V^\top X$, $V = [\mathbf{v}_1, \ldots, \mathbf{v}_m]$, to derive a $m \leq n - c$ dimensional representation of the data points x_i, $i = 1, \ldots, n$. The class-number c dependent limit is necessary to guaranty non-singularity of matrices in discriminant analysis.

Fisher discriminant analysis defines as within-scatter matrix:

$$S_w = \sum_i^c \sum_j^{n_i} (\mathbf{y}_j - \mathbf{m}_i)(\mathbf{y}_j - \mathbf{m}_i)^\top, \tag{3.9}$$

and between-scatter matrix:

$$S_b = \sum_i^c (\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^\top, \tag{3.10}$$

and aims at maximizing the between-scatter while minimizing the within-scatter, i.e. we search a projection W that maximize $\frac{\det(S_b)}{\det(S_W)}$. It has been proven that W equal to the largest eigenvectors of $S_w^{-1} S_b$ maximizes this ratio. Consequently a second projection $Z = W^\top Y$, $W = [w_1, \ldots, w_k]$, $k \leq c - 1$ is applied to derive our final feature representation Z.

During classification each class is represented by its mean vector $\mathbf{m}_i$. Any new action $\mathbf{z}$ is then classified by summing Euclidean distances over the discriminant features and with respect to the closest action class:

$$d(\mathbf{m}_i, \mathbf{z}) = ||\mathbf{m}_i - \mathbf{z}||^2. \tag{3.11}$$

In the experiments the magnitudes of the Fourier representation (equation (3.5)) are projected

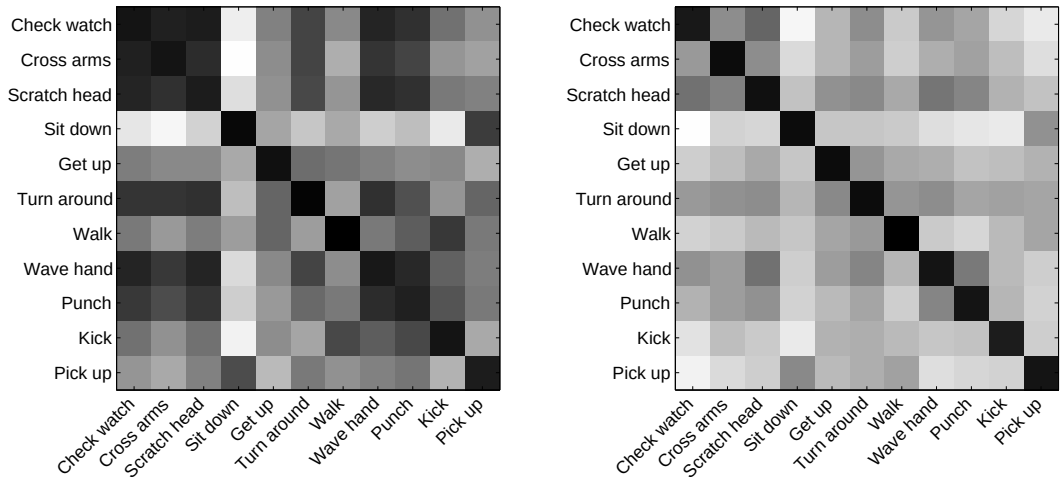


Figure 3.9.: Average class distance: (Left) before discriminant analysis. (Right) after discriminant analysis.

onto $k = 10$ discriminant features. Successively we use 9 of the actors to learn the motions, the 10th is used for testing. The average rate of correct classifications is then 92.42%. Class specific results are shown in Table 3.1 and Figure 3.9.

We note that we obtain much better results with the Mahalanobis distance, using the 296 largest components of the PCA decomposition, as compared to using the PCA components alone. LDA allows us to further reduce the number of features to 10, but otherwise does not further improve the overall classification results.

3.3.4. Motion History vs. Motion Energy and Key Frames

With the same dataset as before, we compare our MHV based descriptors with a combination of key poses and energy volumes. While Davis and Bobick suggested in the original paper the use of history and binary images, our experiments with motion volumes showed no improvement in using a combination of MHVs and the binary MEVs. We repeated the experiment described in section 3.3.3, for MEVs. Using the binary information the recognition rate becomes 80.00% only. See Table 3.2 for detailed results. As can be expected: reverse actions, e.g. “sit down” - “get up”, present lower scores with MEVs than with MHVs. The MHVs show also better performance in discriminating actions on more detailed scales, e.g. “scratch head” - “wave”.

Also, to show that integration over time plays a fundamental role of information, we compare our descriptor with descriptors based on a single selected *key frame*. The idea of key frames is to represent a motion by one specific frame, see e.g. Carlson and Sullivan [Carlsson and Sullivan, 2001]. As invariant representation, we use the magnitudes of equation (3.5). The

Action	MEV (%)	Key frame (%)	MHV (%)
Check watch	86.66	73.33	86.66
Cross arms	80.00	93.33	100.00
Scratch head	73.33	86.66	93.33
Sit down	70.00	93.33	93.33
Get up	46.66	53.33	93.33
Turn around	90.00	60.00	96.66
Walk	100.00	80.00	100.00
Wave hand	80.00	76.66	80.00
Punch	93.33	80.00	96.66
Kick	90.00	90.00	96.66
Pick up	70.00	96.66	90.00
Average rate	80.00	80.30	93.33

Table 3.2.: IXMAS data classification results. Results using the proposed MHVs are presented. For comparison we also include results using binary MEVs and key frame descriptors.

Resolution	$d = 64$ (%)	$d = 32$ (%)	$d = 16$ (%)	$d = 12$ (%)	$d = 10$ (%)
Average rate	93.33	93.33	88.18	81.52	63.64

Table 3.3.: Recognition rate for different grid sizes ($d \times d \times d$) and Mahalanobis distance based classification.

average recognition rate becomes 80.30%.

Note that for the purpose of this comparison we simply choose the last frame of each MHV computation as corresponding *key frame*. An improved method for key-pose selection and matching will be presented in Chapter 7.

3.3.5. Using Smaller Grid Resolution

In the previous tests we used a voxel grid of size $64 \times 64 \times 64$, which was an empirical choice leading to good results in all our experiments. Although feature vector sizes were further dimensionally reduced using PCA or LDA, the grid resolution nevertheless affects the computational performance of the initial steps of our method, *i.e.* 3D reconstruction, MHV computation, and FFT. Consequently using lower dimensional grids can lead to an improved performance of the overall framework. In this section we experimented with different smaller voxel grid resolutions. Results for the Mahalanobis based classification are shown in Table 3.3.

Interestingly, the recognition rates remain high, even for the $16 \times 16 \times 16$ sized grid. For size $10 \times 10 \times 10$ the rate drops to 63.64%. Figure 3.10 shows a confusion matrix for size

	check watch	cross arms	scratch head	sit down	get up	turn around	walk	wave hand	punch	kick	pick up
check watch	70	0	13	0	3	3	0	7	3	0	0
cross arms	3	73	0	0	0	3	0	0	20	0	0
scratch head	7	0	70	0	3	0	0	10	7	3	0
sit down	0	0	3	87	3	3	0	0	0	0	3
get up	0	3	0	0	87	3	3	0	3	0	0
turn around	0	0	0	0	7	87	0	3	3	0	0
walk	0	0	0	0	0	0	100	0	0	0	0
wave hand	0	0	13	0	0	0	0	73	13	0	0
punch	3	10	3	0	0	0	0	0	80	3	0
kick	3	0	3	0	0	0	0	3	3	87	0
pick up	3	0	0	3	0	7	0	0	3	0	83

Figure 3.10.: Confusion matrix (in %) for recognition using grid size $12 \times 12 \times 12$.

$12 \times 12 \times 12$. We observe, that recognition rate for actions involving larger parts of the body, *e.g*. *walk*, *sit down*, and *kick*, remain acceptable, while actions that involve only smaller body parts, *e.g*. the arms, have a larger decrease in recognition rate.

3.3.6. Invariance vs. Alignment

In this experiment we compare our invariant representation against results that can be achieved with a view-dependent framework, *i.e*. a scenario where all actors have the same body orientation. We therefore use a semi-automatic method to rotationally align all MHVs: we first compute a rough alignment of all volumes using a correlation based method; thereafter misalignment are manually corrected if necessary. Further, we inverted volumes in case of a left right ambiguities, such that in the final set every action is only performed with a single body side, *i.e*. left or right arm/leg. for matching, the resulting set is then only normalized with respect to scale and position, but no rotation invariant descriptor is computed. Classification is performed as previously, using PCA + Mahalanobis distance. This representations lead to a recognition rate of 94.85%. Figure 3.11 shows the confusion matrix in that case, as well as the confusion matrix using our invariant descriptor (both with grid size $32 \times 32 \times 32$[2]). We observe that using our invariant representation in 3D leads to only marginal differences in recognition rates, compared to results on view-aligned data.

The experiment shows that we are indeed preserving all the useful discriminative information

[2]Actually, for the invariant representation we only need to store $32 \times 16 \times 32$, because of the symmetry of the Fourier magnitudes

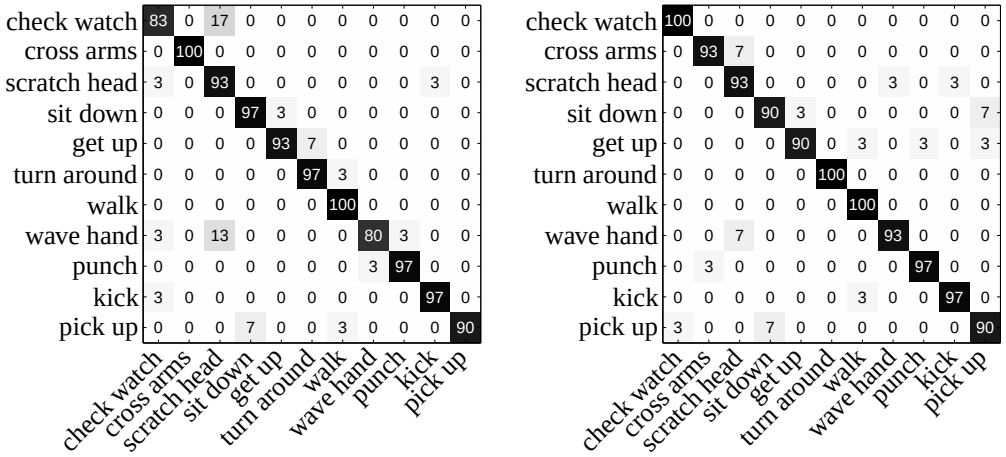


Figure 3.11.: Confusion matrix (in %) for (left) using invariant representation (right) using manually aligned volumes.

with our view-independent MHV. Using invariant motion descriptors is of course advantageous in practice, because we do not need to align training examples for learning a class model, neither do we depend on the correct alignment of test examples with all class prototypes for recognition. On the other hand, we are losing the benefits of view angle estimation, which may be a useful information to keep for recognizing sequences of primitive actions over time. We investigate such an approach in Chapter 6.

3.4. Conclusion

Using a data set of 11 actions, we have been able to extract 3D motion descriptors that appear to support meaningful categorization of simple action classes performed by different actors, irrespective of viewpoint, gender and body sizes. Best results are obtained by discarding the phase in Fourier space and performing dimensionality reduction with a combination of PCA and LDA. Further, LDA allows a drastic dimension reduction (10 components). This suggests that our motion descriptor may be a useful representation for view invariant recognition of an even larger class of primitive actions.

A limitation of the method is that the representation depends on the full knowledge of all action classes. Adding a new action class requires us to recompute everything. Ideally, we would like to derive a more compositional approach. Also, in this chapter we only addressed recognition from single temporally-segmented instances of actions since our focus was to validate the action descriptor. For practical applications, we still need a method for performing the segmentation in the first place. This is the topic of our next chapter.

CHAPTER 4

Action Segmentation using Motion History Volumes.

In this chapter we use the previously introduced *motion history volumes* (chapter 3) to automatically segment action sequences into primitive actions that can be represented by a single MHV. We then cluster the resulting MHVs into a hierarchy of action classes, which allow us to recognize multiple occurrences of repeating actions. We are able to perform those two steps automatically, mainly because MHVs work in a volume space which considerably reduces the ambiguities traditionally associated with changes in viewpoints and occlusions even in multiple views.

Our framework is a first step to automatically generate high-level descriptions of video sequences in terms of the actions that can be recognized or inferred from the given visual input. Actions generally fall under two distinct categories - composite actions which can be broken down into distinct temporal parts or segments, and primitive actions, which cannot be broken down further. In order to build a general action recognizer, we need the ability to break down a given sequence into primitive action segments, to label those segments into primitive actions using a vocabulary of learned action models, and to assemble the labeled segments into composite actions, using concept hierarchies [Kojima et al., 2002] or grammars [Ogale et al., 2005] for instance.

As a concrete example, we asked two members of our lab to perform a sequence of simple actions, each repeated several times with different poses and styles, in front of 6 calibrated cameras. The resulting data set consists of unsegmented and unlabeled synchronized video sequences such as the one depicted in Figure 4.1. Using the new motion descriptor, we were able to segment (Section 4.1) and cluster (Section 4.2) such sequences into primitive actions,

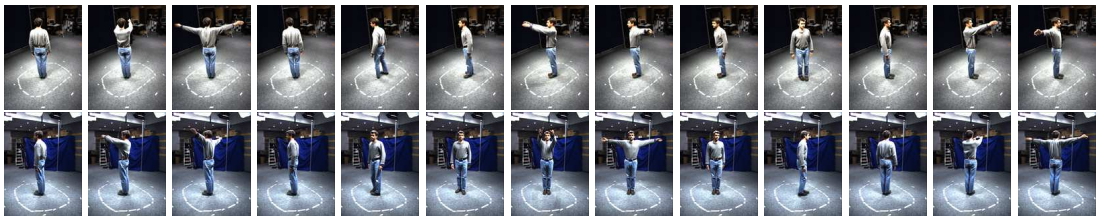


Figure 4.1.: Example action sequence: Raise arms - rotate arms - turn left - raise arms - rotate arms - turn left - raise arms - rotate arms, seen from two different viewpoints. Such sequences are difficult to segment and label consistently from monocular cues, but are easily segmented and labeled using our view-independent motion descriptors.

which we used as training examples for learning statistical classifiers. Such a semi-supervised scheme is important in practical terms because it facilitates the creation of large training sets for action recognition in the large.

Our method generates action taxonomies based on purely visual cues since we create higher-level action classes by abstracting two or more recorded actions which *look the same* from all viewpoints (as measured by the differences in a metric space of motion descriptors extracted from their MHVs). We believe this is an important step towards building complete, semantic taxonomies of actions and plans.

Segmentation and labeling of action sequences from *multiple views* is a relatively little-studied area. Previous work assumes either that the cameras are uncalibrated (so that reconstruction is not possible) or that a full human body model can be recovered (so that reconstruction includes body part recognition and tracking). To the best of our knowledge, no previous work has attempted to perform segmentation and clustering from *volumetric* reconstructions. In this chapter, we propose such a method, which extends monocular methods most naturally by means of our view-invariant MHV representation. Compared with previous work, our method has the advantage that we perform all three steps of segmenting, clustering and classifying action sequences in 3D with a representation which is fully view-invariant, and is much simpler to recover than a full human body model.

The chapter is organized as follows. We describe our segmentation algorithm in Section 4.1 and our clustering algorithm in Section 4.2. Finally, we show experiments with automatic segmentation and recognition on continuous streams of actions in Section 4.3, before we discuss issues and conclude in Section 4.4.

4.1. Temporal Segmentation

As mentioned earlier, temporal segmentation consists in splitting a sequence of motions into elementary segments. It is a necessary preliminary step to higher level processing of motion sequences including classification and clustering. In supervised approaches, segments are usually manually labeled in an initial set of motion sequences, and further operations are achieved by correlating unknown motion sequences with these learned segments on a frame by frame basis, using possibly various temporal scales (see our discussion on *sliding window segmentation*, Section 2.4.1). In this work, we do not assume such *a priori* knowledge and propose instead a simple but efficient *boundary detection* based approach to automatically segment 3D motion sequences.

Any temporal segmentation relies on the definition of elementary motion segments. For boundary detection based methods, segments are implicitly defined through characteristic motion features representing start and end points of an action. There are two main approaches to such segmentation: Energy minima can be used to detect reversal of motion direction, following an early proposal by Marr and Vaina [Marr and Vaina, 1982]. Or discontinuities can be used to detect changes in the temporal pattern of motion [Rubin and Richards, 1985, Rui and Anandan, 2000]. From experiments we found energy minima more stable, i.e. similar action sequences are segmented more consistently.

The function over time that we segment is then a global motion energy function. This function is an approximation of the global body velocity estimated using the motion history volumes. It is based on the observation that rest states correspond to instants where few motions only occur, and thus result in few voxels encoding motion in the MHV, when small temporal windows are considered. Therefore, segment detection simply consists in finding minima of the sum of voxel values in the MHV, assuming a small value for the window size τ during the MHV computation (equation 3.2). Figure 4.2 shows several examples of sequences segmented this way. As can be seen in the figure, detection of energy minima is fairly unambiguous in this examples.

In our implementation we use a derivative of Gaussian filter and zero crossing to detect the minima. Parameter τ in equation (3.2) was set to constant 10 frames during all experiments. In practice, the minima detection appears to be very successful in segmenting motions, even for coupled motions, like moving torso and arms in parallel, local minima occur. Of course, this measure is still sensitive to small variations of velocity that can result in local minima. However, by allowing a possible over-segmentation the method will detect most of the motion segment boundaries.

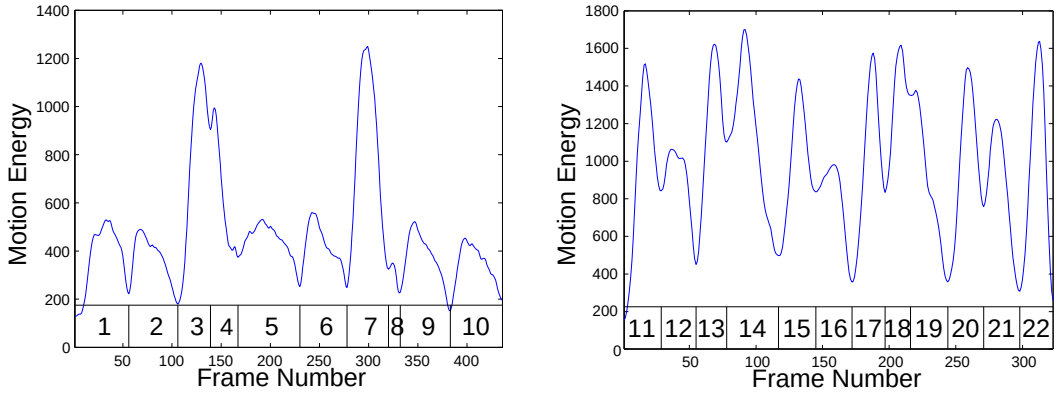


Figure 4.2.: Motion energy for action: Lift arms - rotate arms - lower arms and turn in new position. Executed three times by (left) female actor, (right) male actor. Local energy minima serve as segmentation criteria of sequences. Note that the female actor simultaneously lowers the arms and turns around, which results in a single motion segment with a very high energy. On the other hand, the male actor first lowers the arms and then turns around, which results in two separate motion segments. Motion volumes for each segment are shown in Figure 4.3.

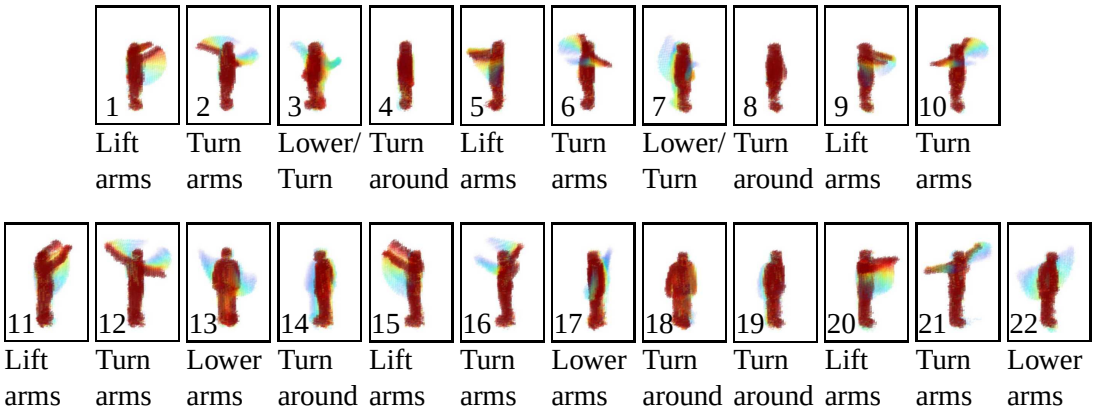


Figure 4.3.: History volumes computed at segments of varying duration, and their clusters, using segmentation from Figure 4.2. (Top) female actor repeating three times: Lift arms ahead - rotate arms - lower arms and turn in new position. (Bottom) the same done by a male actor, from original sequence shown in Figure 4.1. The clusters are labeled manually for presentation purposes.

4.2. Action Taxonomies

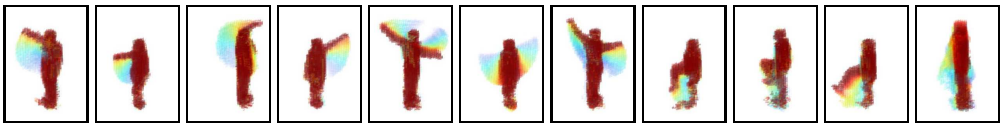
Given a segmented action sequence, we would like to recognize multiple occurrences of the same primitive actions and to label the sequence accordingly. This capability will be important in the next section when we attempt to train classifiers for all primitive actions in a semi-supervised fashion.

We build an action taxonomy from a segmented sequence by hierarchically clustering the segments into classes. Initially, each segment is a single occurrence of its own action class, and is represented as a single point in the space of view-invariant motion descriptors of Section 3.2, which is a high-dimensional Euclidean space. We then apply a standard hierarchical clustering method to the segments. This creates a binary tree of action classes, where each class is now represented by a point cloud in the space of motion descriptors (see Figure 4.5).

In this section we report experiments on two different datasets of increasing complexity. In each we segment the sequences as explained in Section 4.1 and compute a single MHV per segment. This is illustrated in Figures 4.3 and 4.7. The experiments were conducted on MHVs obtained from 6 silhouettes extracted using a standard background subtraction method. The resulting motion templates were mapped into a discrete cylindrical coordinate representation of size $64 \times 64 \times 64$. Clustering was achieved using an agglomerative scheme, where the distance between objects is the Euclidean distance, and clusters were linked according to their furthest neighbor. The first dataset shows how actions performed by different persons, with different bodies, are handled by our system. The second dataset is a more realistic set of natural actions in arbitrary orders. Its interpretation is less straightforward, but it gives strong insights on the potential of our motion descriptors to yield consistent high-level interpretations.

4.2.1. Clustering on Primitive Actions

Here a dataset of 22 motion sequences performed by both a male and a female actor were considered. Segmented key actions are shown in Figure 4.4. The actors perform successively each action three times while changing their orientations in between. The automatic motion segmentation returns 203 motion volumes (100 for the woman, 103 for the man). We start by computing a dendrogram of all male segments, using Euclidean distances and furthest neighbor assignments. A good trade-off between motion variation within single clusters and multiple clusters having same labels is then to cut the hierarchy into 21 clusters. All segments inside these clusters are labeled according to the most obvious interpretation. From these labels, the 21 clusters are then labeled with respect to the most current actions which occurs in each cluster. Figure 4.5 shows the labeled dendrogram. Within these clusters, 7 (6.8%) actions were obviously assigned a wrong cluster, 4 actions give birth to single clusters, and one cluster is



(1) (2) (3) (4) (5) (6) (7) (8) (9) (10) (11)

Figure 4.4.: Perspective views of the motion history volumes computed for each action category. (1) lift right arm ahead. (2) lift right arm sideways. (3) lift left arm sideways ahead. (4) lift left arm sideways. (5) rotate both arms lifted. (6) lower both arms sideways. (7) lift both arms sideways. (8) lift right leg bend knee. (9) lift left leg bend knee. (10) lift right leg firm. (11) jump.

ambiguous (lower or lift arm sideways).

We next compute a hierarchy from the male and female data. The procedure is the same as in the previous experiment. Because of higher variations in the dataset the clusters result in a coarser action grouping. A good trade-off between motion variation within single clusters and multiple clusters having same labels is this time to cut the hierarchy into 9 clusters, as shown in Figure 4.6. With respect to this labeling only two actions are wrongly assigned.

4.2.2. Clustering on Composite Actions

In another clustering experiment we used a different dataset of actions with a much more complex semantics. Those sequences are pantomimes of various daily life actions such as catching a ball, picking up, stretching, laughing, etc. The segmentation and clustering methods were applied to each of these sequences. Figures 4.7 and 4.8 show the segmented motion templates and the hierarchy obtained for one such sequence. Again groups of higher level actions in Figure 4.8 have a simple interpretation such as lift or lower arms. Note also the group *rest in position* where segments without motion, typically between actions, have been consistently clustered.

4.3. Continuous Action Recognition

In this experiment we use the segmentation for recognition on continuous streams of motions. In particular we test the descriptor on unseen motion categories as they appear in realistic situations. For this purpose we work on the raw video sequences of the IXMAS dataset (see Section 3.3.1). In a first step the dataset is segmented into small motion primitives using the automatic segmentation. Then each segment is either recognized as one of the 11 learned classes or rejected. As described in Section 3.3.2, for classification we work in normalized PCA space spanned by the 11 sample motions and perform nearest-mean assignment. To decide for the “garbage”-class we use a global threshold on the distance to the closest class.

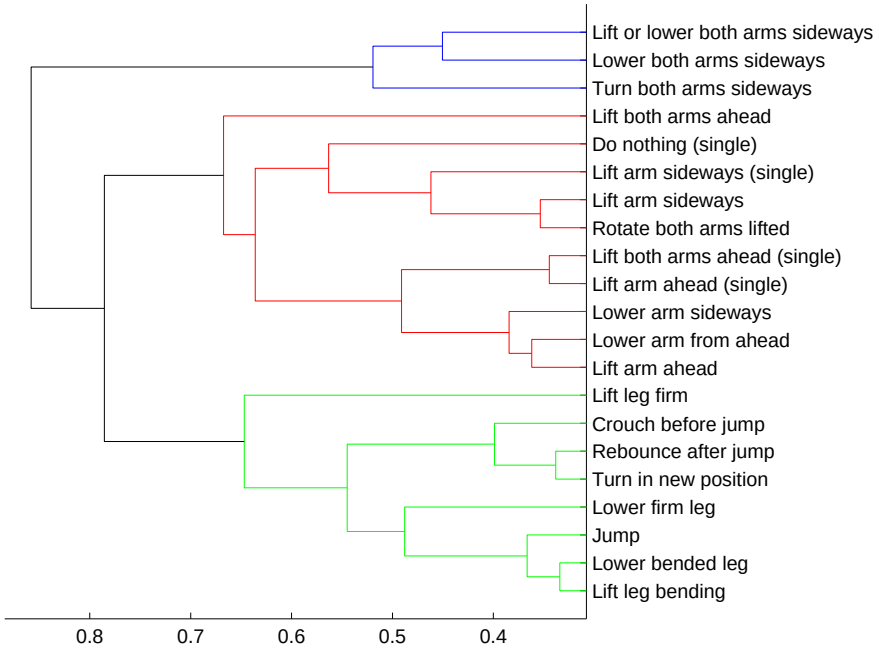


Figure 4.5.: Hierarchical clustering of 103 male actions. 21 top nodes labeled with respect to the most occurring action.

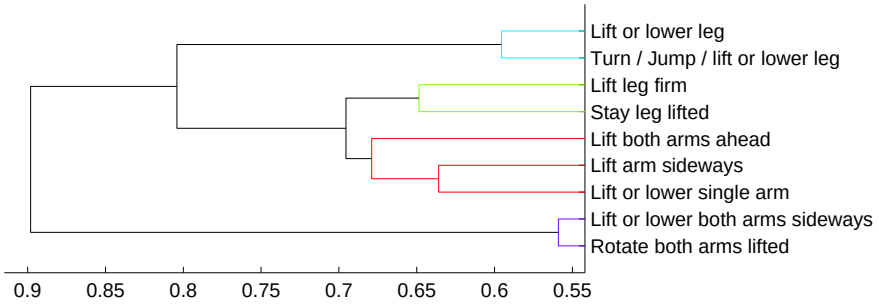


Figure 4.6.: Hierarchical clustering of 203 male and female actions. 9 top nodes labeled with respect to the most occurring action.

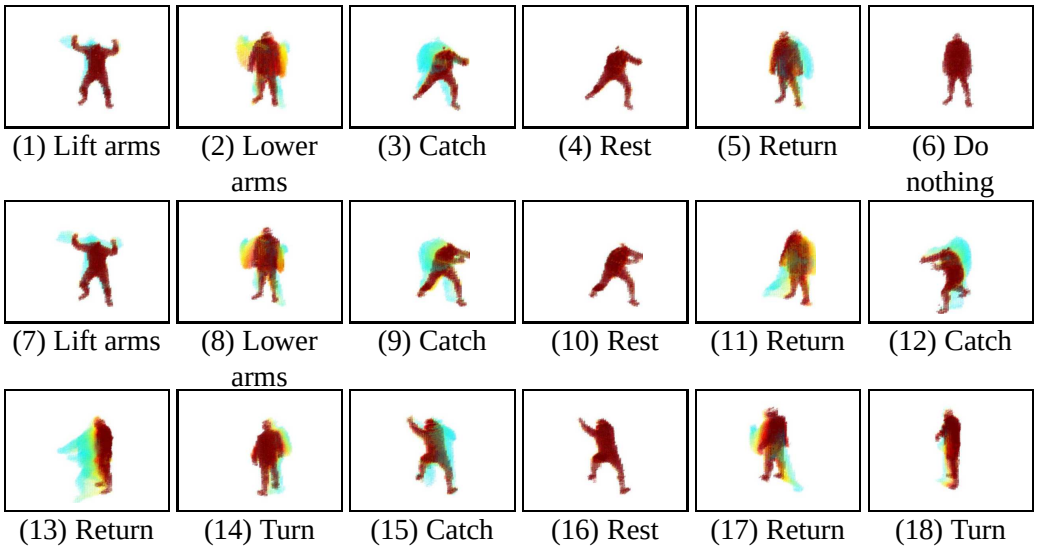


Figure 4.7.: History Volumes for pantomime sequence "catching ball".

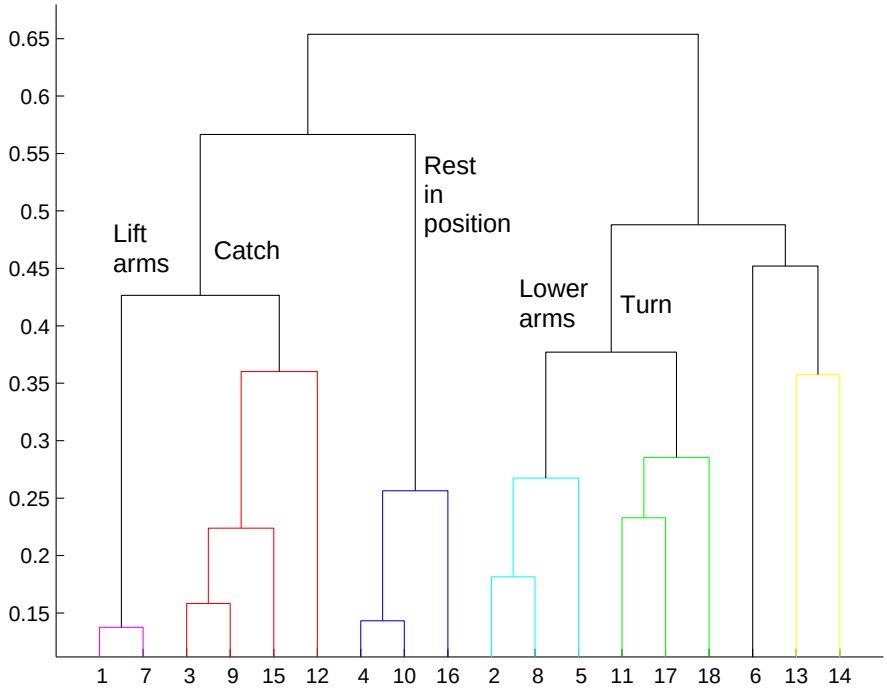


Figure 4.8.: Hierarchical clustering of "catching ball" sequence. Resulting segments are shown in Figure 4.7.

The automatic segmentation of the videos results in 1188 MHVs, corresponding to approximately 23 minutes of video. In this experiment we set parameter $\tau = 2$, which results in a slightly finer segmentation compared to the previous experiments. In manual ground truth la-

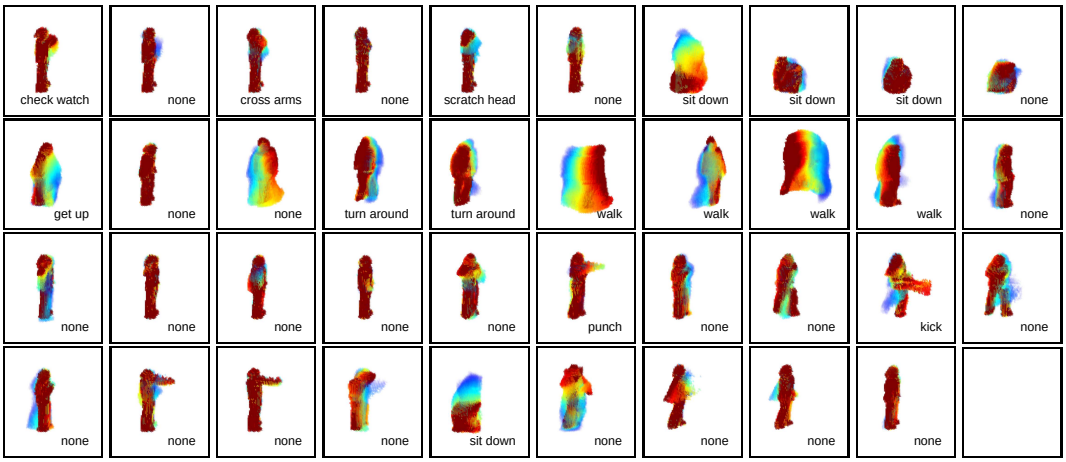


Figure 4.9.: Continues sequence segmented into MHVs and classification results. Action labels as recognized are displayed. Note that the actions *wave* is incorrect labeled as *none*, and *pick up* is labeled as *sit down*. Further the dataset contains two actions (*point* and *throw*), that we did not learn, and that are correctly recognized as *none*.

beling we discover 495 known motions and 693 "garbage"-motions. Note, that such a ground truth labeling is not always obvious. A good example is the "turn"-motion that was included in the experiments, but additional turn-like motions also appear as the actors where free to change position during the experiments. Moreover, it might be that an actor was accidentally checking his watch or scratching his head.

A sample sequence and recognition results are shown in Figure 4.9. Testing in a leave-one-out manner, using all possible combinations of 9 actors for training and the remaining 10th for testing, we show a multi-class ROC curve, Figure 4.10, plotting the average number of correctly classified samples, against the number of false positives. We found a maximal overall recognition rate (including correctly rejected motions) of 82.79%, for 14.08% false positives and 78.79% correctly classified motions. Figure 4.11 shows the average distance between the "garbage"-motions and the learned classes.

4.4. Discussion and Conclusion

In this chapter, we introduced MHV based methods for segmenting and clustering sequences of volumetric reconstructions of a human actor performing actions, without recognition or tracking of body parts. This has allowed us to learn classifiers for a small vocabulary of primitive actions, independently of style, gender and viewpoint. We have also applied our algorithms to discover meaningful hierarchies of action concepts in more complex composite sequences. Moreover, in experiments we demonstrate the ability of MHVs to work with large amounts of data and

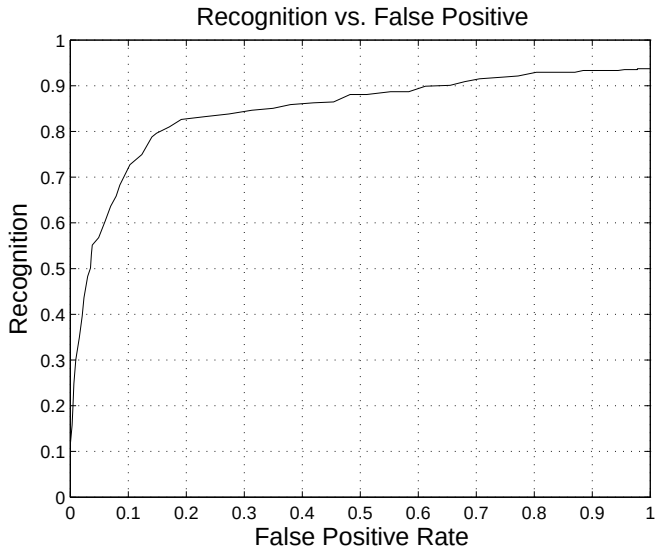


Figure 4.10.: Recognition rate on raw video sequences: Plots recognition rate into 11 classes against false positive rate.

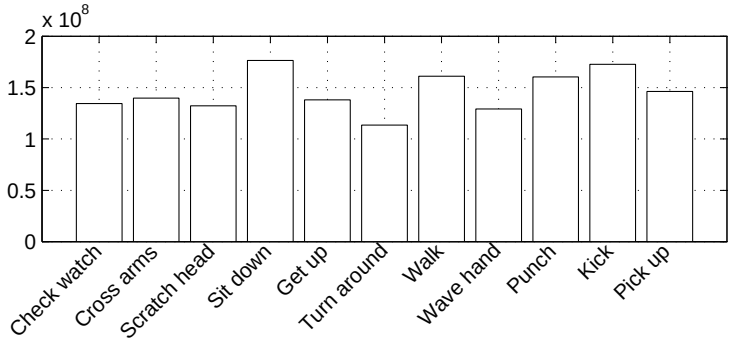


Figure 4.11.: Average distance between “garbage”-samples and training classes.

under realistic situations (23 minutes of video, 1188 motion descriptors). The segmentation proved to almost always detect the important parts of motions; MHVs showed good quality in discriminating learned and unseen motions.

Although we had surprisingly good results, a ”garbage” class of motions in more realistic settings may require more than a single threshold; multiple thresholds and explicit learning on samples of unknown motions becomes important. Another problem we found is, that many motions can not be modeled by a single template. Several of the IXMAS ”actions” are in fact not primitive actions, but already composite actions. This problem is made more acute as a result of possible over-segmentation. For example the turn around motion consists of several small steps, which may easily be confused with a single side step action. This means that we should not only be concerned with building taxonomies of actions, but also with building action ”partonomies” (how actions are built up from primitive actions). The HMM approach in Part II provides an extensive discussion of that problem.

One of the major conclusions we draw from our work on MHVs is, that an invariant 3D representation can easily resolve many of the difficulties that are inherent to view-dependent/single-view representations. In particular, ambiguities arising from changes in view and self occlusion are naturally handled by our representation. We would like to emphasize, that our representation does not need additional information in form of point correspondences or a joint body models.

An evident drawback of MHVs is the dependency on multiple views during recognition, which may not always be available in realistic scenarios. Multiple views during learning are not an evident drawback. In fact, they are even a very useful feature, as this chapter demonstrates. Therefore, we present in the next part of this thesis an action representation, which is specially designed for learning from multiple-views and recognition from single and arbitrary number of views. The representation nevertheless preserves the advantages of MHVs *i.e.* no point correspondences or joint model estimation, and rotational view-independence.

Part II

Action Recognition from Arbitrary Views: 3D Exemplar-based HMM

In this second part, we explore a different approach to view invariance. Whereas in the first part, we had extracted view-invariant features, here we model view-transform explicitly and search exhaustively over the unknown view parameters. To this aim we propose a new framework, where we model actions with an exemplar-based HMM. Compared to MHV, this approach allows us to tackle longer action sequences and to perform recognition without 3D reconstruction, even possibly from single views. Because MHV do not readily project to single views, we instead use an instantaneous representation of key poses in the next two chapters. A more thorough comparison of the two models is deferred to Section 6.6.3.

In Chapter 5 we start with a general introduction into HMMs and their extension to exemplar-based HMMs. In Chapter 6 we derive and evaluate our framework for view-independent action recognition using a 3D exemplar-based HMM.

CHAPTER 5

Markov Models for Action Recognition

In this section we briefly review *hidden Markov models* (HMM) and their extensions to exemplar-based HMMs. As such, this section serves as an introduction into the basic HMM concepts and terminologies, which are important for the understanding of the framework presented in the next chapter. For a detailed introduction into HMMs we refer the reader to the tutorial [Rabiner, 1990]. We will start by introducing *Markov chains*, the simplest form of a Markov model.

5.1. Markov Chain

A Markov chain is a *finite state automata*, see Figures 5.1(a) and 5.1(b), which consist of a single, discrete random state variable $q \in \{1, 2, \ldots, N\}$, and state transition probabilities $p(q_t|q_{t-1})$. These transition probabilities present the so called *Markov property*, *i.e.* the state of variable q_t at time t only depends on the directly temporal preceding state q_{t-1} at time $t-1$.[1] Hence given q_{t-1}, q_t is conditional independent of all preceding states $q_1, \ldots, q_{t-2}$

$$p(q_t|q_{t-1}, q_{t-2}, \ldots, q_1) = p(q_t|q_{t-1}). \tag{5.1}$$

Further, a Markov chain is said to be *time invariant* or *homogenous* if the transitions are constant over time

$$p(q_t = i|q_{t-1} = j) = p(i|j). \tag{5.2}$$

[1] Precisely: in a *first-order Markov model*, each state only depends on its direct predecessor. Higher-order Markov models with dependencies over several time steps exist as well, but are not discussed in this thesis.

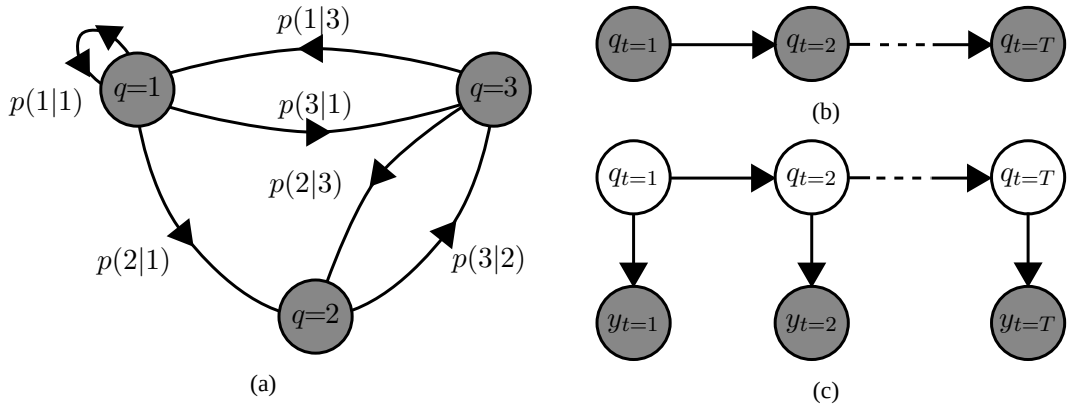


Figure 5.1.: Markov models and different visualization techniques: (a) Markov chain over state variable q with state transition probabilities $p(q_t|q_{t-1})$, visualized as *finite state machine* (transitions with $p(q_t|q_{t-1}) = 0$ are not shown). (b) Same Markov chain visualized as *graphical model*. (c) Hidden Markov model. State variable q is no longer directly observed and is therefore *hidden*. A different source of observations is added, represented through variable y (observed variable are displayed as shaded nodes, hidden/latent variables as white nodes).

A Markov chain is thus completely described through a parameter set $\lambda = \{A, \pi\}$, where A is conditional probability table with $a_{ij} = p(q_t = j|q_{t-1} = i)$, and π is vector of initial probabilities at time $t = 1$, with $\pi_i = p(q_1 = i)$.

To give an example: in action recognition, each state label $q \in \{1, 2, \ldots, N\}$ can represent one of N different body configurations. Let us assume that we can extract such labels directly from a video sequence as $Q = q_1, q_2, \ldots, q_T$. Respectively, we can compute the joint observation probabilities of this sequence with respect to a given action model λ, *i.e.* the probability that this posture sequence was generated by the action model, as

$$p(Q|\lambda) = p(q_1, q_2, \ldots, q_T|\lambda) = p(q_1|\lambda) \prod_{t=2}^{T} p(q_t|q_{t-1}, \lambda). \tag{5.3}$$

Note, that to keep the notation uncluttered, we will omit the dependency on λ in the following whenever not critical for the context.

5.2. Hidden Markov Models

Often, *e.g.* typically in action recognition, we will not observe meaningful state labels, as we assumed in the previous example. While we nevertheless can assume, that a Markov sequence over primitive posture states generated the observed signal, our actual observations are usually abstract features extracted from video sequences. To model such relations we can employ a

hidden Markov model (HMM), which explicitly separates between *latent* Markov states and observations. A HMM is illustrated in Figure 5.1(c), where an additional observation variable y (in the simplest case a discrete variable $y = 1, ..., M$) is introduced. We say the actual state sequence becomes *hidden* or *latent*, as we can no longer observe it directly (which is indicated by unshading the nodes in the figure). Hidden states and observations are linked by conditional distributions $p(y|q)$, *i.e.* the probability of observing feature y while being in hidden state q. The resulting HMM is then described by parameter set $\lambda = \{A, B, \pi\}$, with A and π being state transitions and initial probabilities, as described in the previous section, and B is probability table of conditional observation probabilities, *i.e.* $b_{ij} = p(y = j|q = i)$.

HMM is a good model for action recognition because a HMM can naturally represents the uncertainties between posture configuration and visual observation, in terms of probabilistic dependencies. For instance, a state can have high probability for all observations that represent variations of a single posture, as result of different body proportion, action style, view, or simply camera noise. However, it is important to emphasize, that in practice HMM states not necessarily represent semantically meaningful parts of actions. Often they are simply discovered as a result of learning a discrete model from experimental data.

5.2.1. Observation Probability

Given a state sequence $Q = q_1, q_2, \ldots, q_T$ we can compute the joint probability of observing $Y = y_1, y_2, \ldots, y_T$ and Q as

$$p(y_1, y_2, \ldots, y_T, q_1, q_2, \ldots, q_T) = p(q_1)p(y_1|q_1) \prod_{t=2}^{T} p(q_t|q_{t-1})p(y_t|q_t). \tag{5.4}$$

In practice, however, we often only have access to the observation sequence, while the responsible state sequence is not observable, *i.e.* is hidden. If we hence want to compute the probability of observing Y, given a model, and independent of the state sequence, we have to marginalize over all possible sate sequences

$$p(y_1, y_2, \ldots, y_T, \lambda) = \sum_{q_1, q_2, \ldots, q_T \in Q} p(y_1, y_2, \ldots, y_T, q_1, q_2, \ldots, q_T). \tag{5.5}$$

Such a brute force marginalization, which in fact has time complexity $O(2T \cdot N^T)$ [Rabiner, 1990], can become rapidly computationally unfeasible, even for small values of N and T. Luckily, there is an algorithm based on dynamic programming to compute such marginals more efficiently. This is the *forward-backward algorithm*, which has time complexity $O(N^2 \cdot T)$.

5.2.2. Forward-Backward Algorithm

The forward-backward algorithm is an efficient recursive method based on dynamic programming for learning and inference with HMM. It is described in terms of a *forward variable* and a *backward variable*, which are notated as $\alpha_t(i)$ and $\beta_t(i)$. Forward variable $\alpha_t(i) = p(y_1, y_2 \ldots, y_t, q_t = i)$ describes the probability of being in state i at time t while observing the partial sequence $y_1, y_2 \ldots, y_t$. The computation of $\alpha_t(i)$ follows an recursive update rule:

$$\alpha_t(i) = p(y_t|q_t = i) \sum_{j=1}^{N} \alpha_{t-1}(j) p(q_t = i|q_{t-1} = j), \quad 2 \leq t \leq T, \tag{5.6}$$

with initial condition

$$\alpha_1(i) = p(q_1 = i) p(y_1|q_t = i), \quad t = 1. \tag{5.7}$$

Using the forward variable, the observation probability (5.4) of a sequence Q can now be efficiently by marginalizing q out of α at time $t = T$

$$p(y_1, y_2, \ldots, y_T) = \sum_{i=1}^{N} \alpha_T(i). \tag{5.8}$$

Accordingly, the backward variable $\beta_t(i) = p(y_t, y_{t+1}, \ldots, y_T|q_t = i)$ describes the probability of observing the last $T - t$ observations $y_t, y_{t+1}, \ldots, y_T$ given that the state at time t is i. For more details we refer to [Rabiner, 1990]. Combinations of forward and backward variable are for instance used for parameter estimation of an HMM, and for computing the best state sequence trough an HMM using the Viterbi algorithm.

5.2.3. Classification using Maximum a Posteriori Estimate

Using Bayes theorem the posterior of a class $c \in \{1, \ldots, C\}$ given observation $Y = y_1, y_2, \ldots, y_T$ is

$$p(c|Y) = \frac{p(Y|c, \lambda_c) p(c)}{p(Y)}. \tag{5.9}$$

Because $p(Y)$ is independent of the class, it follows that

$$p(c|Y) \propto p(Y|c, \lambda_c) p(c), \tag{5.10}$$

and hence we can define a maximum a posteriori (MAP) classifier as

$$g(Y) = \arg\max_c p(Y|c, \lambda_c) p(c). \tag{5.11}$$

Base on this relation, a straightforward strategy to use HMMs in action recognition is to learn one separate model λ_c per action class. For an observation sequence the posterior under each model $p(Y|\lambda_c)$ is then computed using (5.8), and the class is assigned with respect to the MAP estimate (5.11). In principle it is possible to manually set the prior probabilities $p(c)$ to represent natural relation between action, or to estimate $p(c)$ from data. In the rest of this thesis, we used a uniformly distributed prior.

5.2.4. Viterbi Path

As discussed earlier, states q are usually assumed to be hidden. Nevertheless, given an observation $Y = y_1, y_2, \ldots, y_T$ we can ask for the single state sequence $Q^* = q_1^*, q_2^*, \ldots, q_T^*$, which best explains the observation, *i.e.* $Q^* = \arg\max_Q p(Q|Y)$. This is for example of use for action classification with a large HMM network, which was constructed as combination of several class specific smaller HMMs. Although on a subclass level the meaning of states q remains hidden, on a global level these sates become now associated with different action classes. Consequently, computation of the optimal state sequence of an observation through such a HMM network will provide a simultaneous segmentation and classification of the observation sequence. The most common technique used to compute such an optimal path is the Viterbi algorithm [Rabiner, 1990], which is efficiently based on dynamic programming.

5.2.5. Learning HMM Parameters

Estimating the parameters λ is by far the most difficult problem with an HMM, because there exists no analytical solution in closed form. Instead parameters of an HMM are learned in an iterative optimization procedure, which in context of HMMs is known as the *Baum-Welch method* [Rabiner, 1990]. In fact, the Baum-Welch method is identical to *expectation maximization* (EM), which is a well known and frequently used parameter estimation technique for static latent-variable models.

Given a set of training sequences $\mathcal{Y} = \{Y_1, Y_2, \ldots\}$, EM seeks to estimate parameters λ such that to maximizes the likelihood

$$p(\mathcal{Y}|\lambda) = \prod_{Y \in \mathcal{Y}} p(\mathcal{Y}|\lambda). \tag{5.12}$$

To that aim, the EM procedure iterates between the two steps which are known as *expectation* and *maximization*.

1. E-step — Using the current parameter values and training data, the expected sufficient statistics (ESS) are estimated. For an HMM the ESS are: number of expected transitions

from state i to state j, and number of expected observations u while being in state i.

2. M-step — The parameters are reestimate using the ESS, which for HMMs is achieved by normalizing the ESS into conditional probabilities.

Note that efficient implementations of the EM iteration usually use the forward-backward algorithm to estimate the ESS.

It can be shown that the above procedure will always converge to a maximum. However, $p(\mathcal{Y}|\lambda)$ is generally highly non-convex, and therefore prone to local maxima. The outcome of EM therefore strongly depends on the initial parameter choice for λ.

5.3. Discrete HMM and Vector Quantization

In the above description of HMMs, observations where assumed to be be discrete, *e.g.* $y \in \{1, \ldots, M\}$. The HMM is thus called a *discrete HMM*. In case that observations are continuous, *e.g.* $y \in \mathbb{R}^n$, a simple approach to use such observation with a discrete HMM is vector quantization (VQ). In VQ each observation is assigned to one of M previously selected *codeword* vectors $\mathsf{V} = \{\mathsf{v}_{i \in [1 \ldots M]}\} \in \mathbb{R}^n$, which together form the so called VQ codebook. Discretization of y follows then typically a nearest neighbor rule

$$\mathrm{VQ}(y) = \arg\min_i d(y, \mathsf{v}_i), \tag{5.13}$$

where d is usually the Euclidean distance.

While VQ is an efficient way to generate discrete observations y, such a discretization can cause a strong degradation, which becomes especially evident when we consider a very sparse set of codewords V in a high dimensional space. In this case, many observations may not be accurately described trough a single prototype instance. To overcome such limitations, we can use a continuous output HMM, as described in the next section.

5.4. Continuous and Semi-Continuous HMM

Instead of quantizing continuous observations into discrete labels, we can directly use continuous observation probabilities with an HMM. While such probabilities can be generally of any form, not all results in models which can be estimated in an efficient manner. An important class of continuous HMMs uses mixtures of Gaussian observation probabilities, *i.e.*

$$p(y|q = i) = \sum_j^M p(x = j|q = i)\mathcal{N}(y|\mu_{ij}, \Sigma_{ij}), \tag{5.14}$$

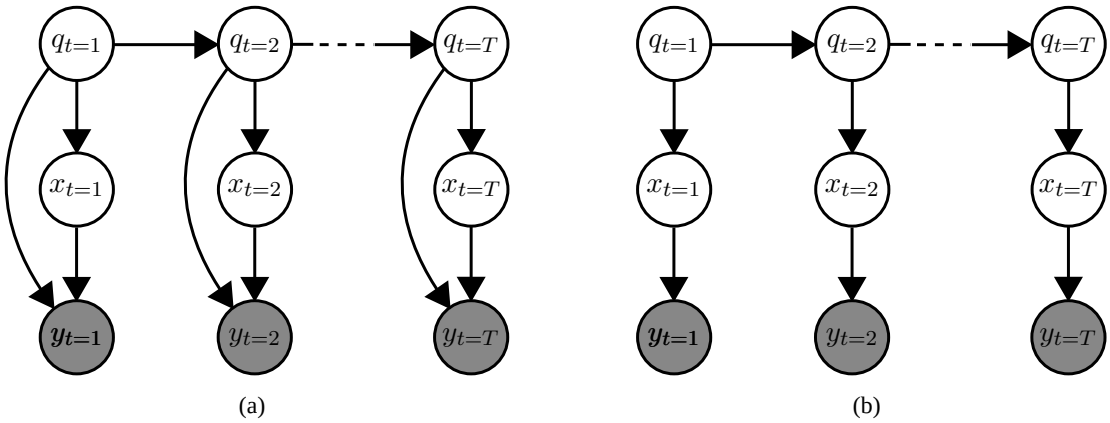


Figure 5.2.: Continuous hidden Markov models. (a) HMM with mixture of Gaussian output. (b) In a semi-continuous HMM all states q share the same set of mixture density functions $p(y|x)$ but with different mixture coefficients $p(x|q)$.

where discrete probability $p(x = j|q = i) = c_{ij}$ is *mixture coefficient*, and $\mathcal{N}$ are normal distributions parameterized by mean μ_{ij} and covariance Σ_{ij}. Such models can be estimated very efficiently using a slightly modified version of the Baum-Welch algorithm. The resulting graphical model is shown in Figure 5.2(a).

Another form of continuous output HMM is the so called *semi-continuous* or *tied mixture* HMM, Figure 5.2(b). The semi-continuous HMM is a simplification of the Gaussian mixture HMM, where all states share the same set of mixture density functions, *i.e.* $\mu_{ij} = \mu_j, \Sigma_{ij} = \Sigma_j, \forall i \in N$, and only the mixture coefficients c_{ij}, *i.e.* the probability that a certain mixture component appears within a certain state, is different between states. Such a model is especially appropriate in case of limited amount of training data, as it has only a single set of Gaussians to estimate.

The semi-continuous HMM can as well be interpreted as an extension of the VQ discrete HMM, where each codeword is now represented through a mean vector and a covariance. It is hence an improvement over the discrete HMM, because it does not distort observations trough the deterministic quantization operation. Instead each observation is probabilistically expressed in terms of all codewords. Moreover, it has the advantage to learn the codebook simultaneously with the HMM.

5.5. Exemplar-based HMM

A difficulty in applying HMMs for action recognition is when the space of observations is not Euclidean. In those cases, the mean and variance cannot be defined. To resolve that problem,

exemplar-based HMMs were introduced by Jojic et al. [2000] and Toyama and Blake [2001], and later also by Elgammal et al. [2003]. The structure of an exemplar-based HMM is generally similar to the previously introduced semi-continuous HMM, and both share in fact the same graphical description shown in Figure 5.2(b). The novelty is, that mixture density functions $p(y|x)$ are no longer centered on arbitrary mean values μ_j, but instead mixture density functions $p(y|x)$ are explicitly centered on prototypical data instances, the exemplars. In the following we notate such exemplars $\mathsf{X} = \{\mathsf{x}_{i\in[1\ldots M]}\} \in \mathbb{X}$, where $\mathbb{X}$ is the space of all data instances of a certain class, e.g. images, shapes, visual-hulls, *etc*. Often advanced distance measures centered on the exemplars are used in $p(y|x)$, *i.e.*

$$p(y|x=i) = \frac{1}{Z_i} \exp\big(-d(y,\mathsf{x}_i)/\sigma_i^2\big), \tag{5.15}$$

with d being any distance function defined over $\mathbb{X}$. For example in Toyama and Blake [2001], a specialized image distances, *e.g.* a distance with searches for the best alignment between image regions, and the chamfer-distance, *e.g.* a distance specially defined between binary edge images, was used for d.

Importantly for exemplar-based HMMs, the estimation of parameters in (5.15) is no longer coupled with the HMM estimation. Exemplars X are usually selected previously, and fixed during the HMM estimation. A simple solution therefore is to cluster or subsample the exemplars in the training sequences. A better technique based on a discriminative feature selection will be discussed in Sections 5.5.2 and 7.3. Estimation of variance σ_i and normalization constant Z_i is generally not straightforward. The work of Toyama and Blake [2001] deals extensively with this issue and provides and approximation for both σ_i and Z_i, which can be estimated from data. Another solution to estimate these parameters is based on the *PDF projection theorem* [Minka, 2004]. A more simple solution, proposed by Elgammal et al. [2003], is to use a non-parametric density for (5.15), such that independent of x all distributions share the same manually adjusted parameter $\sigma_i = \sigma, i = 1,\ldots,M$. Under such a setting and with certain independence assumptions, Z_i can be ignored.

5.5.1. Transformed Exemplar-based HMM

The idea behind the transformed HMM [Jojic et al., 2000] is to explicitly model view transformation as latent process in a HMM. Therefore a latent variable l, which describes a finite set of possible view transformations $P_l, l = 1,\ldots,L$, is insert into the HMM. P_l can thereby represent various classes of transformations, simple 2D similarity transformations, *e.g.* translation, scale, and in-plane rotation. In Chapter 6, we will extend this idea for handling out-of-plane rotation as well. Temporal constraints applied onto the view parameters l, in form of Markovian

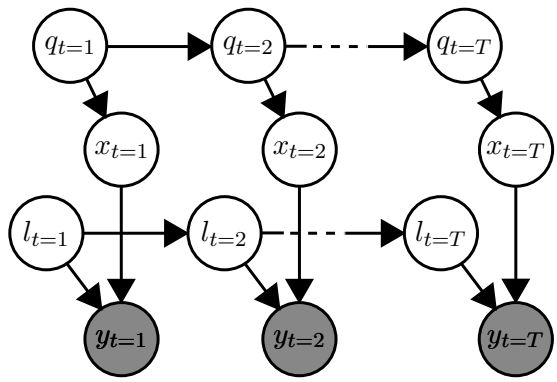


Figure 5.3.: Transformed exemplar based HMM.

dependencies $p(l_t|l_{t-1},\ldots,l_1) = p(l_t|l_{t-1})$, allow them to evolve over time as with a smoothly moving camera. The resulting model is shown in Figure 5.3.

In the resulting *transformed exemplar-based HMM* observation are explained as transformed exemplars $P(\mathsf{x})$. Consequently the probability of observing y, given x and l is

$$p(y|x=i,l=j) = \frac{1}{Z}\exp\big(-d(y,P_j(\mathsf{x}_i))/\sigma_i^2\big). \tag{5.16}$$

Inference and learning with the model, Figure 5.3, is generally not straightforward as it consists two independent random processes, which results in a *loopy graph*, *i.e.* a graph with more than one possible path between two nodes. However, for a reasonable small number of transformations L and states N, a simple solution is to introduce a new variable $\acute{q} = (q,l)$ of size $L \times N$, which encodes both, state and transformation. Probabilities of this *extended* states are then simply defined as Cartesian products of the transition probabilities for q and l, *i.e.* $p(\acute{q}_t|\acute{q}_{t-1}) = p(q_t|q_{t-1})p(l_t|l_{t-1})$. The resulting model consist a single Markov process, and its graphical structure equals the semi-continuous HMM shown in Figure 5.2(b). Inference and learning of state transitions and initial probabilities follows therefore the standard estimation algorithms for HMMs.

5.5.2. Exemplar Selection

As mentioned earlier, exemplars are typically selected separately and prior to the learning of the remaining HMM parameters. In a classical way, such selection has to deal with two conflicting objectives. First, the set of exemplars must be small to avoid learning and classification in high dimensions (*curse of dimensionality*) and to allow for fast computations. Second, the set must contain enough elements to account for variations within and between classes. We will use the wrapper technique for feature selection Kohavi and John [1997], which we first used for

exemplar selection in [Weinland et al., 2007], but other possibilities will be discussed in Section 5.5.3.

Several criteria exist to measure and optimize the quality of a feature set (see *e.g.* [Guyon and Elisseeff, 2003]). The wrapper approach can be seen as a direct and straightforward solution this problem. The criterion optimized is the validation of the considered classifier (in this thesis the MAP classifier (5.11)), which is itself used as a black box by the wrapper while performing a greedy search over the feature space. There are different search strategies for the wrapper and we use a *forward selection*, which we describe in the following.

Forward Selection

The wrapper method we use is called "forward selection" [John et al., 1994], and proceeds as follows: Forward selection is a bottom-up search procedure that adds new exemplars to the final exemplar set one at a time until the final set is reached. Candidate exemplars are all instances in the training set, or a sub-sampled set of those. In each step of the selection, classifiers for each candidate exemplar set are learned and evaluated. Consequently, in the first iteration classifier for each single candidate exemplar are learned, the exemplar with the best evaluation performance is added to the final exemplar set, and the learning and evaluation step is repeated using pairs of exemplars (containing the already selected), triples, quadruples, *etc*. The algorithm is given below (see Algorithm 1).

Algorithm 1 Forward Selection

Input: training sequences $\mathcal{Y} = \{Y_1, \dots Y_m\}$, validation sequences $\hat{\mathcal{Y}} = \{Y_1, \dots Y_{\hat{m}}\}$

1. let candidate exemplar set $\mathcal{X} = \{y : y \in \mathcal{Y}\}$
2. let final exemplar set $X = \emptyset$
3. while size of X smaller than n
   a) for each $y \in \mathcal{X}$
      i. set $X' \leftarrow \{y\} \cup X$
      ii. train classifier g with $\mathcal{Y}$ and keep validation performance on $\hat{\mathcal{Y}}$
   b) set $X \leftarrow \{y^*\} \cup X$ where y^* corresponds to the best validation performance obtained in step 3(a). If multiple y^* with same performance exist, randomly pick one.
   c) set $\mathcal{X} \leftarrow \mathcal{X} \setminus \{y^*\}$
4. return X

Note that the above procedure assumes one shared exemplar set for all action models.

5.5.3. Selection Discussion

Many techniques have been used in the literature to select exemplars and vocabulary sets in related approaches. For instance, several methods sub-sample or cluster the space of exemplars, *e.g.* [Toyama and Blake, 2001, Athitsos and Sclaroff, 2003]. While generally applicable in our context, such methods require nevertheless very large sets of exemplars in order to reach the performance of a smaller set that has been specifically selected with respect to an optimization criterion. Moreover, as we observed in Section 5.5.2, a clustering can miss important discriminative exemplars, *e.g.* clusters may discriminate body shapes instead of actions.

Another solution is to select exemplars based on advanced classification techniques such as support vector machines [Vapnik, 1998] or Adaboost [Freund and Schapire, 1995]. Unfortunately, support vector machines are mainly designed for binary classifications and, though extensions to multiple classes exist, they hardly extract a single feature set for all classes. On the other hand, Adaboost[Freund and Schapire, 1995] can be extended to multiple classes and is known for its ability to search over large numbers of features. Using the framework introduced in Chapter 7, we experimented with Adaboost using weak classifiers based on single exemplars and pairs of exemplars but performances were less consistent than with the forward selection.

Wrapper methods, such as the forward selection, are known to be particularly robust against over-fitting [Guyon and Elisseeff, 2003] but sometimes criticized for being slow due to the repetitive learning and evaluation cycles. In combination with the framework introduced in Chapter 7, we need approximately $n \times m$ learning and validation cycles to select n features out of a candidate set with size m. With a non-optimized implementation in MATLAB, selection of approximately 50 features out of a few hundreds will take around 5 minutes. This is a very reasonable computation time considering that this step is only required during the learning phase and that a compact exemplar set will benefit to all recognition phases.

CHAPTER 6

Action Recognition from Arbitrary Views using 3D Exemplars.

We now address the problem of learning view-independent, realistic 3D models of human actions, for the purpose of recognizing those same actions from a single or few cameras, without prior knowledge about the relative orientations between the cameras and the subjects. A major enhancement with regard to our previous work on MHVs is, that we no longer require a 3D reconstruction during the recognition phase. Instead we will use learned, exemplar-based 3D models to produce 2D image information that is compared to the observations. Consequently, actions can be observed with any camera configuration, from single to multiple cameras, and from any viewpoint. Our main motivation is to cope with unknown recognition scenarios without learning multiple and specific databases. This has many applications, including video-surveillance, where actions are often observed from a single and arbitrary viewpoint.

The requirement to perform recognition from single views is a very strong one, with the effect that whatever representation is built of the 3D action, there should be an efficient algorithm for projecting it into an arbitrary view. The simplest model in this respect uses exemplars of ”poses” described with occupancy grids. The projection of an MHV to a single view is not a simple operation. In particular it is not an MHI. This precludes the use of MHVs in this chapter, although we refer the reader to the discussion in Section 6.6.3.

In this chapter we proceed as follows. In 6.1 we contrast our approach with similar techniques used in motion capture (MOCAP). In Section 6.2 we present an overview of the proposed approach. Details on the exemplar-based HMM are given in Section 6.3. In Section 6.4 the exemplar selection and the model learning are explained. Section 6.5 details recognition. Experiments using a challenging dataset of 11 actions are presented and discussed in Section 6.6,

before we conclude the chapter in Section 6.7.

6.1. Motivation

Our approach shares some similarities with generative models used in recent work dealing with markerless motion capture. Vision based motion capture approaches, such as [Gavrila and Davis, 1996, Deutscher et al., 2000, Sidenbladh et al., 2000, Peursum et al., 2007, Knossow et al., 2008], recover a body model from images by searching for the 3D configuration that, when projected into 2D, best explains the image observation. By using a 3D model to generate 2D information, such approaches can thus adjust to arbitrary views. On the downside, modeling the full human body kinematics, possibly in combination with unknown view parameters, result in a large parameter space, which makes the search for the optimal body configuration a difficult highly non-convex problem. As a result, such approaches are prone to local maxima and are very difficult to calibrate.

In this chapter we propose a generative model similar to MOCAP, with the difference that we represents postures simply through a fixed set of previously selected discriminative 3D key-postures, the exemplars. We will show that using a set of exemplars, instead of a parametric joint model, is a sufficient representation for most actions. This follows our personal experience: Humans can recognize action without exactly reconstructing body configuration, *e.g.* as also demonstrated in the experiments by Bobick and Davis [1996b] with highly blurred imagery of human motion (see Section 2.1.2), and humans can recognize most actions from temporally sparse yet discriminative sequences of key-frames, *e.g.* when reading a comic book (an excellent comic book, which deals with exactly this issue is [McCloud, 1993]).

The view-independent action mode that we propose in this chapter is a generative model with a very low dimensional parameter space. That is, our model has variables for global position and orientation of the body, and a single variable to represent posture, *i.e.* an index to the actual key-pose exemplar. Using this representation, actions are modeled as Markov sequences over exemplar index and view parameters. The resulting model is a transformed exemplar-based HMM, in the spirit of [Frey and Jojic, 2000, Toyama and Blake, 2001]. Such a model allows to use standard inference techniques to identify the action sequence that best explains the image observations. In addition, explicitly modeling the view transformation between exemplars and image cues allows such transformation to change over time during recognition as with a smoothly moving camera.

6.2. Overview of Approach

We model an action as a Markov sequence over a set of key-poses, the exemplars. Figure 6.1 shows examples of observation sequences and the corresponding best matching exemplar sequences computed as Viterbi path (Section 5.2.4) through our model.

Exemplars are represented in 3D as visual hulls, which we computed using a system of 5 calibrated cameras. The observation sequence comes in this example from a single camera and is represented trough silhouettes obtained from background subtraction. To match observation and exemplars, the visual hulls are projected into 2D and a match between the resulting silhouettes is computed. The recognition phase thus generates 2D from 3D and never has to infer 3D from a single view observation.

Modeling actions and views The matching between model and observation is represented in a probabilistic framework (Section 6.3). Consequently, and crucially, that neither the best matching exemplar sequence, nor the exact projection parameters need to be known. Instead a probability of all potential exemplar sequence and projection is computed. Using the classical HMM algorithms (Section 5), such a probability can be efficiently computed under the following conditions: First, we use a small set of exemplars that is shared by all models. As we show in Section 5.5.2, a small set of exemplars is sufficient to describe a large variety of actions, if the exemplars are discriminative with respect to these actions. Second, we make a few reasonable assumptions on the parameters of the projective transformation, *i.e*. the camera calibration and position of a person can be robustly observed during recognition and only the orientation of a person around the vertical axis is unknown.

Exemplar selection and model learning Learning an action model consists of two steps: A set of exemplars is selected, which is shared by all actions models (Section 5.5.2); probabilities over these exemplars are learned individually for each action (Section 6.4.2).

When selecting the exemplars, we are interested in finding the subset of poses from the training sequences, that bests discriminates actions. To this purpose, we present in Section 5.5.2 an approach based on a method for feature subset selection, a *wrapper* [John et al., 1994].

Given a set of exemplars, the action specific probabilities are estimated using standard probability estimation techniques for HMMs, as described in Section 6.4.2. Interestingly, the learning of dynamics over a set of selected 3D exemplars can be performed either on 3D sequences of aligned visually hulls, thus under ideal conditions, or simply from single view observations. Hence 3D information is not mandatory for that step.

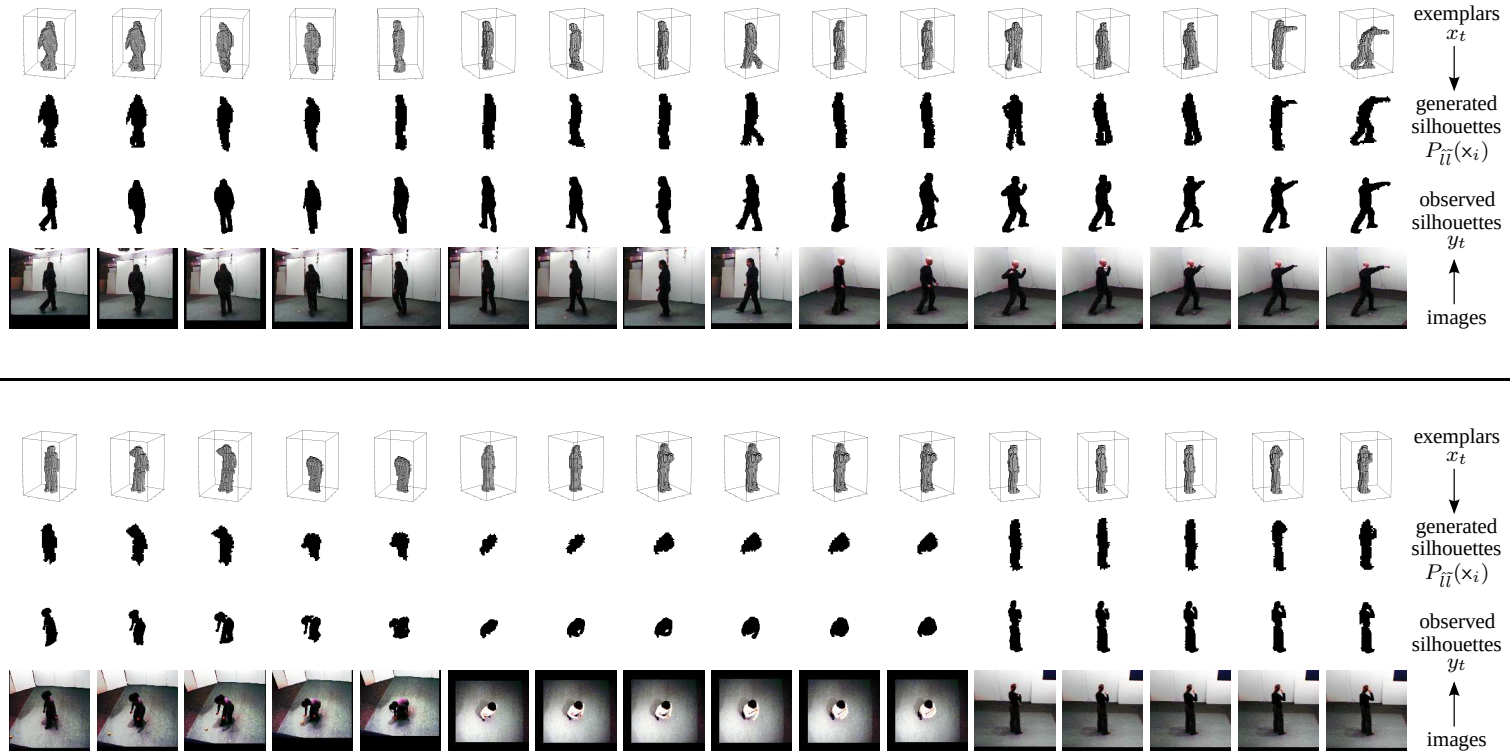


Figure 6.1.: 2D observation sequences y_t (Top: *walk in cycle* and *punch*, Bottom: *pick up*, *cross arms*, and *scratch head*), observed from different viewpoints and with unknown orientation of the persons, are explained trough 3D action models. The best matching exemplar sequence x_t and the best matching 2D projection $P_{\tilde{l}\tilde{l}}(\mathsf{x}_i)$, as generated by the models, are displayed. The models share a small set of exemplars.

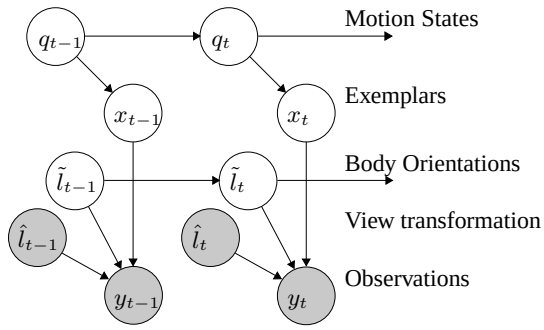


Figure 6.2.: Probabilistic dependencies of actions: an action is modeled as a hidden state sequence Q, *e.g.* a motion sequence in a posture space. At each time step t, a 3D exemplar x_t, *i.e.* a visual hull, is drawn from the motion sequence Q. Observations y_t, *i.e.* silhouettes, result then from a geometric transformation of exemplars that is defined by 2 sets of parameters $\hat{l}$ and $\tilde{l}$. $\hat{l}$ are observed parameters, *e.g.* camera parameters determined in a preliminary step, and $\tilde{l}$ are latent parameters, *e.g.* body orientation determined during recognition. Shaded nodes in the graph correspond to observed variables.

Classification Classification is performed using standard HMM algorithms, as described in Section 6.5.

6.3. Probabilistic Model of Actions and Views

Our representation for human action is a product of two independent random processes, one for the orientation of the subject relative to the camera, and the other for the view-independent, body-centered poses taken by the performer during the various stages of the action. The two processes are modeled in an exemplar-based HMM, shown in Figure 6.2, in the spirit of [Frey and Jojic, 2000] and [Toyama and Blake, 2001].

Hidden Motion States Dynamics in exemplar space are represented by a discrete N-state latent variable q that follows a first order Markov chain over time. Thus: $p(q_t|q_{t-1},\ldots,q_1) = p(q_t|q_{t-1})$, with $t \in [1\ldots T]$, and with the prior $p(q_1)$ at time $t = 1$. Though generally hidden, q can intuitively be interpreted as a quantization of the joint motion space into action-characteristic configurations.

Exemplars At each time t, a three dimensional body template x_t is drawn from $p(x_t|q_t)$. A crucial remark here is that these templates do not result from body models and joint configurations, but are instead represented by a set of M exemplars: $\mathsf{X} = \{\mathsf{x}_{i\in[1\ldots M]}\}$, learned from three dimensional training sequences.

Note here that $p(x_t|q_t)$ models the non-deterministic dependencies between motion states and body configuration. Thus motion states q are not deterministically linked to exemplars as *e.g.* in [Toyama and Blake, 2001, Lv and Nevatia, 2007], allowing therefore a single motion state q to be represented with different exemplars, to account for different body proportions, style, or clothes.

View Transformation and Observation To ensure independence with respect to the view projection onto the image plane: $P_{\hat{l}\tilde{l}}(\mathsf{x}) = \hat{P}[R_\theta, u]\mathsf{x}$, we condition observations y on parameters that represent this transformation. We separate view transformation parameters $\{\hat{l}_t\}$ computed separately, using robust methods (*i.e.* the camera matrix $\hat{P}$ and position u), and body pose parameters $\{\tilde{l}_t\}$ that are latent (*i.e.* the orientation around the vertical axis θ).

The resulting density $p(y_t|x_t, \hat{l}_t, \tilde{l}_t)$ is represented in form of a kernel function centered on the transformed exemplars $P_{\hat{l}\tilde{l}}(\mathsf{x}_i)$:

$$p(y_t|x_t = i, \hat{l}_t, \tilde{l}_t) \propto \frac{1}{Z} \exp\big(- d(y_t, P_{\hat{l}\tilde{l}}(\mathsf{x}_i))^2/\sigma^2\big), \tag{6.1}$$

where d is a distance function between between the resulting silhouettes, *e.g.* the Euclidean distance (i.e. the number of pixels which are different), or a more specialized distance such as the chamfer distance [Gavrila and Philomin, 1999]. (Note that both were giving similar results in our experiments in this chapter, where we were exclusively working with background subtracted sequences.)

The temporal evolution of the latent transformation variables is modeled as a Markov process with transitions probabilities $p(\tilde{l}_t|\tilde{l}_{t-1})$, and a prior $p(\tilde{l}_1)$. This is equivalent to a temporal filtering of the transformation parameters where, interestingly, various assumptions could be made on the dynamic of these parameters: a static model or an autoregressive model, or even a model taking into account dependencies between an action and view changes.

In our implementation all variables $\{\tilde{l}, \hat{l}\}$ are discretized. For instance, the orientation θ is discretized into L equally spaced angles within $[0, 2\pi]$ and u is discretized into a set of discrete positions. The temporal evolution of θ is modeled using a von Mises distribution: $p(\theta_t|\theta_{t-1}) \propto \exp(\kappa \cos(\theta_t - \theta_{t-1}))$, that can be seen as the circular equivalent of a normal distribution, and a uniform prior $p(\theta_1)$.

6.4. Learning

We learn separate action models λ_c for each action class $c \in \{1, \ldots, C\}$. A sequence of observations $Y = \{y_1, \ldots, y_T\}$ is then classified with respect to the maximum a posteriori

(MAP) estimate:

$$g(Y) = \arg\max_{c} p(Y|\lambda_c)p(\lambda_c), \tag{6.2}$$

see also Section 5.2.3. The set λ_c is composed of the probability transition matrices $p(q_t|q_{t-1}, c)$, $p(q_1|c)$ and $p(x_t|q_t, c)$, which are specific to the action c, as they represent the action's dynamics. In contrast, the observation probabilities $p(y_t|x_t, \hat{l}_t, \tilde{l}_t)$ are tied between classes, meaning that all actions $\{c = 1..C\}$ share a common exemplar set, *i.e.* $\mathsf{X}_c = \mathsf{X}$, and a unique variance $\sigma_c^2 = \sigma^2$. In the context of HMMs, such an architecture is known as a *tied-mixture* or *semi-continuous* HMM[Bellegarda and Nahamoo, 1990], see also Section 5.4. This architecture is particularly well adapted to action recognition since different actions naturally share similar poses. For example, many actions share a neutral rest position and some actions only differ by the sequential order of poses that composed them. In addition, sharing parameters dramatically reduces complexity during recognition, when every exemplar must be projected with respect to numerous latent orientations.

Learning consists then in two main operations: selecting the exemplar set that is shared by all models; learning the action specific probabilities. As we will see in the following, the two operations are tightly coupled. Selection uses learning to evaluate the discriminant quality of exemplars, and learning probabilities relies on a selected set of exemplars. Both operations are detailed below.

6.4.1. Exemplar Selection

Identifying discriminative exemplars is an essential step of the learning process. A general discussion on exemplar selection methods was given in Section 5.5.3. In action recognition, previous works use motion energy minima and maxima [Lv and Nevatia, 2007, Ogale et al., 2004], or k-means clustering (adapted to return exemplars) [Toyama and Blake, 2001] to identify key-pose exemplars. However, there is no apparent relationship between such criteria and the action discriminant quality of the selected exemplars. In particular for the adapted k-means clustering [Toyama and Blake, 2001] we observed experimentally, that clusters tend to consist of different poses performed by similar actors rather than similar poses performed by different actors. Consequently, selecting exemplars as poses with minimum within-cluster distance often leads to neutral and therefore non-discriminative poses.

To better link the discriminant quality of exemplars and the selection, we propose in this thesis to use a novel approach for exemplar selection, which is the wrapper approach introduced in Section 5.5.2.

The selection of exemplar is then performed entirely on 3D sequences of rotational aligned exemplars. Training and evaluation of the remaining probabilities can be performed in 3D or 2D, as detailed in Section 6.4.2. The approach is illustrated in Figures 6.3 and 6.4 where

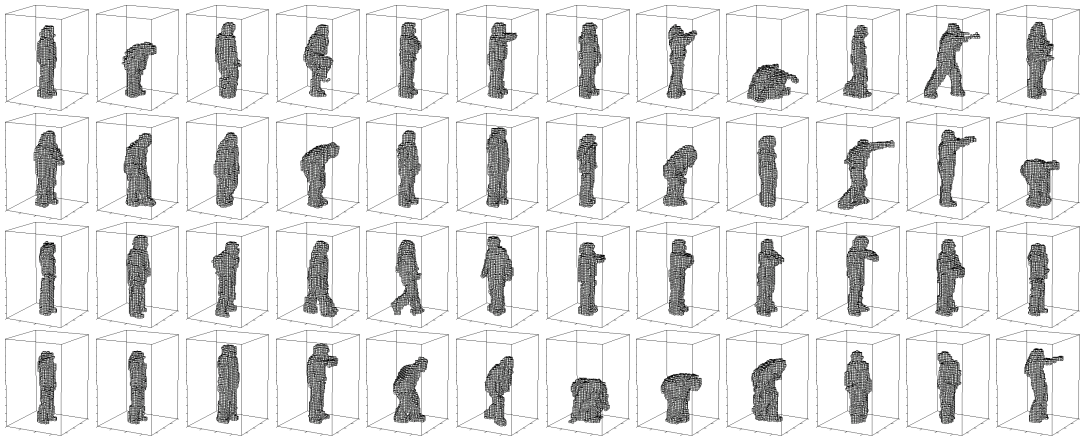


Figure 6.3.: Selected exemplars: first 48 discriminative exemplars as returned by the forward selection. The dataset is composed of 11 actions performed by 10 actors. Recognition rates are shown in Figure 6.4.

exemplars and the associated validation rates are shown. Figure 6.3 shows that the selected poses naturally represent key-postures, *i.e.* characteristic frames of an action.

6.4.2. Learning Dynamics

Given a set of exemplars, the action parameters $\lambda_{c\in\{1,\ldots,C\}}$: probabilities $p(q_t|q_{t-1},c)$, $p(q_1|c)$ and $p(x_t|q_t,c)$, can be learned. Various strategies can be considered for that purpose. In the following, we sketch two of them: learning from 3D observations (sequences of visual hulls), and learning from 2D observations (image sequences). Note that in both cases, motion is learned in 3D over the set of 3D exemplars, obtained as described in section 5.5.2.

Learning from 3D Observations

In this training scenario, several calibrated viewpoints are available, leading therefore to 3D visual hull sequences, and all actions are performed with the same orientation. In that case, motion dynamics are learned independently from any viewing transformation, thus $p(y_t|x_t,\hat{l}_t,\tilde{l}_t) = p(y_t|x_t)$ with y being 3D. Transformation parameters appear later during the recognition phase where both dynamics and viewing process are joined into a single model.

Each model λ_c is learned through a forward-backward algorithm (Section 5.2.2) that is similar to the standard algorithm for Gaussian mixture HMMs [Rabiner, 1990], except that the kernel parameters, that correspond to mean and variance of the Gaussians (*i.e.* X and σ), are not updated. Note that a similar forward-backward algorithm was already proposed in the context of exemplar based HMMs [Elgammal et al., 2003].

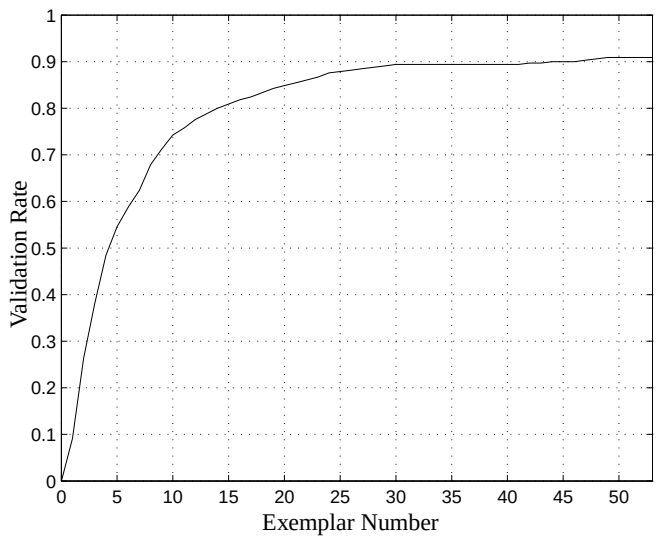


Figure 6.4.: Validation rate vs. number of selected exemplars.

Learning from 2D Observations

In this scenario, dynamics in the exemplar 3D space are learned using 2D cues only. In that case, the situation is similar when either learning or recognizing. A nice feature here is that only a valid set of 3D exemplars is required, but no additional 3D reconstruction. This is particularly useful when large amounts of 2D observations are available but no 3D inference capabilities (*e.g.* 3D exemplars can be synthesized using a modeling software; the dynamics over these exemplars are learned form real observations).

View observations are not aligned and so the orientation variable $\tilde{l}$ is latent. Nevertheless, the number of latent states remains in practice small, (*i.e.* $L \times N$, with L being the number of discrete orientations $\tilde{l}$ and N the number of states q). The model can be learned by introducing a new variable $\acute{q} = (q, \tilde{l})$ of size $L \times N$ that encodes both state and orientation, as explained in Section 5.5.1. Loops in the model are thus eliminated, and learning can be performed via the forward-backward algorithm introduced in Section 5.2.2.

6.5. Action Recognition from 2D Cues

A sequence of observations Y is classified using the MAP estimate (6.2), as explained in Section 5.2.3. Such a probability can now be computed using the classical forward variable (Section 5.2.2) $\alpha(\acute{q}_t|\lambda_c) = p(y_1, \ldots, y_t, \acute{q}_t|\lambda_c)$, where $\acute{q} = (q, \tilde{l})$ is a variable encoding state and orientation as explained in Section 6.4.2

Arbitrary viewpoints do not share similar parameters; in particular scales and metrics can

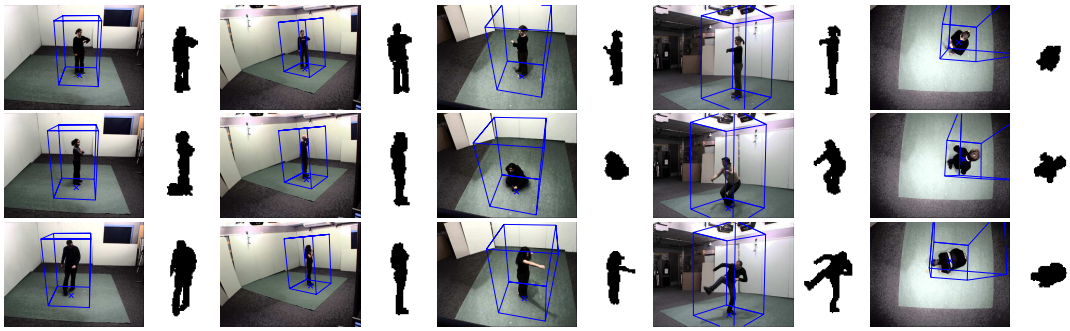


Figure 6.5.: Camera setup and extracted silhouettes: (Top) the action "watch clock" from the 5 different camera views. (Middle and bottom) sample actions: "cross arms", "scratch head", "sit down", "get up", "turn", "walk", "wave", "punch", "kick", and "pick up". Volumetric exemplars are mapped onto the estimated interest regions indicated by blue box.

be different. However, the kernel parameter σ^2 is uniquely defined, with the consequence that distances computed in equation (6.1) can be inconsistent when changing the viewpoint. To adjust σ^2 with respect to changes in these parameters, we introduce $\sigma^2_{\tilde{l}} = s_{\tilde{l}}\sigma^2$. Ideally, $\sigma^2_{\tilde{l}}$ should be estimated using test data. In practice, the following simple approximation of $\sigma^2_{\tilde{l}}$ appears to give satisfactory results with the distance functions we are considering:

$$s_{\hat{l}} = \frac{1}{M}\sum_{i=1}^{M}\frac{\frac{1}{L}\sum_{\tilde{l}=1}^{L}||P_{\hat{l}\tilde{l}}(\mathsf{x}_i)||^2}{||\mathsf{x}_i||^2}. \tag{6.3}$$

The idea is here generally that of a Monte-Carlo method, where we sample points in the new space and use the estimated average change in scale to update the kernel parameter. For efficiency, the method simply takes the set of projected exemplars as samples in the new space.

Another remark is that observations from multiple calibrated cameras can easily be incorporated. Assuming multiple view observations $\{y_t^1, \ldots, y_t^K\}$ at time t, we can write their joint conditional probability as:

$$p(y_t^1, \ldots, y_t^K | x_t, \hat{l}_t, \tilde{l}_t) \propto \prod_{y_t^k}^{K} p(y_t^k | x_t, \hat{l}_t, \tilde{l}_t). \tag{6.4}$$

6.6. Experiments

Experiments were conducted on the IXMAS dataset, see Section 3.3.1 and Figure 6.5. Our experimental scheme is as follows: 9 of the actors are used for exemplar selection and model learning, the remaining actor is then used for testing. We repeat this procedure by permuting

cameras	2 4	3 5	1 3 5	1 2 3 5	1 2 3 4
%	81.6	61.6	70.2	75.9	81.6

Table 6.1.: Recognition rates with camera combinations. For comparisons, a full 3D recognition considering 3D manually aligned models as observations, instead of 2D silhouettes, yields 91.11%.

the test-actor and compute the average recognition rate. Exemplar selection is performed on sub-sampled sequences (*i.e.* 2.5 frames/s) to save computational costs. Example results for exemplars are shown in Figure 6.3. The number M of exemplars was empirically set to 52 . Parameter learning and testing is performed using all frames in the database. Action are modeled with 2 states, which appears to be adequate since most segmented actions cover short time periods. Voxel grids are of size: $64 \times 64 \times 64$ and image ROIs: 64×64. The rotation around the vertical axis is discretized into 64 equally spaced values. Consequently, each frame is matched to 52×64 exemplar projections. The ground plane is clustered into 4 positions.

6.6.1. Learning in 3D

In these experiments, learning is performed in 3D (as explained in 6.4.2). Recognition is then performed on 2D views with arbitrary actor orientations. Recognition rates per camera are given in Figure 6.6, the corresponding views are shown in Figure 6.5.

Unsurprisingly, the best recognition rates are obtained with fronto-parallel views (cameras 2 and 4). The top camera (camera 5) scores worst. For this camera, we observe that: the silhouette information is not discriminative; the perspective distortion results in strong bias in distances; estimating the position of the actor is difficult. All these having a strong impact on the recognition performance.

In the next experiment, several views were used in conjunction to test camera combinations. First, 2 view combinations were experimented. Camera 2 and 4 give the best recognition rate at 81.59%. Those 2 cameras are both approximately fronto-parallel and perpendicular one another. Figure 6.7 shows the resulting confusion matrix for this specific setup. Adding further cameras did not improve results. We also try other camera combinations (Table 6.1). For instance, combining the two cameras with the worst recognition results (camera 3 and 5) raises the recognition rate to 61.59%.

6.6.2. Learning from single views

In this experiment, learning is performed using single cameras (as explained in Section 6.4.2). Observations during learning and recognition are thus not aligned. The exemplars considered

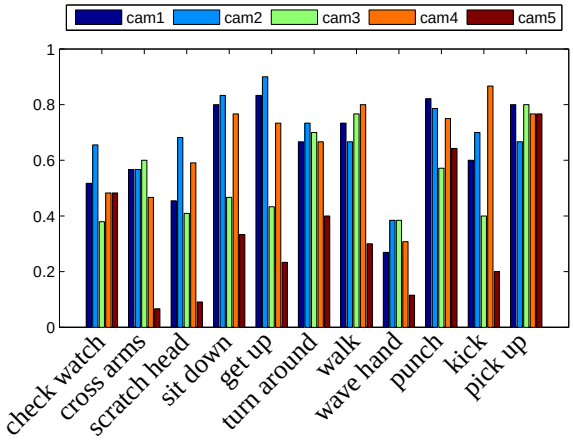


Figure 6.6.: Recognition rates when learning in 3D and recognizing in 2D. The average rates per camera are $\{65.4, 70.0, 54.3, 66.0, 33.6\}$.

	check watch	cross arms	scratch head	sit down	get up	turn around	walk	wave hand	punch	kick	pick up
check watch	.86	.00	.00	.00	.00	.07	.03	.03	.00	.00	.00
cross arms	.13	.73	.00	.00	.00	.03	.00	.03	.07	.00	.00
scratch head	.00	.09	.68	.00	.00	.00	.00	.09	.09	.05	.00
sit down	.00	.00	.00	.93	.07	.00	.00	.00	.00	.00	.00
get up	.00	.00	.00	.00	.93	.07	.00	.00	.00	.00	.00
turn around	.00	.00	.00	.00	.00	.97	.00	.03	.00	.00	.00
walk	.00	.00	.00	.00	.00	.33	.67	.00	.00	.00	.00
wave hand	.04	.04	.27	.04	.04	.00	.00	.50	.08	.00	.00
punch	.00	.00	.00	.04	.00	.04	.00	.00	.82	.00	.11
kick	.00	.00	.00	.00	.00	.07	.00	.00	.00	.90	.03
pick up	.00	.00	.00	.10	.00	.00	.00	.00	.03	.00	.87

Figure 6.7.: Confusion matrix for recognition using cameras 2 and 4. Note that actions performed with the hand are confused, *e.g.* “wave” and “scratch head” as well as “walk” and “turn”.

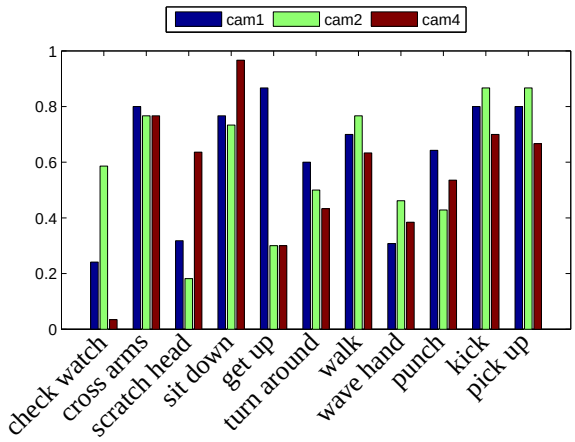


Figure 6.8.: Recognition rates when learning and recognizing in 2D.

are the same than in the previous section. Learning from a single view is obviously prone to ambiguities, especially when the number of training samples is limited. We thus restricted the experiments to the 3 best cameras with respect to the previous experiments. Figure 6.8 shows the recognition results per action class and per camera. Compared to the previous scenario, recognition rates drop drastically, as a consequence of learning from non-aligned data and single view observations. Surprisingly, some of the actions, *e.g.* "cross arms", "kick" still get very acceptable recognition rates, as well as "sit down" and "pick up" that would normally be confused. The average rate for camera 1 is 55.24%, 63.49% for camera 2 and 60.00% for camera 4.

6.6.3. Comparison with MHVs

In this section we compare the performance of the two view-independent approaches, MHVs and exemplar-based HMM, on the IXMAS dataset. Best recognition rate for MHVs was 93.33%, using PCA plus Mahalanobis distance based normalization, while the exemplar-based HMM had best results with 81.3%, when fusing the views of several cameras. A direct comparison of the results indicates thus, that a 3D reconstruction is more powerful than simple fusing of observation likelihoods form multiple views, even though camera calibration information was used in the latter to impose consistency between the views.

When comparing the two approaches it is important to notice that there are several issues, which only exist when working with 2D observations. An example is the scale normalization, which we applied in 3D along the main directions of the cylindrical representation before matching the MHVs. Such scale is easily estimate in 3D. In 2D, however, given only single silhouettes we can not infer 3D scale. Respectively, for the exemplar based approach all volumes were kept at their original scale when matched against 2D, and variations in scale were only

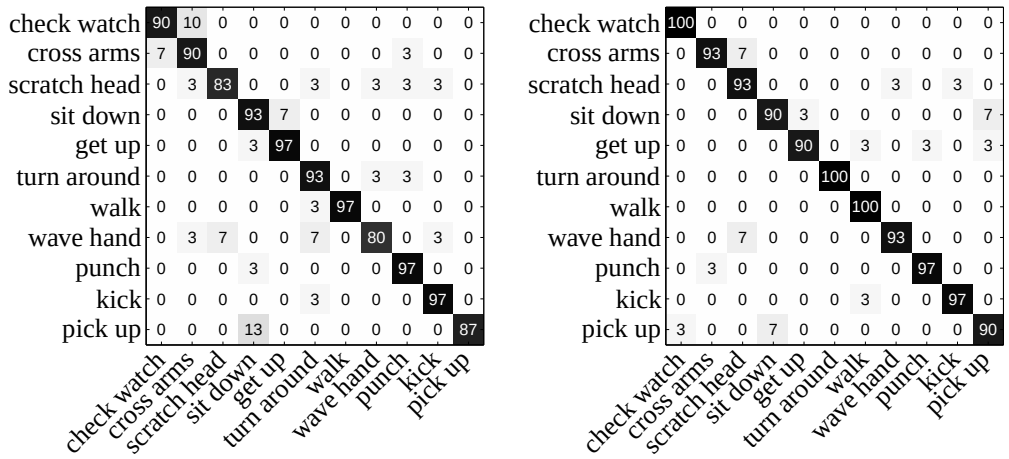


Figure 6.9.: Comparison exemplar-based HMM vs. MHVs: (Left) Confusion matrix (in %) using exemplar based HMM with average rate 91.21%. (Right) using MHVs with average rate 94.85%.

explained through different sized exemplars in the training set (resulting from different sized actors).

To directly compare the quality of two different action models, *i.e.* space-time representation and state-transition model (Section 2.2), we perform experiments with both models in 3D, on manually aligned volumes. As we reported in Section 3.3.6, the recognition ratio with MHVs under this setting is 94.85%. For a exemplar based HMM that we learn and evaluate using 3D exemplars, the best recognition rate is 91.21%. The two approaches have thus similar recognition rates under such a unified setting, although the space-time approach is slightly better (*i.e.* 3.74%). Confusion matrixes for this experiment are shown in Figure 6.9.

In summary, the space-time MHV representation has very good recognition results and is very efficient to compute. On the other hand, the exemplar-based HMM scores slightly worse, but therefore overcomes one of the main limitations of the MHV approach, *i.e.* dependency on multiple views.

A Generative Space-Time Approach?

In the previous comparison between the MHV space-time representation and exemplar-based model, the former had slightly better results, while the latter was able to work from single view observations. This respectively suggests to implement a combination of both approaches, *i.e.* an approach that uses 3D MHVs to generate arbitrary 2D MHI observations. While this is indeed an interesting direction, it will result in several difficulties as explained in the following.

In particular, generating 2D motion-templates from a 3D model, other than simple silhouettes, is *not* a straightforward process. We exemplary illustrate this for 3D MHVs and 2D MHIs

(a) (b) (c)

Figure 6.10.: Issues when generating 2D MHIs from a 3D MHV model. (a) The 2D projection of a 3D MHV computed from visual-hulls sequence is different from (b) the MHI computed from 2D projections of the sequence. (c) Modifying the rendering pipeline we can generate an equivalent 2D MHI directly from the 3D MHV.

in Figure 6.10, but similar issues exist for other space-time representations, *e.g*. optical flow, and non-exemplary representation, *e.g*. statistically represented space occupancy.

In the top row of Figure 6.10 silhouettes of a sample *kick* motion are shown. Figure 6.10(a) shows the resulting MHV that was computed on the 3D volumes of this motion and projected onto the image plane, while Figure 6.10(b) shows the MHI that was computed in 2D after the silhouettes were each separately projected onto the image plane. Clearly, the 2D MHI does not correspond to the projected 3D MHV. Hence a generative approach based on MHVs can not simply project 3D MHVs into 2D and match these against MHIs computed from 2D observations. Instead, such an approach would require an explicit projection of all training frames into 2D, and re-computation of the 2D representation for each new view. When working with changing views, such a process can become rapidly over expensive.

Interestingly, for MHVs a solution exists, which requires a re-implementation of the standard 2D rendering pipeline. In short, the rendering pipeline was modified so that for each viewing ray through a pixel not the value (time stamp) of the closest occupied voxel is adopted, as typically for a rendering pipeline, but instead for each viewing ray through a pixel the maximum voxel value (*i.e*. the longest occupied time stamp) along that direction is adopted. The result of such a modified rendering applied to a MHV is shown in Figure 6.10(c), which indeed is equivalent to the MHI in 6.10(b) computed form the 2D silhouettes sequence.

It is however important to recall, that our original MHV approach used a classifier based on averaged MHVs. For such a mean value representation the above rendering would no longer

Latent motion states (No.)	1	2	3	4
Recognition rate (%)	75.87	80.63	81.59	79.05

Table 6.2.: Recognition rates with HMMs of different complexity.

apply. A solution would be a classifier based on exemplary MHVs/MHIs; in the simplest case a nearest neighbor approach. We believe this is an interesting direction for future work.

6.6.4. On Importance of Modeling Dynamics

In this section we experiment with different number of motion states N used in the HMM. Looking at existing work in action recognition, we found that this parameter is usually set empirically, and that few investigations on its importance are presented. To get more insight on the importance of this parameter, we repeat the experiment from Section 6.6.1 using the two cameras 2 and 4 with different values of N. Results are shown in Table 6.2. We observe that best average recognition rate is achieved using $N = 3$ states. In Figure 6.11 we show confusion matrixes for individual actions for the cases $N = 1$ and $N = 3$. We observe, that actions such as *sit down*, *get up*, and *pick up* are more frequently confused if no dynamics are modeled, *i.e.* by using a single state HMM. This is not surprising, because those actions share similar characteristic postures that only differ in temporal order. On the other hand, we observe that actions such as *kick* and *punch* even have slightly worse recognition results when dynamics are added. Such actions that are very characteristic in posture, seem not to benefit from the additional dynamic modeling. Instead, using a more complex dynamical model, with several latent states, complicates the representation and recognition of those actions.

We can thus summarize, that not all actions need dynamic modeling, and that some actions can even benefit from a simplified representation. We will examine this issue further in the next part of this thesis, where we will specially design an exemplar-based representation to be independent of temporal order. We then demonstrate, that indeed many actions, and in particular those currently considered by the computer vision community, can benefit from such a simplified and highly efficient representation.

6.7. Conclusion

This chapter presented a new framework for view independent action recognition from a single or few cameras. To that aim we introduced a probabilistic 3D exemplar model that can generate arbitrary 2D view observations. It results in a versatile recognition method that adapts to various camera configurations. The approach was evaluated on our dataset of 11 actions and with

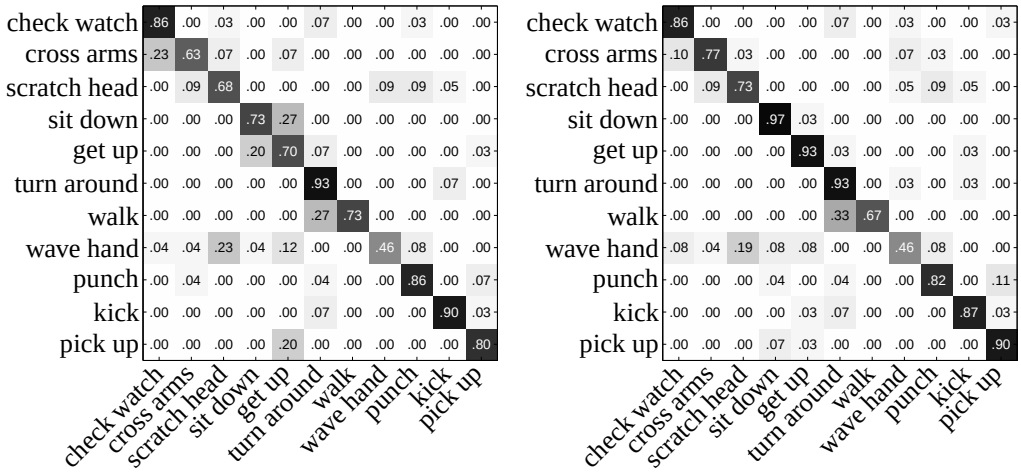


Figure 6.11.: Comparison between (left) HMM with single latent motion state vs. (right) HMM with three motion states: Actions such as *sit down*, *get up*, and *pick up* are evidently confused when modeling no dynamics. On the other hand, actions such as *kick* and *punch* have worse recognition rate when additional complexity is introduced in the model.

different challenging scenarios. The best results where obtained with a pair of fronto-parallel perpendicular cameras, validating the fact that actions can be recognized from view arbitrary viewpoints.

Our experiments using a small exemplar set of approx. 50 different key-poses, which we found sufficient to model 11 different actions performed by 10 different actors, also confirmed what we initially anticipated, *i.e*. that many actions are sufficiently describable through a set of characteristic key-posture, without need for recovery of exact joint configuration.

In the next part of this thesis we develop an approach which addresses several issues that are related to the work in chapter. We experiment with exemplars for recognition in non-background subtracted scenes, and we examine action recognition using exemplars in a simplified representation, which in particular does not account for dynamics.

Part III

Action Recognition without Modeling Dynamics: Exemplar-based Embedding

The previous parts of this thesis were centered around view-independent modeling of actions. In this part, we work on a view-dependent setting, and focus on different issues. We derive a new and strongly simplified exemplar-based representation for action recognition, which in particular does not account for modeling dynamics. This is in contrary to most existing approaches, which model actions with representations that either explicitly or implicitly encode the dynamics of actions through temporal dependencies. The representation proposed in this part does not account for such dependencies. Instead, motion sequences are represented with respect to a set of discriminative static key-pose exemplars and without modeling any temporal ordering. The interest is a time-invariant representation, which drastically simplifies learning and recognition by removing time related information such as speed or length of an action. We demonstrate on a publicly available dataset, that such a representation indeed can precisely recognize actions with result that equal or exceed those of the current state-of-the-art approaches.

CHAPTER 7

Action Recognition using Exemplar-based Embedding

7.1. Introduction

A challenging issue in action recognition originates from the diversity of information which describes an action. This includes purely visual cues, e.g. shape and appearance, as well as dynamic cues, e.g. space-time trajectories and motion fields. Such diversity raises the question of the relative importance of these sources and also to what degree they compensate for each other.

As discussed in Section 2.1.1, Johansson [1973] demonstrated through psychophysical experiments that humans can recognize actions merely from the motion of a few light points attached to the human body. Following this idea, several works (see Section 2.1.1) attempted to recognize actions using trajectories of markers with specific locations on the human body. While successful in constrained environments, these approaches do not however extend to general scenarios.

Besides, static visual information give also very strong cues on activities. In particular, humans are able to recognize many actions from a single image (see for instance Figure 7.1). Consequently a significant effort has been put in representations witch fuse strong visual cues with temporal models. As we discussed in chapter 2, different directions have been followed. Space-time representations, such as the motion history volume approach introduced in Chapter 3, simultaneously model in space and time. Other approaches equip traditional state-transition models, such as hidden Markov models (HMMs), with powerful image matching abilities based on exemplar representations, e.g. see the exemplar-based HMM in Chapter 6.

In this chapter we take a different strategy and represent actions using static visual information without temporal dependencies. Our experiments in the previous chapter using an exemplar-based HMM with a single motion state (Section 6.6.4) already demonstrated, that certain actions do not require dynamic modeling, and that such actions can even benefit from a less complex model. In this chapter we elaborate on this finding and develop a simplified exemplar-based representation, which specially avoids modeling temporal relations. Our results show that such a representation can effectively model actions and yield recognition rates that equal or exceed those of the current state-of-the-art approaches, with the virtues of simplicity and efficiency.

Our approach builds on recent works on example-based embedding methods [Athitsos and Sclaroff, 2003, Guo et al., 2007]. In these approaches complex distances between signals are approximated in a Euclidean embedding space that is spanned by a set of distances to exemplar measures. Our representations is grounded on such embedding, focusing only on the visual components of an action. The main contribution is a time-invariant representation that does not require a time warping step and is insensitive to variations in speed and length of an action. To the best of our knowledge, no previous work has attempted to use such an embedding based representation to model actions.

Exemplars are selected, as in our previous work, using a forward feature selection technique [Kohavi and John, 1997]. To compare our representation to the state of the art, we experiment in this chapter on a view-dependent dataset, the well known Weizmann-dataset [Blank et al., 2005]. Our results confirm that action recognition can be achieved using small sets of discriminatively selected exemplars, and without considering temporal dependencies.

Another important feature of our approach is that it can be used with advanced image matching techniques, such as the Chamfer distance [Gavrila and Philomin, 1999], for visual measurements. In contrast to the classical use of dimensional reduction with silhouette representations, *e.g.* [Wang and Suter, 2007], such a method can be used in scenarios where no background subtraction is available. In a second experiment we will demonstrate, that even on cluttered non-segmented sequences, our method has precise recognition results.

The chapter is organized as follows: In Section 7.2 we present our action representation. In Section 7.3 we show how to compute a small but discriminative exemplar set. In Section 7.4 we evaluate our approach with a publicly available dataset before concluding and discussing issues in Section 7.5.

7.2. Action Modeling

Actions can been recognized using the occurrences of *key-frames*. In the work of Carlsson and Sullivan [2001], class representative silhouettes are matched against video frames to recognize

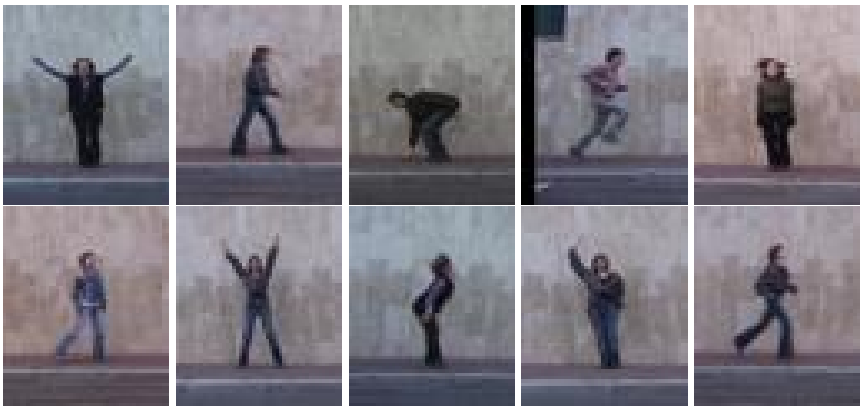


Figure 7.1.: Sample images from the *Weizmann*-dataset [Blank et al., 2005]. A human observer can easily identify many, if not all, actions from a single image. The interested reader may recognize the following actions: *bend*, *jumping-jack*, *jump-in-place*, *jump-forward*, *run*, *gallop-sideways*, *walk*, *wave one hand*, *wave two hands*, and *jump-forward-one-leg*. Note, that the displayed images have been automatically identified by our method as discriminative exemplars.

forehand and backhand strokes in tennis recordings. In a similar way, our approach uses a set of representative silhouette like models, *i.e. the exemplars*, but does not assume a deterministic framework as in [Carlsson and Sullivan, 2001], where exemplars are exclusively linked to classes, and decisions are based on single frame detections.

A non-deterministic, probabilistic framework, using key-frame-like exemplars in an HMM, was introduced in the previous part of this thesis. In such a *top-down* framework, representative exemplars are matched against video frames, and the resulting distances are converted into likelihood probabilities. Consequently, the non-deterministic uncertainty relations between actions, exemplars, and observation, are represented elegantly in terms of probabilities. On the downside, such a framework can rapidly become over complicated, computations of normalization constants and joint probabilities can become infeasible (see also Section 5.5), and consequently many simplifying assumptions become necessary for such an approach to remain computable.

Those difficulties inspired our last approach in this thesis, where we take a different direction and derive an exemplar-based approach from *bottom-up*. As previously, we start with matching representative exemplars against video frames. Instead of converting those into probabilities, however, we simply work on the resulting set of distances. In our experiments we found that learning a simple static classifier over those distances is entirely sufficient to discriminate the actions that we considered. We interpret the resulting set of distances as an embedding into a space defined by distances to key-pose exemplars. Although we use no probabilities, uncertainties are nevertheless preserved in such a space, in terms of distances.

Our approach is illustrated in Figure 7.2. An action sequence is matched against a set of

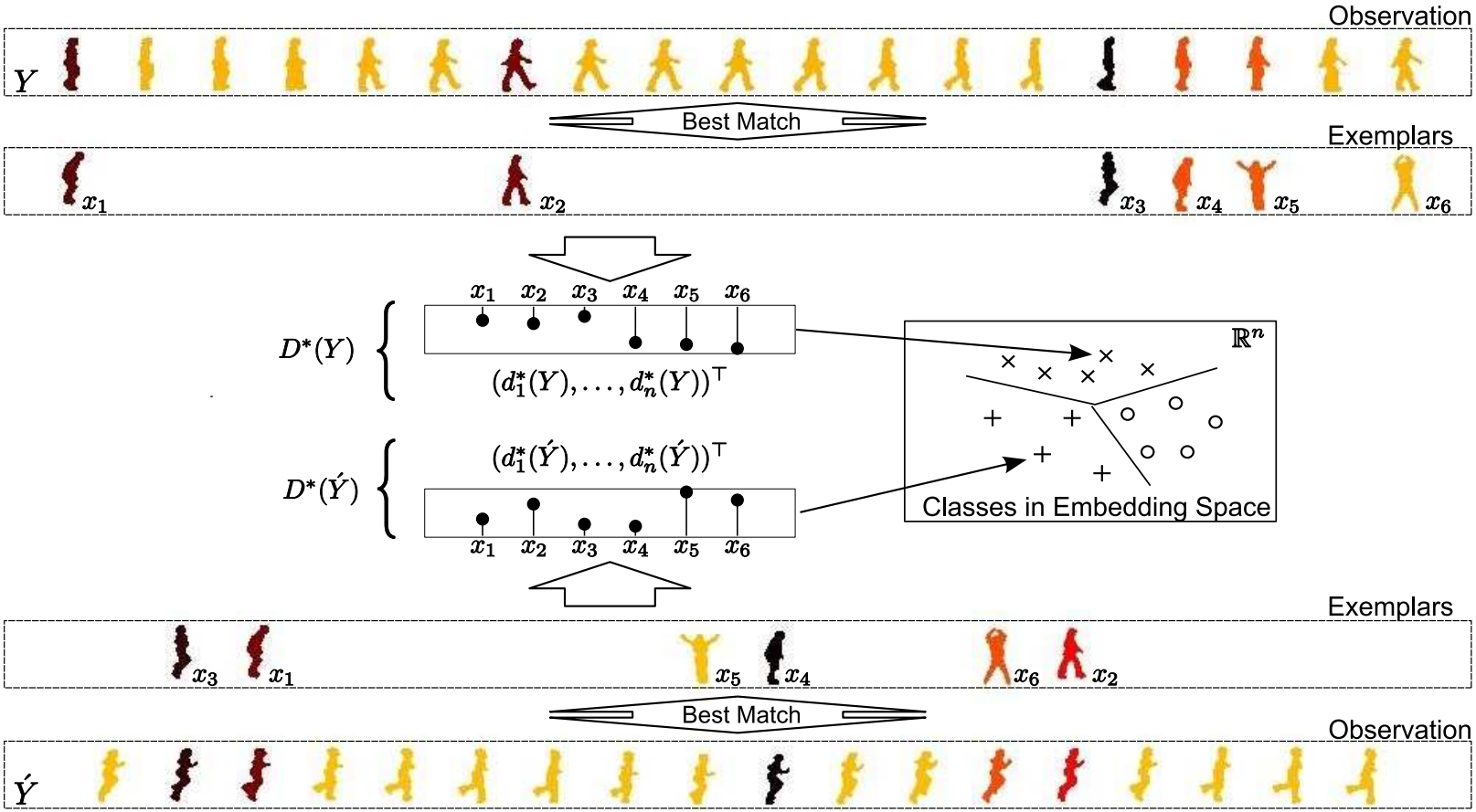


Figure 7.2.: Overview of the embedding method: Two action sequences Y (*walk*) and $\acute{Y}$ (*jump forward on one leg*) are matched against a set of silhouette exemplars x_i. For each exemplar the best matching frame in the sequence is identified (exemplar displayed on top of the corresponding frame; light colors correspond to high matching distances; dark colors to low matching distances). The resulting matching distances d_i^* form vector D^*, which is interpreted as an embedding of the sequences into a low dimensional space $\mathbb{R}^n$. The final classifier is learned over $\mathbb{R}^n$, where each point represents a complete sequence.

n exemplars. For each exemplar the minimum matching distance to any of the frames of the sequence is determined, and the resulting set of distances forms a vector D^* in the embedding space $\mathbb{R}^n$. The intuition we follow is that similar sequences will yield proximities to discriminative exemplars which are similar. Hence their point representation in $\mathbb{R}^n$ should be close. We thus model actions in $\mathbb{R}^n$ where both learning and recognition are performed. This is detailed in the following sections.

7.2.1. Exemplar-based Embedding

Our aim is to classify an action sequence $Y = y_1, \ldots, y_t$ over time with respect to the occurrence of known representative exemplars $X = \{x_1, \ldots, x_n\}$, e.g. silhouettes. The exemplar selection is presented in a further section (see Section 7.3) and we assume here that they are given.

We start by computing for each exemplar x_i the minimum distance to frames in the sequence:

$$d_i^*(Y) = \min_j d(x_i, y_j), \tag{7.1}$$

where d is a distance function between the primitives considered, as described in Section 7.2.3.

At this stage, distances could be thresholded and converted into binary detections, in the sense of a *key-frame* classifier [Carlsson and Sullivan, 2001]. This requires however thresholds to be chosen and furthermore does not allow to model uncertainties. Probabilistic exemplar-based approaches [Toyama and Blake, 2001] do model such uncertainties by converting distances into probabilities, but as mentioned earlier, at the price of complex computations. We instead simply work on the vectors that result from concatenating all the minimum distances

$$D^*(Y) = (d_1^*(Y), \ldots, d_n^*(Y))^\top \in \mathbb{R}^n, \tag{7.2}$$

without any probabilistic treatment. Note that our representation is similar in principle to the embedding described in [Athitsos and Sclaroff, 2003, Guo et al., 2007] in a static context. We extend it to temporal sequences.

7.2.2. Classifier

In the embedding space $\mathbb{R}^n$, classification of time sequences reduces to a simple operation which is to label the vectors $D^*(Y)$. A major advantage over traditional approaches is that such vectors encode complete sequences without the need for time normalizations or alignments. These vectors are points in $\mathbb{R}^n$ that are labelled using a standard Bayes classifier. Each class $c \in 1...C$ is represented through a single Gaussian distribution $p(D^*|c) = \mathcal{N}(D^*|\mu_c, \Sigma_c)$, which

we found adequate in experiments to model all important dependencies between exemplars. Assignments are determined through maximum a posteriori estimations:

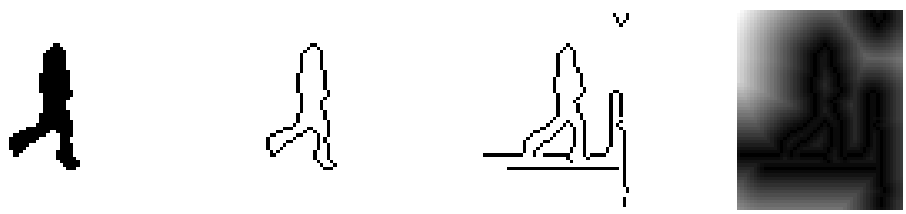
$$g(D^*) = \arg\max_c (D^*|c)p(c), \tag{7.3}$$

with $p(c)$ being the prior of class c that, without loss of generality, is assumed to be uniform.

Note that when estimating covariance Σ and depending on the dimension n, it is often the case that insufficient training data is available for Σ, and consequently the estimation may be non-invertible. We hence work with a regularized covariance of the form $\hat{\Sigma} = \Sigma + \epsilon I$, with I being the identity matrix and ϵ a small value.

7.2.3. Image Representation and Distance Functions

Actions are represented as vectors of distances from exemplars to the frames in the action's sequence. Such distances could be of several types, depending on the available information in the images, e.g. silhouettes or edges. In the following, we assume that silhouettes are available for the exemplars, which is a reasonable assumption in the learning phase, and we consider two situations for recognition. First, silhouettes, obtained for instance with background subtractions, are available; Second only edges can be considered.



(a) (b) (c) (d)

Figure 7.3.: Different types of image representations: (a) Silhouettes obtained using background subtraction. (b) and (c) Silhouette exemplar and filtered edge images used for matching when background subtraction can not be performed. A Chamfer matching is then achieved by correlating the silhouette (b) with the distance transformed edge image (d).

Silhouette-to-Silhouette Matching As in all previous parts of this thesis, we assume in this scenario that background subtracted sequences are available. Consequently, x and y are both represented through silhouettes, as illustrated in Figure 7.3(a). While difficult to obtain in many practical contexts, silhouettes, when available, provide rich and strong cues. Consequently they can be matched with a standard distance function and we choose the squared Euclidean distance $d(x, y) = |x - y|^2$, which is computed between the vector representations

of the binary silhouette images. Hence, the distance is simply the number of pixels with different values in both images.

Silhouette-to-Edge Matching In a more realistic scenario, background subtraction will not be possible due to moving or changing background as well as changing light, among other reasons. In that case, more advanced distances dealing with imperfect image segmentations must be considered. In our experiments, we use such a scenario where edge observations y (see Figure 7.3(c)), instead of silhouettes, are taken into account. In such observations, edges are usually spurious or missing. As mentioned earlier we assume that exemplars are represented through edge templates (see Figure 7.3(b)), computed using background subtraction in a learning phase. The distance we consider is then the Chamfer distance [Gavrila and Philomin, 1999]. This distance function has the advantage of being robust to clutter, since distances between the template edges, i.e. the exemplar, and their closest edges in the observed image are computed. In detail, the Chamfer distance measures the closest distance for each edge point on the observation x to any edge point in the exemplar y,

$$d(x, y) = \frac{1}{|x|} \sum_{f \in x} \mathsf{d}_y(f), \tag{7.4}$$

where $|x|$ is the number of edge points in x and $\mathsf{d}_y(f)$ is the distance between edge f and the closest edge-point in y. An efficient way to compute the Chamfer distance is by correlating the distance transformed observation (Figure 7.3(d)) with the exemplar silhouette (Figure 7.3(b)) .

In the above distance functions, we assume that we can locate the approximate person-centered region of interest (ROI) in an image, so that we do not have to scan over the whole image. Note also that the amount of clutter in a sequence can affect distances. A normalization where each vectors $D^*(Y)$ is translated and scaled with respect to its mean $\mu_D = \frac{1}{n}\sum_i^n d_i^*(Y)$ and standard deviation $\sigma_D = (\frac{1}{n}\sum_i^n (d_i^*(Y) - \mu_D)^2)^{1/2}$ can then be useful.

7.3. Key-Pose Selection

In the previous section, we assume that the exemplars, a set of discriminative primitives, are known. In practice we obtain them using a wrapper technique for feature selection [Kohavi and John, 1997], which was explained in detail in Section 5.5.2, and previously used for exemplar selection in Chapter 6.

In Figure 7.4 we show a sample exemplar set which we collected from the Weizmann-dataset [Blank et al., 2005] (the dataset is detailed in the next section). Figure 7.4 shows the average validation rate of all actions, and Figure 7.6 the average validation rates per action, which were

Figure 7.4.: A set of exemplar silhouettes and their original images as returned by the forward selection (from left to right). Validation rates computed during selection are shown in Figure 7.4 and 7.6.

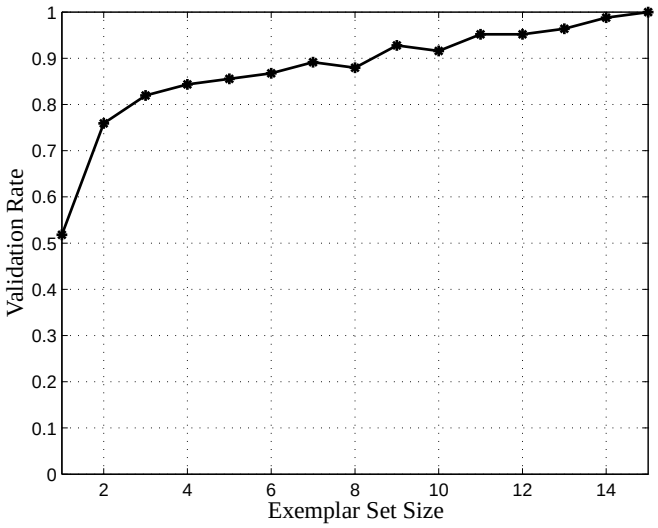


Figure 7.5.: Average validation rate during selection of exemplar set Figure 7.4.

computed on the training set during the selection. We observe that the selected poses naturally represent key-postures. Interestingly also, that even though the overall validation rate reaches 100% for 15 exemplars, not all classes are explicitly represented through an exemplar, indicating that exemplars are shared between actions.

7.4. Experiments

We have experimented our approach with the Weizmann-dataset [Blank et al., 2005] (see Figure 7.1 and 7.7) which has been recently used by several authors [Ali et al., 2007, Jhuang et al., 2007, Niebles and Fei-Fei, 2007, Scovanner et al., 2007, Wang and Suter, 2007]. It contains 10 actions: *bend (bend), jumping-jack (jack), jump-in-place (pjump), jump-forward (jump), run (run), gallop-sideways (side), jump-forward-one-leg (skip), walk (walk), wave one hand (wave1), wave two hands (wave2),* performed by 9 actors. Silhouettes extracted from backgrounds and original image sequences are provided.

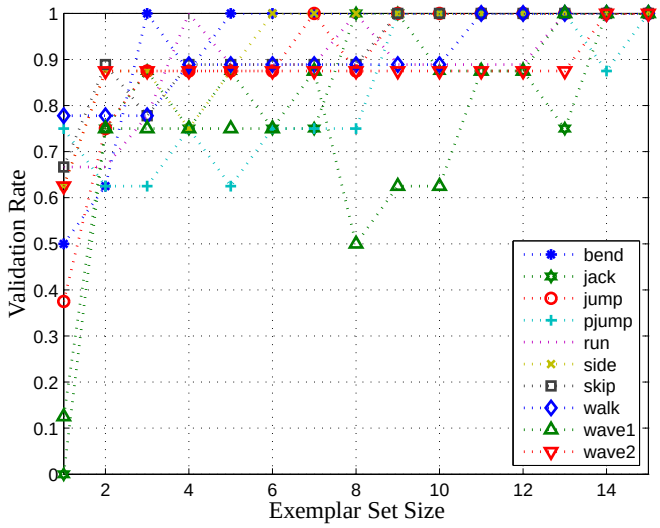


Figure 7.6.: Validation rate per action during selection of exemplar set Figure 7.4.

All recognition rates were computed with the leave-one-out cross-validation. Details are as follows. 8 out of the 9 actors in the database are used to train the classifier and select the exemplars, the 9th is used for the evaluation. This is repeated for all 9 actors and the rates are averaged. For the exemplar selection, we further need to divide the 8 training actors into training and validation sets. We do this as well with a leave-one-out cross-validation, using 7 training actors and the 8th as the validation set, then iterating over all possibilities. Exemplars are constantly selected from all 8 actors, but never from the 9th that is used for the evaluation. Also note that due to the small size of the training set, the validation rate can easily reach 100% if too many exemplars are considered. In this case, we randomly remove exemplars during the validation step, to reduce the validation rate and to allow new exemplars to be added. For testing we nevertheless use all selected exemplars.

7.4.1. Evaluation on Segmented Sequences

In these experiments, the background-subtracted silhouettes which are provided with the Weizmann-dataset were used to evaluate our method. For the exemplar selection, we first uniformly sub-sample the sequences by a factor $1/20$ and perform the selection on the remaining set of approximately 300 candidate frames. When we use all the 300 frames as exemplars, the recognition rate of our method is 100%.

To reduce the number of exemplars we search via forward selection over this set. The recognition rate on the test set, with respect to the number of exemplars is shown in Figure 7.8. Note that the forward selection includes one random step, in case that several exemplars have the

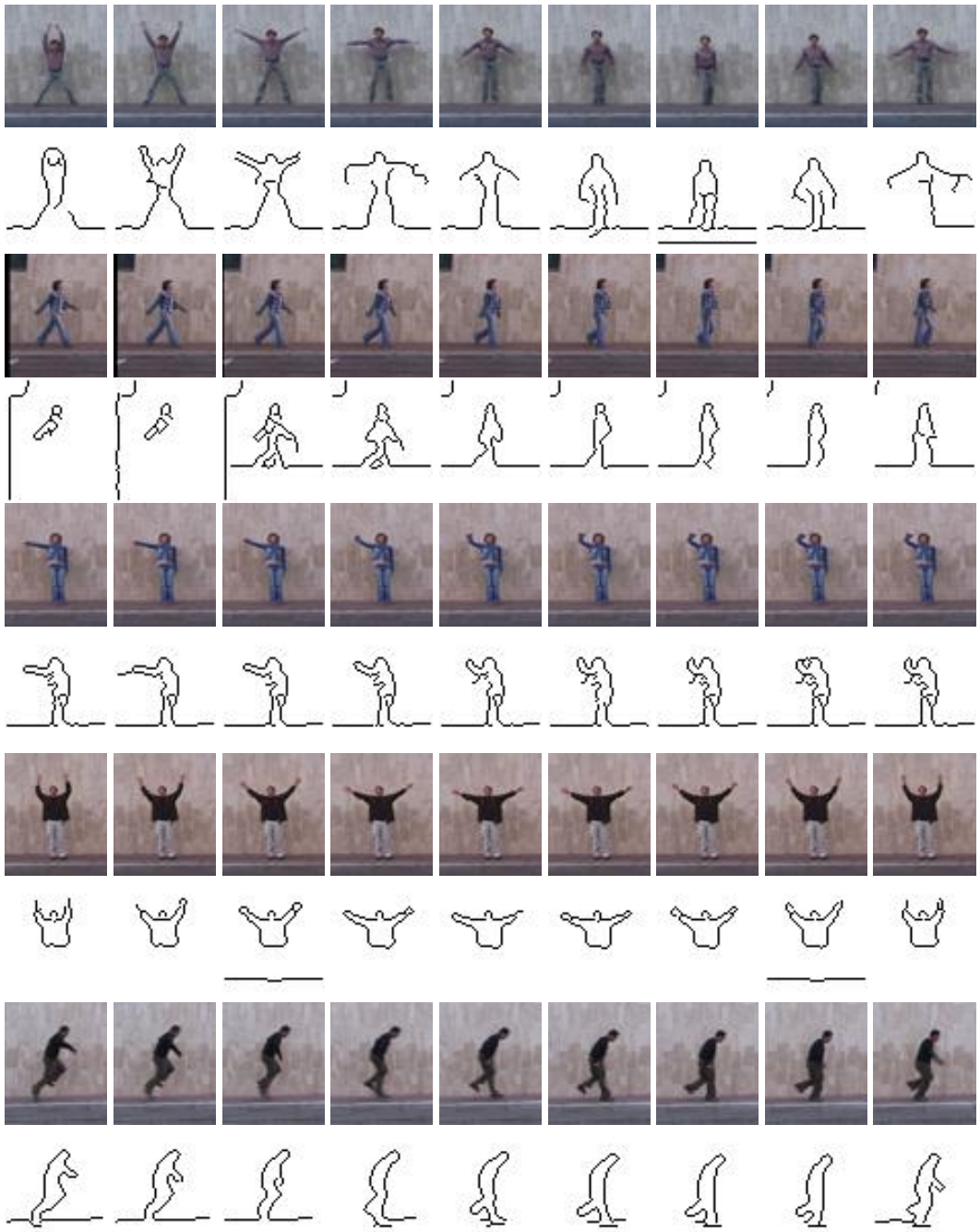


Figure 7.7.: Sample sequences and corresponding edge images. (Top to bottom) *jumping-jack, walk, wave one hand, wave two hands, jump-forward-one-leg*.

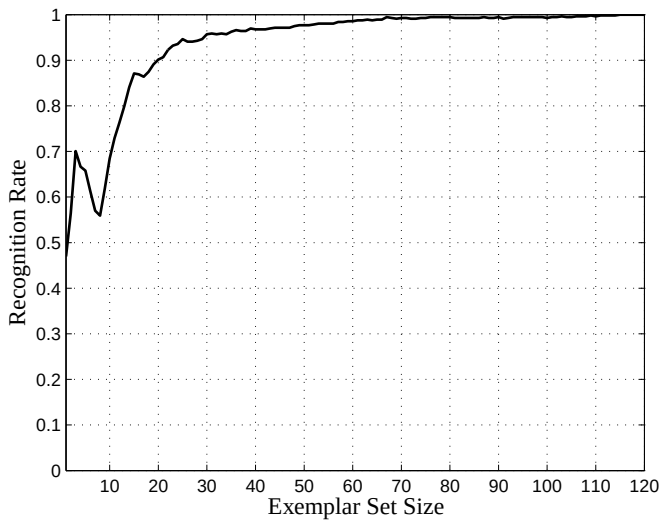


Figure 7.8.: Recognition rates vs. exemplar set size. Recognition is performed on background subtracted sequences.

same validation rate. We therefore repeat the experiment 10 times with all actors, and average over the results. In Figure 7.9, we show recognition rates for the individual classes. Note in particular the actions *jump-forward* and *jump-forward-one-leg* that are difficult to classify, because they are easily confused.

In summary, our approach can reach recognition rates up to 100% with approximately 120 exemplars. Moreover, with very small exemplar sets (e.g. around 20 exemplars), the average recognition rate on a dataset of 10 action and 9 actors is already higher than 90% and continuously increasing with additional exemplars (*e.g.* 97.7% for 50 exemplars). In comparison (see Table 7.1), the space-time volume approach proposed by Blank et al. [2005] had a recognition rate of 99.61%. Wang and Suter [2007] report a recognition rate of 97.78% with an approach that uses kernel-PCA for dimensional reduction and factorial conditional random fields to model motion dynamics. The work of Ali et al. [2007] uses a motion representation based on chaotic invariants and reports 92.6%. Note, however, that a precise comparison between the approaches is difficult, since experimental setups, *e.g.* number of actions and length of segments, slightly differ with each approach.

7.4.2. Evaluation on Cluttered Sequences

In this experiment, we used edge filtered sequences instead of background subtracted silhouettes. Edges are detected independently in each frame of the original sequences using a Canny edge detector. The resulting sequences contain a fair amount of clutter and missing edges, as can be seen in Figure 7.7. Exemplars are nevertheless represented through silhouettes since we

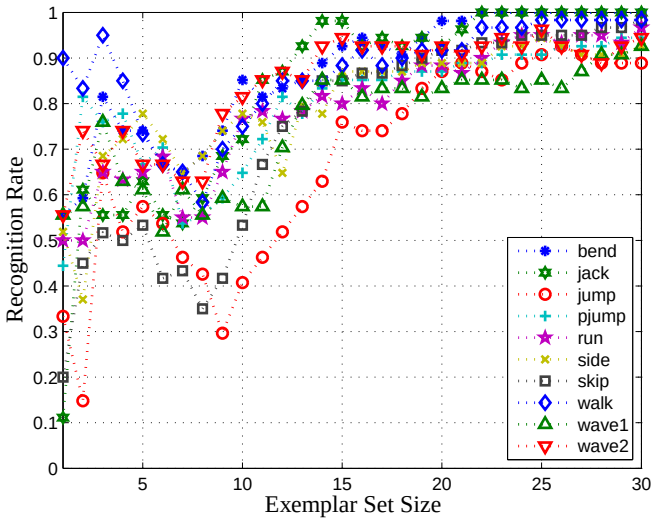


Figure 7.9.: Recognition rates per action vs. exemplar set size. Recognition is performed on background subtracted sequences.

Method	Recognition Rate (%)
Our Method	100.0
Blank et al. [2005]	99.6
Wang and Suter [2007]	97.8
Ali et al. [2007]	92.6

Table 7.1.: Results of approaches that use background subtraction.

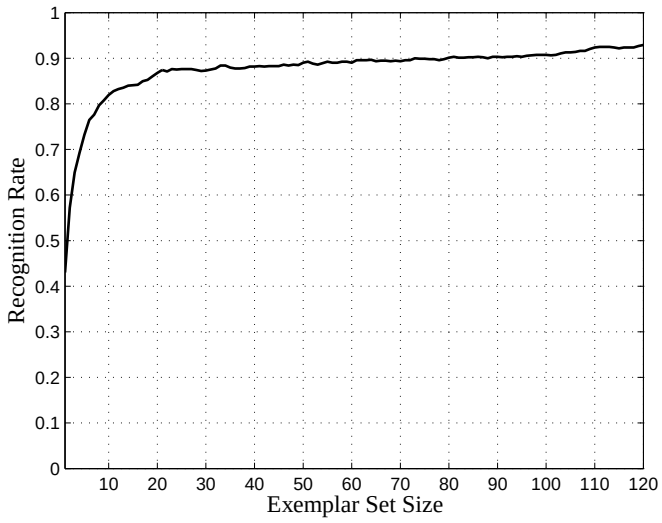


Figure 7.10.: Recognition rates vs. exemplar set size. Recognition is performed on non-background subtracted, edge filtered, sequences.

assume that background subtraction is available during the learning phase though not during recognition. We also assume that the person centered region of interest in the image can be located.

For a uniformly sub-sampled exemplar set of size 300, our method presents a recognition rate of 93.6% in cross-validation on all 10 actions and 9 actors. Similarly to the previous experiment, we compute the recognition rate with respect to the number of selected exemplars. Figure 7.10 shows the average recognition rate, and Figure 7.11 the rate per action.

We observe that after selection a recognition rate of 93% can be achieved with 110 exemplars. Figure 7.12 shows the resulting confusion matrix in that case.

As in the previous experiment, the two actions *jump-forward* and *jump-forward-one-leg* are difficult to classify, because they present many similarities. Another interesting observation is that, with only 2 exemplars, more than 50% of the actions are correctly classified.

In summary, our method shows very good results also on non-background subtracted sequences (up to 93.6% recognition rate). To our knowledge, methods that where tested on the Weizmann-dataset without using backgound subtraction are [Jhuang et al., 2007, Scovanner et al., 2007, Niebles and Fei-Fei, 2007], see Table 7.2. Jhuang et al. [2007] report up to 98.8% recognition rate with their biologically motivated system. These results are however computed from only 9 actions and without the *jump-forward-one-leg* action which leads in our case to 4 false recognitions out of a total of 6. Scovanner et al. [2007] mention 82.6% recognition rate using 3D SIFT descriptors and Niebles and Fei-Fei [2007] 72.8% using spatial-temporal features. As in previous experiments, experimental setups are slightly different with each approach, *e.g.*

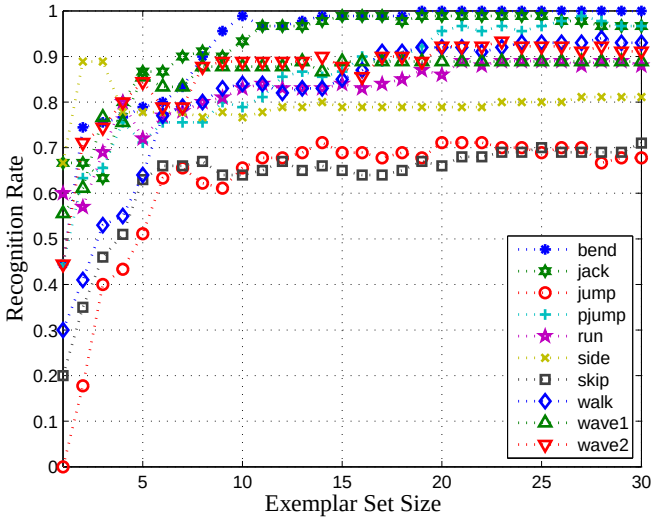


Figure 7.11.: Recognition rates per action vs. exemplar set size. Recognition is performed on non-background subtracted, edge filtered, sequences.

	bend	jack	jump	pjump	run	side	skip	walk	wave1	wave2
bend	9	0	0	0	0	0	0	0	0	0
jack	0	9	0	0	0	0	0	0	0	0
jump	0	0	6	1	0	0	2	0	0	0
pjump	0	0	0	9	0	0	0	0	0	0
run	0	0	0	0	9	0	0	0	0	0
side	0	0	0	1	0	8	0	0	0	0
skip	0	0	2	0	0	0	7	0	0	0
walk	0	0	0	0	0	0	0	9	0	0
wave1	0	0	0	0	0	0	0	0	9	0
wave2	0	0	0	0	0	0	0	0	0	9

Figure 7.12.: Confusion matrix for recognition on edge filtered sequences.

Method	Recognition Rate (%)
Jhuang et al. [2007]	96.3
Our Method	93.6
Scovanner et al. [2007]	82.6
Niebles and Fei-Fei [2007]	72.8

Table 7.2.: Results of approaches that use *no* background subtraction.

Niebles and Fei-Fei [2007] and Scovanner et al. [2007] additionally try to locate the person in the scene.

7.5. Conclusion and Discussion

We presented a new, compact, and highly efficient representation for action recognition. The representation is based on simple matching of exemplars to image sequences and does not account for dynamics. Based on exemplars, our representation supports advanced image matching distances and can be used with cluttered non-segmented sequences.

The experiments on sequences with and without background subtraction demonstrated that many actions can be recognized without taking dynamics into account. This was especially true on the publicly available Weizmann dataset, where our method has recognition rates which equal or exceed those of state-of-the-art approaches. To our opinion, this is an important result. However, it should be noticed that not all actions can be discriminated without dynamics. A typical example is an action and its reversal, *e.g.* as seen in Section 6.6.4: *sit-down* and *get-up*. Without taking temporal ordering into account, it will be very difficult to discriminate them. To recognize such actions, a modeling of dynamics is required, either coupled with the descriptor or on a higher level. Nevertheless, note also that, as demonstrated with many of the datasets currently used in the field that do not include such ambiguous actions, many recognition applications, *e.g.* visual surveillance, do not necessarily need to discriminate such particular cases. On the other hand, we strongly think, that approaches could be experimented with more realistic scenes to better evaluate their limitations.

For future work we plan to extend the approach of this chapter as well to view-independent action recognition. Further, temporal segmentation of sequences with such a representation needs to be investigated.

Part IV

Conclusion and Perspectives

CHAPTER 8

Conclusion

In the final chapter of this thesis, we conclude our work, recapitulate our contributions, and discuss further research directions and open issues.

8.1. Summary

In this thesis we explored different directions for action recognition. We proposed three new general frameworks. Major focus of our work was on view-independence, but we made as well contribution on other issues, such as learning of pose and motion primitives, modeling of action dynamics, and temporal segmentation of actions. We thereby emphasized on designing body model free representations, to avoid the difficulties inherent in MOCAP.

In our first approach we proposed a new view-invariant action representation, motion history volumes (Chapter 3), as 3D extension of motion history images [Davis, 2001]. The novelty of this representation, compared to existing view-invariant representations, is that neither a body model nor other intermediate cues, such as point correspondences, are required to compare actions over different views. Instead, the use of 3D information, which we computed from multiple calibrated and background subtracted cameras, was key to the success of our representation. Orientation invariant alignment and comparisons were then performed efficiently using Fourier transforms in cylindrical coordinates around the vertical axis. In experiments, MHVs demonstrated excellent recognition results, even using very simple machine learning techniques.

Based on MHVs we proposed a new method for automatic segmentation of continuous ac-

tions streams into primitive motions (Chapter 4). MHVs allowed us to efficiently compute average motion velocities in 3D. We used minima in velocity to define motion boundary. We used those boundaries for unsupervised segmentation and clustering of motion streams into classes of primitive motions. Again, our method resulted in segmentation and classification methods which were surprisingly efficient and robust. Moreover, in contrary to most existing work, our method is purely based on visual cues without the need for an intermediate body representation.

Using 3D information showed surprisingly efficient in our first approach. Computation of 3D information is, however, not always possible in practical scenarios. In our second approach (Chapter 6) we hence developed a framework for action recognition from arbitrary views, and in particular from a single view. Key to view independence was, nevertheless, that we modeled action in 3D. In a generative framework, 3D exemplar-based action models were then used to explain 2D image observations. In a probabilistic fashion, dynamics over exemplars and viewpoint were modeled as two independent Markov processes in an exemplar based HMM. View-independence was then achieved by marginalizing over all possible views. Hence our second approach achieves view-independence even from a single view, yet again without depending on additional body model information.

In our experiments we validated the fact that actions can be recognized from view arbitrary viewpoints, but results where not as good as when working solely in 3D, even when we fused observation likelihoods from multiple 2D view. Also we found that our space-time approach (MHVs), has slightly better results compared to the HMM-based state-transition model, even when both works were evaluated exactly on the same data in 3D. For the exemplar-based HMM, which uses a set of discriminatively selected exemplars, we were surprised to find that even a small number of exemplars are sufficient to models a variety of actions performed by different actors and with different viewpoints.

Those results, using small sets of discriminative key-pose exemplars, motivated our last approach. In Chapter 7 we purely focused on a bottom-up representation of observations through sets of discriminative exemplars. In particular, no further modeling in terms of motion dynamics was attempted. The resulting representation is hence in particular simple and efficient. In evaluation we validated the fact, that sets of characteristic static key poses are highly discriminant action descriptors. We established those result on a well known view-dependent dataset, where our approach has results that equal or exceed those of the current state of the art.

8.2. Conclusion

As a result of our work, we have found that the choice of initial representation is a very important aspect of action recognition. An issue which is thereby often overlooked is view-independence. Initially, our idea was to derive a ”view-invariant” action representation, *i.e.*

features that are independent with respect to the class of view transformations considered. As demonstrated in our MHV approach, we were able to derive such view-invariance by using 3D information extracted from multiple views. MHV is a good case in point, since it is a carefully-crafted representation which we have demonstrated to achieve excellent recognition results, even using very simple machine learning techniques.

When working with single view observations, the problem becomes more difficult, and we came to the conclusion that a single set of view-invariant features can not suffice to explain all possible views. Instead we proposed an approach that explicitly generates and searches over the space of possible 2D views. This was nevertheless realized using a learned underlying 3D action model. So the use of 3D information was as well key to the success of our second approach. We then demonstrated how such a search over arbitrary view and pose configurations can remain computation feasible while achieving good recognition results, by using an HMM framework that uses a small set of exemplary 3D key-poses, instead of a kinematic body model.

Elaborating further on the key-pose principle, we found that often solely the use of characteristic key-poses without any further temporal modeling is sufficient to discriminate actions. This finding was in contrary to our initial believe, that temporal modeling can always improve on a static action representation. Although there are evident cases where temporal modeling is indeed indispensable, *e.g.* for different actions that share the same set of poses, we conclude, that in many situation action recognition can benefit form a simplified representation, which is invariant to variations in time scale. This was demonstrated in our third approach, where we represented actions through a key-pose based embedding representation, and achieved excellent recognition results in comparison with other state of the art approaches.

8.3. Limitations, Future Work, and Open Issues

In the following we discuss limitations of our work and directions for future research.

Background Subtraction

As demonstrated in many works, including this thesis, silhouettes provide strong cues for action recognition. Robust extraction of silhouettes from realistic scenes is, however, an open problem, and existing methods for background substraction function only in constrained settings. Dependency on silhouettes therefore strongly limiting the application of many approaches.

In future work we want to investigate several directions to solve such issues. First, we want to investigate alternative image features. As an example, we replaced in our last approach (Chapter 7) silhouette exemplars through edge images and chamfer matchings. Alternatively, the integration of other representations in our frameworks can be investigated: optical flow [Efros et al.,

2003], Gabor filter banks [Jhuang et al., 2007], and space-time interest points [Laptev and Lindeberg, 2003, Dollar et al., 2005], for instance. Generally we think, an exhaustive evaluation of all these representations, on realistic datasets and under a unified framework, is an important contribution for future work, to truly discover advances and limitations of each representation.

Second, we want to investigate new techniques for background segmentation based on advanced foreground models. The background subtraction techniques currently used in action recognition are typically based on independent pixel-vise color statistics. We think that the use of prior models that model foreground not only as color, but instead as complex objects, *e.g.* as human bodies, can strongly improve background subtraction. For instance in image segmentation, Cremers et al. [2002] learn kernel-densities over shape exemplars to constrain segmentation with active contours. Zhao and Davis [2005] couple object detection and image segmentation using template hierarchies and chamfer matching. Bray et al. [2006] use human pose priors in graph cut based image segmentation. Ideally, we would like to extend such techniques over time, using actions as prior models.

Recognition of Coincidental Actions

Our methods can currently not recognize coincidental actions, *i.e.* several actions performed at the same time, such as waving hands while running for instance. In particular because we are using global templates, we can only recognize actions which are similar to templates in our database. Ideally, to recognize coincidental actions we need to identify individual body parts, which would be equivalent to using a body model. For instance in the work of Ikizler et al. [2007], activities are composed from actions individually recognized for each arm and each leg. Our intension for this thesis was, however, to avoid using a body model, because of the difficulties involved in estimating exact position of body parts. Alternatively, we want to investigate intermediate representations, which use local information, however not as detailed as body models, *i.e.* local templates or global templates factorized into local regions.

Group Actions and Interactions

Currently our methods assume that there is a single person in the scene, which we can locate. Detecting actions of multiple person and their interactions is a difficult problem. Although some interactions may be most easily described as single action, *e.g.* we could learn a global model of two persons shaking hands; more complex interactions need an individual modeling of persons, their actions, and the relations between those actions. For instance coupled latent models, *e.g.* [Oliver et al., 2000, Park and Aggarwal, 2003], have been used to model complex interaction. The success of such methods, however, strongly depend on the prepossessing steps which are necessary to identify individual people, to segment them from each other, and to

track them overt time; this possibly in crowded scenes and under occlusion. We think only with strong developments in all those fields, modelings of realistic actions and interaction will become practical.

Realistic Datasets and Applications

Including our dataset, there are currently three known and publicly available action recognition datasets (KTH [Schuldt et al., 2004] , Weizmann [Blank et al., 2005], IXMAS) used by state-of-the-art approaches for evaluation and comparison. Each of those datasets contains approximately 10 different action; those actions are mostly performed in controlled setting, without background motion, few clutter, and, with exception of our data, view-dependent. Evaluation on such data is unrealistic and very one-sided. This was for instance clear in our last approach, where we demonstrated on the view-dependent Weizmann dataset, that our very simplistic model can have results (up to 100% recognition rate) equally or better than much more ambitious approaches. Evaluation on such limited data makes thus comparison of approaches difficult, and does not help much to discover their true limitations. Moreover, there is a trend in current work to ignore important issues, which are not present in such artificial settings.

For the purpose of this thesis, our own IXMAS dataset proved very useful, pushing us forward to experiment with various representation frameworks, and allowing us to perform precise evaluation and comparison between them. For future work, we will, however, need more challenging data sets. Clearly, acquiring data is a time consuming complex tasks, but there is probably very little to be gained from over-simplified situations, *e.g.* where actions are limited to walking and running motions. Working on true surveillance footage, sport recordings, movies, and video data from the internet, will help us to discover the real requirements for action recognition, and it will help us to shift focus to other important issues involved in action recognition, such as previously discussed segmentation of continuous actions, dealing with unknown motions, composite actions, multiple persons, and view invariance, for instance. Only when we can handle all those issues, we will be able to deal with realistic scenes, such as those initially displayed in Figure 1.1. Until then, a lot of difficult and challenging work remains to be done in action recognition.

Bibliography

Aganj, E., Aganj, E., Pons, J.P., Segonne, F., and Keriven, R. (2007). Spatio-temporal shape from silhouette using four-dimensional delaunay meshing. In *IEEE International Conference on Computer Vision*, pages 1–8. 13, 28

Agarwal, A. and Triggs, B. (2006). Recovering 3d human pose from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1):44–58. 33

Aggarwal, J.K. and Cai, Q. (1999). Human motion analysis: a review. *Computer Vision and Image Understanding*, 73(3):428–440. 31

Aggarwal, J.K. and Park, S. (2004). Human motion: Modeling and recognition of actions and interactions. In *International Symposium on 3D Data Processing, Visualization and Transmission*, pages 640–647. 31

Ahmad, M. and Lee, S.W. (2006). Hmm-based human action recognition using multiview image sequences. In *International Conference on Pattern Recognition*, volume 1, pages 263–266. 50

Ali, S., Basharat, A., and Shah, M. (2007). Chaotic invariants for human action recognition. In *IEEE International Conference on Computer Vision*. 40, 132, 135, 136

Aloimonos, J. (1990). Purposive and qualitative active vision. In *International Conference on Pattern Recognition*, volume 1, pages 346–360 vol.1. 44

Alon, J., Athitsos, V., and Sclaroff, S. (2005). Accurate and efficient gesture spotting via pruning and subgesture reasoning. In *International Workshop on Human-Computer Interaction*, pages 189–198. 54, 57

Arikan, O., Forsyth, D.A., and O'Brien, J.F. (2003). Motion synthesis from annotations. *ACM Transactions on Graphics*, 22(3):402–408. 33, 55

Athitsos, V. and Sclaroff, S. (2003). Estimating 3d hand pose from a cluttered image. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 432–439. 103, 126, 129

Bellegarda, J.R. and Nahamoo, D. (1990). Tied mixture continuous parameter modeling for speech recognition. *Acoustics, Speech, and Signal Processing*, 38:2033–2045. 111

Belongie, S. and Malik, J. (2000). Matching with shape contexts. In *IEEE Workshop on Content-based Access of Image and Video Libraries*, pages 20–26. 47

Bissacco, A., Chiuso, A., Ma, Y., and Soatto, S. (2001). Recognition of human gaits. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages II–52–II–57 vol.2. 40

Blank, M., Gorelick, L., Shechtman, E., Irani, M., and Basri, R. (2005). Actions as space-time shapes. In *IEEE International Conference on Computer Vision*, pages 1395–1402. 6, 21, 35, 41, 42, 126, 127, 131, 132, 135, 136, 147

Bobick, A. and Davis, J. (1996a). An appearance-based representation of action. In *International Conference on Pattern Recognition*, page 307. 35

Bobick, A. and Davis, J. (1996b). Real-time recognition of activity using temporal templates. In *Workshop on Applications of Computer Vision*, pages 39–42. 14, 29, 35, 41, 46, 61, 63, 65, 106

Bobick, A.F. and Davis, J.W. (2001). The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):257–267. 42, 50, 63

Bobick, A.F. and Wilson, A.D. (1997). A state-based approach to the representation and recognition of gesture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(12):1325–1337. 40

Bobick, A. and Ivanov, Y. (1998). Action recognition using probabilistic parsing. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 196–202. 54

Bodor, R., Jackson, B., Masoud, O., and Papanikolopoulos, N. (2003). Image-based reconstruction for view-independent human motion recognition. In *International Conference on Intelligent Robots and Systems*, volume 2, pages 1548–1553 vol.2. 45

Boiman, O. and Irani, M. (2005). Detecting irregularities in images and in video. In *IEEE International Conference on Computer Vision*, volume 1, pages 462–469 Vol. 1. 39

Brand, M., Oliver, N., and Pentland, A. (1997). Coupled hidden markov models for complex action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 994–999. 33, 34, 40

Brand, M. (1999). Shadow puppetry. In *IEEE International Conference on Computer Vision*, pages 1237–1244. 40

Brand, M. and Kettnaker, V. (2000). Discovery and segmentation of activities in video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):844–851. 41, 54, 55, 56

Bray, M., Kohli, P., and Torr, P.H.S. (2006). Posecut: Simultaneous segmentation and 3d pose estimation of humans using dynamic graph-cuts. In *European Conference on Computer Vision*, pages 642–655. 146

Bregler, C. (1997). Learning and recognizing human dynamics in video sequences. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 568–574. 33, 40, 55

Campbell, L.W. and Bobick, A.F. (1995). Recognition of human body motion using phase space constraints. In *IEEE International Conference on Computer Vision*, pages 624–630. 33, 34

Campbell, L.W., Becker, D.A., Azarbayejani, A., Bobick, A.F., and Pentland, A. (1996). Invariant features for 3-d gesture recognition. In *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 157–163. 33, 47

Canton-Ferrer, C., Casas, J.R., and Pardàs, M. (2006). Human model and motion based 3d action recognition in multiple view scenarios (invited paper). In *European Signal Processing Conference*. ISBN: 0-387-34223-0. 49

Carlsson, S. and Sullivan, J. (2001). Action recognition by shape matching to key frames. In *Workshop on Models versus Exemplars in Computer Vision*. 43, 74, 126, 127, 129

Cedras, C. and Shah, M. (1994). A survey of motion analysis from moving light displays. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 214–221. 31

Cedras, C. and Shah, M. (1995). Motion-based recognition: A survey. *Image and Vision Computing*, 13(2):129–155. 31

Chen, Q., Defrise, M., and Deconinck, F. (1994). Symmetric phase-only matched filtering of fourier-mellin transforms for image registration and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(12):1156–1168. 66

Cohen, I. and Li, H. (2003). Inference of human postures by classification of 3d human body shape. In *IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, pages 74–81. 47, 49

Cremers, D., Kohlberger, T., and Schnörr, C. (2002). Nonlinear shape statistics in mumford-shah based segmentation. In *European Conference on Computer Vision*, pages 93–108. 146

Cutler, R. and Turk, M. (1998). View-based interpretation of real-time optical flow for gesture recognition. In *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 416–421. 37

Cutting, J. and Kozlowski, L. (1977). Recognizing friends by their walk: Gait perception without familiarity cues. *Bulletin Psychonomic Soc*, 9(5):353–356. 32

Cuzzolin, F., Sarti, A., and Tubaro, S. (2004). Action modeling with volumetric data. In *International Conference on Image Processing*, volume 2, pages 881–884 Vol.2. 45

Darrell, T. and Pentland, A. (1993). Space-time gestures. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 335–340. 37, 43, 54, 57

Davis, J.W. (2001). Hierarchical motion history images for recognizing human motion. In *IEEE Workshop on Detection and Recognition of Events in Video*, pages 39–46. 143

Deutscher, J., Blake, A., and Reid, I. (2000). Articulated body motion capture by annealed particle filtering. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 126–133 vol.2. 50, 106

Dollar, P., Rabaud, V., Cottrell, G., and Belongie, S. (2005). Behavior recognition via sparse spatio-temporal features. In *International Workshop on Performance Evaluation of Tracking and Surveillance*, pages 65–72. 37, 38, 44, 146

Efros, A.A., Berg, A., Mori, G., and Malik, J. (2003). Recognizing action at a distance. In *IEEE International Conference on Computer Vision*, pages 726–733. 36, 37, 145

Elgammal, A.M., Shet, V.D., Yacoob, Y., and Davis, L.S. (2003). Learning dynamics for exemplar-based gesture recognition . In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 571–578. 35, 100, 112

Felzenszwalb, P. and Huttenlocher, D. (2000). Efficient matching of pictorial structures. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 66–73 vol.2. 39

Feng, Z. and Cham, T.J. (2005). Video-based human action classi.cation with ambiguous correspondences. In *IEEE Conference on Computer Vision and Pattern Recognition*, page 82. 54

Fischler, M. and Elschlager, R. (1973). The representation and matching of pictorial structures. *IEEE Transactions on Computers*, 22(1):67–92. 39

Forstner, W. and Gulch, E. (1987). A fast operator for detection and precise location of distinct points, corners and centres of circular features. In *Intercommission Conference on Fast Processing of Photogrammetric Data*, pages 281–305. 37

Forsyth, D. (2006). Human motion tutorial: Activity recognition. CVPR 2006 Tutorial. 5, 20

Forsyth, D., Arikan, O., Ikemoto, L., O'Brien, J., and Ramanan, D. (2005). Computational studies of human motion: part 1, tracking and motion synthesis. *Found. Trends. Comput. Graph. Vis.*, 1(2-3):77–254. 31

Freund, Y. and Schapire, R.E. (1995). A decision-theoretic generalization of on-line learning and an application to boosting. In *European Conference on Computational Learning Theory*, pages 23–37. 103

Frey, B.J. and Jojic, N. (2000). Learning graphical models of images, videos and their spatial transformations. In *UAI*, pages 184–191. 50, 106, 109

Gavrila, D.M. and Davis, L.S. (1996). 3-d model-based tracking of humans in action: a multi-view approach. In *IEEE Conference on Computer Vision and Pattern Recognition*, page 73. 106

Gavrila, D.M. (1999). The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82–98. 31

Gavrila, D. and Davis, L. (1995). Towards 3-d model-based tracking and recognition of human movement. In *International Workshop on Face and Gesture Recognition*, pages 272–277. 33, 43, 46

Gavrila, D. and Philomin, V. (1999). Real-time object detection for smart vehicles. In *IEEE International Conference on Computer Vision*, pages 87–93. 35, 110, 126, 131

Ghahramani, Z. (1998). Learning dynamic Bayesian networks. *Lecture Notes in Computer Science*, 1387:168–197. 40

Goddard, N.H. (1989). The interpretation of visual motion: recognizing moving light displays. In *Workshop on Visual Motion*, pages 212 – 220. 32

Goddard, N.H. (1992). *The Perception of Articulated Motion: Recognizing Moving Light Displays*. Ph.D. thesis, University of Rochester, Rochester, NY, USA. 33

Grace, A.E. and Spann, M. (1991). A comparison between fourier-mellin descriptors and moment based features for invariant object recognition using neural networks. *Pattern Recognition Letters*, 12(10):635–643. 65

Granlund, G.H. (1972). Fourier preprocessing for hand print character recognition. *IEEE Trans. on Computers*, C–21(2):195–201. 47

Green, R.D. and Guan, L. (2004). Quantifying and recognizing human movement patterns from monocular video images-part i: a new framework for modeling human motion. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(2):179–190. 33, 40, 41, 54, 55

Gritai, A., Sheikh, Y., and Shah, M. (2004). On the use of anthropometry in the invariant analysis of human actions. In *Proc. 17th International Conference on Pattern Recognition ICPR 2004*, volume 2, pages 923–926 Vol.2. 46

Guerra-Filho, G. and Aloimonos, Y. (2007). A language for human action. *Computer*, 40(5):42–51. 55

Guo, Y., Xu, G., and Tsuji, S. (1994). Understanding human motion patterns. In *International Conference on Pattern Recognition*, volume 2, pages 325–329. 33, 34, 41

Guo, Y., Shan, Y., Sawhney, H., and Kumar, R. (2007). Peet: Prototype embedding and embedding transition for matching vehicles over disparate viewpoints. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. 126, 129

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182. 102, 103

Harris, C. and Stephens, M. (1988). A combined corner and edge detector. In *Alvey Conference*, pages 147–152. 37

Heesch, D. and Rueger, S.M. (2002). Combining features for content-based sketch retrieval - a comparative evaluation of retrieval performance. In *Proceedings of the 24th BCS-IRSG European Colloquium on IR Research*, pages 41–52. 65

Hogg, D. (1983). Model-based vision: A program to see a walking person. *Image and Vision Computing*, 1(1):5–20. 33

Hu, M.K. (1962). Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8:179–187. 41, 63, 65

Ikizler, N., , and Forsyth, D. (2007). Searching video for complex activities with finite state models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. 33, 41, 146

Ivanov, Y.A. and Bobick, A.F. (2000). Recognition of visual activities and interactions by stochastic parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):852–872. 40

Jenkins, O. and Mataric, M. (2002). Deriving action and behavior primitives from human motion data. In *International Conference on Intelligent Robots and System*, volume 3, pages 2551–2556 vol.3. 55

Jhuang, H., Serre, T., Wolf, L., and Poggio, T. (2007). A biologically inspired system for action. In *IEEE International Conference on Computer Vision*. 38, 44, 132, 137, 139, 146

Johansson, G. (1973). Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics*, 1414 (2):201–211. 32, 35, 43, 44, 125

John, G.H., Kohavi, R., and Pfleger, K. (1994). Irrelevant features and the subset selection problem. In *ICML*, pages 121–129. 102, 107

Jojic, N., Petrovic, N., Frey, B., and Huang, T. (2000). Transformed hidden markov models: Estimating mixture models of images and inferring spatial transformations in video sequences. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 26–33. 40, 50, 100

Junejo, I., Dexter, E., Laptev, I., and Pérez, P. (2008). Cross-view action recognition from temporal self-similarities. In *European Conference on Computer Vision*. 13, 28

Kahol, K., Tripathi, P., Panchanathan, S., and Rikakis, T. (2003). Gesture segmentation in complex motion sequences. In *International Conference on Image Processing*, volume 2, pages II–105–8 vol.3. 53

Kazhdan, M. (2004). *Shape Representations and Algorithms for 3D Model Retrieval*. Ph.D. thesis, Princeton University. 44

Kazhdan, M., Funkhouser, T., and Rusinkiewicz, S. (2003). Rotation invariant spherical harmonic representation of 3d shape descriptors. In *Eurographics Symposium on Geometry Processing*. 47, 66

Ke, Y., Sukthankar, R., and Hebert, M. (2005). Efficient visual event detection using volumetric features. In *IEEE International Conference on Computer Vision*, volume 1, pages 166–173. 41, 54, 57

Ke, Y., Sukthankar, R., and Hebert1, M. (2007). Event detection in crowded videos. In *IEEE International Conference on Computer Vision*. 39, 54, 57

Knossow, D., Ronfard, R., and Horaud, R.P. (2008). Human motion tracking with a kinematic parameterization of extremal contours. *International Journal of Computer Vision*, 79(2):247–269. 106

Kohavi, R. and John, G.H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324. 14, 29, 101, 126, 131

Kojima, A., Tamura, T., and Fukunaga, K. (2002). Natural language description of human activities from video images based on concept hierarchy of actions. *International Journal of Computer Vision*, 50(2):171–184. 40, 79

Kozlowski, L. and Cutting, J. (1977). Recognizing the sex of a walker from a dynamic point-light display. *Perception & Psychophysics*, 21(6):575–580. 32

Laptev, I. and Lindeberg, T. (2003). Space-time interest points. In *IEEE International Conference on Computer Vision*, pages 432–439 vol.1. 37, 38, 146

Laptev, I. and Pérez, P. (2007). Retrieving actions in movies. In *IEEE International Conference on Computer Vision*. 41

Laurentini, A. (1994). The Visual Hull Concept for silhouette-based Image Understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):150–162. 65

Lee, H.K. and Kim, J. (1999). An hmm-based threshold model approach for gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(10):961–973. 56, 58

Li, L.J. and Fei-Fei, L. (2007). What, where and who? classifying events by scene and object recognition. In *IEEE International Conference on Computer Vision*, pages 1–8. 43

Liu, J. and Shah, M. (2008). Learning human actions via information maximization. In *IEEE Conference on Computer Vision and Pattern Recognition*. 13, 28

Lo, C. and Don, H. (1989). 3-d moment forms: Their construction and application to object identification and positioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(10):1053–1064. 49, 65

Lowe, D.G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110. 37, 38

Lv, F. and Nevatia, R. (2007). Single view human action recognition using key pose matching and viterbi path searching. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. 13, 28, 35, 50, 110, 111

Lv, F. and Nevatia, R. (2006). Recognition and segmentation of 3-d human action using hmm and multi-class adaboost. In *European Conference on Computer Vision*, pages 359–372. 40, 54

Marr, D. and Nishihara, H.K. (1978). Representation and recognition of the spatial organization of three-dimensional shapes. *Philosophical Transactions of the Royal Society of London B*, 200(1140):269–294. 33

Marr, D. and Vaina, L. (1982). Representation and recognition of the movements of shapes. *Philosophical Transactions of the Royal Society of London B*, 214:501–524. 34, 53, 81

Masoud, O. and Papanikolopoulos, N. (2003). A method for human action recognition. *Image and Vision Computing*, 21(8):729–743. 35

McCloud, S. (1993). *Understanding Comics: The Invisible Art*. Kitchen Sink Press. 106

Meng, H., Pears, N., and Bailey, C. (2007). A human action recognition system for embedded computer vision application. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–6. 35, 41

Minka, T. (2004). Exemplar-based likelihoods using the pdf projection theorem. Technical report, Microsoft Research. 100

Moeslund, T.B. and Granum, E. (2001). A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding*, 81(3):231–268. 31

Moeslund, T.B., Hilton, A., and Krûger, V. (2006). A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2):90–126. 31

Morency, L.P., Quattoni, A., and Darrell, T. (2007). Latent-dynamic discriminative models for continuous gesture recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. 40, 54

Morguet, P. and Lang, M. (1998). Spotting dynamic hand gestures in video image sequences using hidden markov models. In *International Conference on Image Processing*, pages 193–197 vol.3. 54, 57

Nguyen, N., Phung, D., Venkatesh, S., and Bui, H. (2005). Learning and detecting activities from movement trajectories using the hierarchical hidden markov model. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 955–960 vol. 2. 40

Niebles, J., Wang, H., Wang, H., and Fei Fei, L. (2006). Unsupervised learning of human action categories using spatial-temporal words. In *British Machine Vision Conference*, page III:1249. 38

Niebles, J.C. and Fei-Fei, L. (2007). A hierarchical model of shape and appearance for human action classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. 39, 132, 137, 139

Niyogi, S. and Adelson, E. (1994). Analyzing and recognizing walking figures in xyt. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 469–474. 33, 43

Nowozin, S., Bakir, G., and Tsuda, K. (2007). Discriminative subsequence mining for action classification. In *IEEE International Conference on Computer Vision*. 38, 41, 44

Ogale, A., Karapurkar, A., Guerra-Filho, G., and Aloimonos, Y. (2004). View-invariant identification of pose sequences for action recognition. In *VACE*. 35, 40, 50, 53, 111

Ogale, A.S., Karapurkar, A., and Aloimonos, Y. (2005). View-invariant modeling and recognition of human actions using grammars. In *Workshop on Dynamical Vision*, pages 115–126. 40, 50, 79

Oliver, N.M., Rosario, B., and Pentland, A. (2000). A bayesian computer vision system for modeling human interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):831–843. 146

Otterloo, P.J.V. (1991). *A contour-oriented approach to shape analysis*. Prentice Hall International (UK) Ltd., Hertfordshire, UK, UK. 47

Parameswaran, V. and Chellappa, R. (2003). View invariants for human action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages II–613–19 vol.2. 47, 48

Parameswaran, V. and Chellappa, R. (2005). Human action-recognition using mutual invariants. *Computer Vision and Image Understanding*, 98(2):295–325. 47

Parameswaran, V. and Chellappa, R. (2006). View invariance for human action recognition. *International Journal of Computer Vision*, 66(1):83–101. 47

Park, S. and Aggarwal, J.K. (2003). Recognition of two-person interactions using a hierarchical bayesian network. In *ACM SIGMM International Workshop on Video Surveillance*, pages 65–76. 40, 146

Peursum, P., Bui, H., Venkatesh, S., and West, G. (2004). Human action segmentation via controlled use of missing data in hmms. In *International Conference on Pattern Recognition*, volume 4, pages 440–445 Vol.4. 41, 54

Peursum, P., West, G., and Venkatesh, S. (2005). Combining image regions and human activity for indirect object recognition in indoor wide-angle views. In *IEEE International Conference on Computer Vision*, volume 1, pages 82–89 Vol. 1. 40

Peursum, P., Venkatesh, S., and West, G. (2007). Tracking-as-recognition for articulated full-body human motion analysis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. 33, 46, 50, 51, 106

Pierobon, M., Marcon, M., Sarti, A., and Tubaro, S. (2006). 3-d body posture tracking for human action template matching. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages II–II. 47

Polana, R. and Nelson, R. (1993). Detecting activities. In *Proc. CVPR '93. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2–7. 41

Polana, R. and Nelson, R. (1994). Low level recognition of human motion (or how to get your man without finding his body parts). In *NAM*. 36, 37

Polana, R. and Nelson, R. (1992). Recognition of motion from temporal texture. In *Proc. CVPR '92. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 129–134. 37

Poppe, R. and Poel, M. (2006). Comparison of silhouette shape descriptors for example-based human pose recovery. In *Proceedings of the IEEE Conference on Automatic Face and Gesture Recognition 2006 (FG 2006)*, pages 541–546. ISBN=0-7695-2503-2. 65

Rabiner, L.R. (1990). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:267–296. 40, 54, 93, 95, 96, 97, 112

Ramanan, D. and Forsyth, D.A. (2003). Automatic annotation of everyday movements. Technical Report UCB/CSD-03-1262, EECS Department, University of California, Berkeley. 33, 34

Rao, C., Gritai, A., Shah, M., and Syeda-Mahmood, T. (2003a). View-invariant alignment and matching of video sequences. In *IEEE International Conference on Computer Vision*, pages 939–945 vol.2. 47

Rao, C., Shah, M., and Syeda-Mahmood, T. (2003b). Invariance in motion analysis of videos. In *ACM International conference on Multimedia*, pages 518–527. 47, 48

Rao, C., Yilmaz, A., and Shah, M. (2002). View-invariant representation and recognition of actions. *International Journal of Computer Vision*, 50(2):203–226. 47, 53

Rittscher, J. and Blake, A. (1999). Classification of human body motion. In *IEEE International Conference on Computer Vision*, pages 634–639. 35, 36, 40, 54

Robertson, N. and Reid, I. (2005). Behaviour understanding in video: A combined method. In *IEEE International Conference on Computer Vision*, pages 808–815. 37

Rogez, G., Guerrero, J., Martinez del Rincon, J., and Orrite Urunuela, C. (2006). Viewpoint independent human motion analysis in man-made environments. In *British Machine Vision Conference*, page II:659. 45

Roh, M.C., Shin, H.K., Lee, S.W., and Lee, S.W. (2006). Volume motion template for view-invariant gesture recognition. In *International Conference on Pattern Recognition*, volume 2, pages 1229–1232. 46

Rohlicek, J., Russell, W., Roukos, S., and Gish, H. (1989). Continuous hidden markov modeling for speaker-independent word spotting. In *International Conference on Acoustics, Speech, and Signal Processing*, pages 627–630 vol.1. 58

Rohr, K. (1994). Towards model-based recognition of human movements in image sequences. *Graphical Model and Image Processing*, 59(1):94–115. 33

Rosales, R. and Sclaroff, S. (2000). Inferring body pose without tracking body parts. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 721–727 vol.2. 33

Rose, C., Cohen, M., and Bodenheimer, B. (1998). Verbs and adverbs: multidimensional motion interpolation. *IEEE Computer Graphics and Applications*, 18(5):32–40. 55

Rose, R. and Paul, D. (1990). A hidden markov model based keyword recognition system. In *International Conference on Acoustics, Speech, and Signal Processing*, pages 129–132 vol.1. 58

Rubin, J.M. and Richards, W.A. (1985). Boundaries of visual motion. Technical report, Massachusetts Institute of Technology, Cambridge, MA, USA. 53, 81

Rui, Y. and Anandan, P. (2000). Segmenting visual actions based on spatio-temporal motion patterns. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1111–1118. 53, 81

Sadjadi, F. and Hall, E. (1980). Three-dimensional moment invariants. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(2):127–136. 65

Sakoe, H. and Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(1):43–49. 43

Schmid, C., Mohr, R., and Bauckhage, C. (2000). Evaluation of interest point detectors. *International Journal of Computer Vision*, 37(2):151–172. 37

Schuldt, C., Laptev, I., and Caputo, B. (2004). Recognizing human actions: A local svm approach. In *ICPR*, pages 32–36. 6, 21, 38, 44, 147

Scovanner, P., Ali, S., and Shah, M. (2007). A 3-dimensional sift descriptor and its application to action recognition. In *ACM International conference on Multimedia*, pages 357–360. 38, 44, 132, 137, 139

Seitz, S.M. and Dyer, C.R. (1997). View-invariant analysis of cyclic motion. *International Journal of Computer Vision*, 25(3):231–251. 47

Sheikh, M. and Shah, M. (2005). Exploring the space of a human action. In *IEEE International Conference on Computer Vision*, volume 1, pages 144–149. 46

Shen, D. and Ip, H.H.S. (1999). Discriminative wavelet shape descriptors for recognition of 2-d patterns. *Pattern Recognition*, 32(2):151–165. 65

Sidenbladh, H., Black, M.J., and Fleet, D.J. (2000). Stochastic tracking of 3d human figures using 2d image motion. In *European Conference on Computer Vision*, pages 702–718. 50, 51, 106

Sminchisescu, C., Kanaujia, A., Li, Z., and Metaxas, D. (2005). Conditional models for contextual human motion recognition. In *IEEE International Conference on Computer Vision*, volume 2, pages 1808–1815 Vol. 2. 40, 54

Sminchisescu, C. and Triggs, B. (2001). Covariance scaled sampling for monocular 3d body tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii, USA*, volume 1, pages 447–454. 33

Smith, P., da Vitoria Lobo, N., and Shah, M. (2005). Temporalboost for event recognition. In *IEEE International Conference on Computer Vision*, volume 1, pages 733–740 Vol. 1. 41

Starner, T. and Pentland, A. (1995). Real-time american sign language recognition from video using hidden markov models. In *International Symposium on Computer Vision*, pages 265–270. 33, 40

Sumi, S. (1984). Upside-down presentation of the johansson moving light-spot pattern. *Perception*, 13(3):283–286. 32

Swets, D.L. and Weng, J. (1996). Using discriminant eigenfeatures for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):831–836. 72

Syeda-Mahmood, T., Vasilescu, M., and Sethi, S. (2001). Recognizing action events from multiple viewpoints. In *EventVideo01*, pages 64–72. 46

Tenenbaum, J.B., Silva, V., and Langford, J.C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323. 55

Thurau, C. (2007). Behavior histograms for action recognition and human detection. In *Workshop on HUMAN MOTION Understanding, Modeling, Capture and Animation*, pages 299–312. 40

Tomasi, C. and Kanade, T. (1992). Shape and motion from image streams under orthography: a factorization method. *Int. J. Comput. Vision*, 9(2):137–154. 46

Toyama, K. and Blake, A. (2001). Probabilistic tracking in a metric space. In *IEEE International Conference on Computer Vision*, pages 50–59. 35, 40, 50, 100, 103, 106, 109, 110, 111, 129

Vapnik, V. (1998). *Statistical Learning Theory*. Wiley. 103

Vasilescu, M. (2002). Human motion signatures: analysis, synthesis, recognition. In *Proc. 16th International Conference on Pattern Recognition*, volume 3, pages 456–460 vol.3. 55

Veeraraghavan, A., Chellappa, R., and Roy-Chowdhury, A. (2006). The function space of an activity. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 959–968. 43

Vitaladevuni, S., Kellokumpu, V., and Davis, L. (2008). Action recognition using ballistic dynamics. In *IEEE Conference on Computer Vision and Pattern Recognition*, page 8 p. 13, 28

Wang, L. and Suter, D. (2007). Recognizing human activities from silhouettes: Motion subspace and factorial discriminative graphical model. In *IEEE Conference on Computer Vision and Pattern Recognition*. 35, 36, 40, 126, 132, 135, 136

Wang, S.B., Quattoni, A., Morency, L.P., Demirdjian, D., and Darrell, T. (2006a). Hidden conditional random fields for gesture recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1521–1527. 40

Wang, T.S., Shum, H.Y., Xu, Y.Q., and Zheng, N.N. (2001). Unsupervised analysis of human gestures. In *IEEE Pacific Rim Conference on Multimedia*, pages 174–181. 40, 53, 55

Wang, Y., Jiang, H., Drew, M., Li, Z.N., and Mori, G. (2006b). Unsupervised discovery of action classes. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1654–1661. 44

Wang, Y., Sabzmeydani, P., and Mori, G. (2007). Semi-latent dirichlet allocation: A hierarchical model for human action recognition. In *Workshop on HUMAN MOTION Understanding, Modeling, Capture and Animation*. 37, 44

Webb, A.R. (2002). *Statistical Pattern Recognition, 2nd Edition*. John Wiley & Sons. 72

Weinland, D. and Boyer, E. (2008). Action recognition using exemplar-based embedding. In *IEEE Conference on Computer Vision and Pattern Recognition*. 15, 29

Weinland, D., Boyer, E., and Ronfard, R. (2007). Action recognition from arbitrary views using 3d exemplars. In *IEEE International Conference on Computer Vision*. 15, 29, 102

Weinland, D., Ronfard, R., and Boyer, E. (2005). Motion history volumes for free viewpoint action recognition. In *IEEE International Workshop on modeling People and Human Interaction*. 14, 29, 68, 69

Weinland, D., Ronfard, R., and Boyer, E. (2006a). Automatic discovery of action taxonomies from multiple views. In *IEEE Conference on Computer Vision and Pattern Recognition*. 14, 29

Weinland, D., Ronfard, R., and Boyer, E. (2006b). Free viewpoint action recognition using motion history volumes. *Computer Vision and Image Understanding*, 104(2-3):249–257. 14, 29

Wilson, A. and Bobick, A. (1995). Learning visual behavior for gesture analysis. In *International Symposium on Computer Vision*, pages 229–234. 40

Wilson, A.D. and Bobick, A.F. (1999). Parametric hidden markov models for gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9):884–900. 33, 54

Wong, S.F., Kim, T.K., and Cipolla, R. (2007). Learning motion categories using both semantic and structural information. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–6. 39

Yacoob, Y. and Black, M. (1998). Parameterized modeling and recognition of activities. In *IEEE International Conference on Computer Vision*, pages 120–127. 33

Yamato, J., Ohya, J., and Ishii, K. (1992). Recognizing human action in time-sequential images using hidden markov model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 379–385. 35, 36, 40

Yan, P., Khan, S.M., and Shah, M. (2008). Learning 4d action feature models for arbitrary view action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*. 13, 28

Yang, M.H. and Ahuja, N. (1999). Recognizing hand gesture using motion trajectories. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages –472 Vol. 1. 40

Yilmaz, A. and Shah, M. (2005a). Actions sketch: A novel action representation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages I: 984–989. 35, 41, 46, 47

Yilmaz, A. and Shah, M. (2005b). Recognizing human actions in videos acquired by uncalibrated moving cameras. In *IEEE International Conference on Computer Vision*, pages 150–157. 34, 46

Zahn, C.T. and Roskies, R.Z. (1972). Fourier descriptors for plane closed curves. *Transactions on Computers, IEEE*, c-21(3):269–281. 47

Zelnik-Manor, L. and Irani, M. (2001). Event-based video analysis. In *IEEE Conference on Computer Vision and Pattern Recognition*. 37, 46, 54, 55

Zhang, J. and Zhuang, Y. (2007). View-independent human action recognition by action hypersphere in nonlinear subspace. In *IEEE Pacific Rim Conference on Multimedia*, pages 108–117. 13, 28

Zhao, L. and Davis, L. (2005). Closely coupled object detection and segmentation. In *IEEE International Conference on Computer Vision*, volume 1, pages 454–461 Vol. 1. 146

Zhao, T. and Nevatia, R. (2002). 3d tracking of human locomotion: a tracking as recognition approach. In *International Conference on Pattern Recognition*, volume 1, pages 546–551. 33, 46

Zhong, H., Shi, J., and Visontai, M. (2004). Detecting unusual activity in video. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages II–819–II–826 Vol.2. 54