# ABSTRACT

Title of Dissertation: VIEW-INVARIANCE IN VISUAL HUMAN MOTION ANALYSIS

Vasudev Parameswaran, Doctor of Philosophy, 2004

Dissertation directed by: Professor Rama Chellappa
Department of Computer Science

This thesis makes contributions towards the solutions to two problems in the area of visual human motion analysis: human action recognition and human body pose estimation. Although there has been a substantial amount of research addressing these two problems in the past, the important issue of viewpoint invariance in the representation and recognition of poses and actions has received relatively scarce attention, and forms a key goal of this thesis.

Drawing on results from 2D projective invariance theory and 3D mutual invariants, we present three different approaches of varying degrees of generality, for human action representation and recognition. A detailed analysis of the approaches reveals key challenges, which are circumvented by enforcing spatial and temporal coherency constraints. An extensive performance evaluation of the approaches on 2D projections of motion capture data and manually segmented real image sequences demonstrates that in addition to viewpoint changes, the approaches are able to handle well, varying speeds of execution of actions (and hence different frame rates of the video), different subjects and minor variabilities in the spatiotemporal dynamics of the action.

Next, we present a method for recovering the body-centric coordinates of key joints and parts of a canonically scaled human body, given an image of the body and the point correspondences of specific body joints in an image. This problem is difficult to solve because of body articulation and perspective effects. To make the problem tractable, previous researchers have resorted to restricting the camera model or requiring an unrealistic number of point correspondences, both of which are more restrictive than necessary. We present a solution for the general case of a perspective uncalibrated camera. Our method requires that the torso does not twist considerably, an assumption that is usually satisfied for many poses of the body. We evaluate the quantitative performance of the method on synthetic data and the qualitative performance of the method on real images taken with unknown cameras and viewpoints. Both these evaluations show the effectiveness of the method at recovering the pose of the human body.

# VIEW-INVARIANCE IN VISUAL HUMAN MOTION ANALYSIS

by

Vasudev Parameswaran

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland at College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2004

Advisory Committee:

Professor Rama  Chellappa, Chairman/Advisor
Professor Larry  Davis
Associate Professor Amitabh Varshney
Associate Professor David Jacobs
Professor Shihab Shamma

# DEDICATION

To my parents, and to Priya, Vani and Surya

# ACKNOWLEDGEMENTS

A number of caring people deserve a lot of credit for helping me get through the long process of finishing my dissertation. Firstly, I thank my advisor, Dr. Rama Chellappa for his support throughout the process and for providing helpful advice and guidance during times of need. I thank him for giving me the opportunity to work with him, for introducing me to the field of computer vision, and for guiding me towards the problem of human action recognition. Thanks to my parents and sister for their unwavering support and encouragement during the dissertation and throughout my life in general, and for their many sacrifices so that I could have a better life. It is impossible to list the number of ways in which my wife Priya supported me through the dissertation. Suffice it to say that I am forever indebted to her for all of her support. Those that know me, know that I stutter. There was a time when severe stuttering got the better of me and made me fear and avoid any kind of speaking beyond the bare minimum required. When I started the dissertation, I worried about whether or not

I would be able to speak for an extended period of time in front of an audience. The days of fear and avoidance of speaking are long gone, thanks to Vivian Sisskin of the department of hearing and speech sciences at the University of Maryland. I thank her for introducing me to the right literature on stuttering and for encouraging me to face the fear and develop a problem solving approach to my stuttering. Like computer vision, stuttering is deceptively simple, eludes a general solution, and will continue to confound us for the foreseeable future. However, continuing the analogy, that need not stop us from studying it as much as we can, or from trying to engineer approximate and probabilistic solutions for it in the short term, or from seeking a breakthrough in the long term.

Finally, thanks to Dr. Larry Davis, Dr. David Jacobs, Dr. Amitabh Varshney and Dr. Shihab Shamma for showing interest in my work and for agreeing to be members of the examining committee.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# Chapter 1

# Introduction

*Visual human motion analysis* consists of problems that fall within the realms of computer vision and computer animation (or graphics). In computer vision the central problem is the recovery of human motion from image sequences while in computer animation, the central problem is the rendering of realistic looking human motion to form image sequences. This thesis deals with problems in the computer vision realm. 'Recovery of human motion' from image sequences may mean recovery of the 3D coordinates of the body joints (referred to as *visual human motion capture*) or a high-level description of the type of action that the human is performing (referred to as *visual human action recognition*). In this thesis, the words *visual* and *human* will be omitted and implied when the terms 'motion analysis', 'motion capture' and 'action recognition' are used. While interesting in itself, the large number of potential applications that could benefit from solutions to these problems, partly explains the explosion of interest in the area over the last decade. Papers surveying the state of the art in the area ([74], [46], [25] [1], [13]) appear every two to three years and provide good summaries of incremental developments in the field while serving as starting points for further reading.

In the following sections, we create a context for the thesis: we discuss major applications of human motion recovery from image sequences, describe specific problems and challenges involved, and survey relevant prior work. Finally, we describe the specific problems addressed and solved by the thesis.

## 1.1 Applications

### 1.1.1 Human Animation

Currently, the computer animation of humans relies heavily on human motion capture, which provides the 3D coordinates of various joints of the body for a sequence of frames. The method of choice for human motion capture involves strapping sensors to various joints of a human and recording signals transmitted by them as the human performs various actions [45]. These signals help determine the 3D coordinates of the joints in some frame of reference. Sensors are intrusive and restrict the types of action that the human can perform. A computer vision based solution eliminates the need for intrusive sensors and provides a passive and unrestricted way of obtaining motion capture data. Further, if there is a need to capture human motion from archived video, as would be the case if one needed to animate the inimitable Charlie Chaplin (as pointed out by Bregler and Malik in [9]), a computer vision based approach is the only recourse.

### 1.1.2 Visual Surveillance

A human action recognition system can process image sequences captured by video cameras monitoring sensitive areas to determine at a high-level, if one or more humans are engaging in suspicious or criminal activity. For example, if the action *human pointing gun* were to be detected, the system would alert appropriate security personnel to take action. Similarly, a camera monitoring a parking lot could do the same when the action *human breaking into car* is detected. Automatic identification of people from their gait, or *gait recognition*, is another possible application of human motion analysis. Recently, Eng et al [20] reported on a ‘drowning-detection’ system consisting of multiple cameras, for proactively detecting drowning behavior in an outdoor swimming pool. Visual surveillance applications such as these constitute a major application area of human action recognition.

### 1.1.3 Content Based Video Retrieval

A query processing system for a video library can have at its core, a human action recognition system which scans through video taking as input, an action-query specified in high-level language and producing as output, the sequence of video frames where the action was found to occur. Such an application could prove very useful for sportscasters to quickly retrieve important events in particular games. For example, when a baseball sportscaster specifies the query *retrieve all catches* or a basketball sportscaster specifies the query *retrieve all slam-dunks*, the system will retrieve those specific frames where these actions occur. Such a system will eliminate the need for cumbersome manual annotation of video.

### 1.1.4 Miscellaneous Applications

A human action recognition system designed to respond to specific gestures made by a computer user can replace or augment speech controlled computer interaction. For example, verbal commands for manipulating a virtual graphical object can be cumbersome, while gestures may be a more appropriate and intuitive medium. Interpretation of lip movements may help disambiguate phonemes [25], providing a measure of robustness to speech controlled interaction. Action recognition can also benefit the area of virtual teleconferencing where, instead of raw video, a high-level description of body-movements is transmitted to the receiving side where the encoded body-movements are used to re-create what is happenning at the transmission side. This model-based coding approach [2] would require very low bit rates for teleconferencing.

## 1.2 Problems and Challenges

Visual human motion analysis is large in scope and can be divided into several different areas of research, each with its own set of open problems and challenges. In contrast to early surveys that implicitly asked the question *What are the characteristics of various approaches for visual motion analysis?*, the last two surveys ([74], [46]) asked the question *What is the structure of an end-to-end system that recovers human motion?*, and grouped previous work into categories corresponding to different components of the end-to-end system. Such an end-to-end system will have solutions to the problems of *segmentation*, *pose-representation and estimation*, and *action-representation and recognition*. We discuss each of these areas below but prior to that, we note here that an end-to-end system need not follow such a clear-cut problem decomposition and that specific applications may not require solutions to all of these problems. We will note such exceptions during the discussions.

### 1.2.1 Segmentation

Before human motion can be recovered, the system first needs to detect a human being and initialize a model of the human. For applications such as content-based video retrieval, an additional pre-processing step of shot detection is also necessary. Segmentation involves the process of extracting the human body from the background and optionally, detecting various body-parts and body-joints. The conditions in which the image sequence has been shot and the assumptions one could make about them drive the kinds of approaches that one could take. Variations on these include: fixed vs moving camera, indoor vs outdoor scene, monochromatic vs color images, steady vs unsteady background, perspective vs weak-perspective projection, single vs multiple cameras, calibrated vs uncalibrated camera(s), loosely-clothed vs tightly-clothed humans, etc. Clearly, while specific combinations of viewing conditions are simpler to work with (e.g. multiple calibrated cameras capturing color images of a tightly-clothed human in a steady, uncluttered background) than others, one would like to remain as general as possible when seeking a solution to the problem of segmentation because one cannot always control the viewing conditions or make these assumptions.

Central sub-problems of the segmentation stage include *figure-ground segmentation*, *human-body detection*, *part-detection*, and *tracking*. We discuss each of these areas below.

### Figure-ground segmentation

Figure-ground-segmentation is the process of identifying those regions in the image, which correspond to a moving human. While this is a difficult problem to solve in the case of a moving camera, the fixed-camera case is challenging as well. Parts of the background can sometimes mimic parts of the moving human body in terms of intensity and color. Fast background variations (flickering computer screens, moving cars, flying birds, swaying trees etc.) make it difficult for the system to focus on the moving human body and extract it from the background. The imaging process is inherently noisy, adding yet another challenge to the problem.

### Human-body detection

Human-body-detection is the process of analyzing moving regions to determine if they correspond to a human being. For systems where it is known *a priori* that the viewed object is a human being, this step is unnecessary. Distinguishing a human being from other objects is a classification problem. The articulation (non-rigidity) of the human body, and differences in viewpoints lead to an infinite number of possible appearances of the body, making it especially difficult to distinguish a moving human from other moving objects in the scene. Color, if available, can help by providing additional cues, such as skin-tone, to the classification process.

### Part-detection

Part-detection is the process of identifying the regions or points on the human body which correspond to major body parts and joints, like the head, hands, feet etc. Body articulation and viewpoint variations that pose challenges for human body detection, pose challenges for part-detection as well. A challenge also arises from self-occlusions (where one body part lies behind another with respect to the camera), which are very common in human motion. A third challenge comes from the unobservability of certain body joints: very often, even when not subject to self-occlusions, many true body joints are not directly observable from the image (e.g. shoulders, knees and elbows) and at best, can only be inferred indirectly through other observable parts, through supporting edge cues or through tracking.

**Tracking**

Tracking is the process of incrementally updating the results of the solutions to the above problems per frame, without having to re-calculate them. In this sense, we track only to aid in a solution to the above problems. Tracking can predict and deal with self-occlusion and non-observability problems, using prior histories and especially when a 3D body model is employed. While tracking, we operate at a higher level of processing in that we encode the results of the above problems into a state-representation, make predictions on the movement of the state and correct the predictions using image data. The number of degrees of freedom of the state dictate the complexity of tracking. If individual body parts are tracked as image regions, the degrees of freedom remain small and tracking becomes easier but at the expense of increased complexity needed to aggregate the image regions into a solution to the above problems. On the other hand, if a complete 3D kinematic model of the human is used, predictions have to be made on the joint-angle changes for a large number of joints making tracking more challenging but with the benefit of an easy mapping to image joint locations.

Although human body segmentation algorithms that work reasonably well in restricted environments have been devised, a general solution is lacking, and the field is still developing as is evidenced by the number of articles devoted to the problem that appear each year in leading computer vision conferences and journals. Nevertheless, the output of a segmentation module would be (a) silhouette(s) of the human body and optionally, the set of body-part and joint locations in the image or in the 3D world.

### 1.2.2 Pose-Representation and Estimation

In contrast to ‘pose’ as used in object recognition where it refers to the position and orientation of an object in a camera-centric coordinate system, a human body ‘pose’ for our purposes means the configuration of the various body parts in a body-centric coordinate system. Quantitatively, pose is completely defined once the positions and orientations of the various body parts are known. Given the positions of various body parts, in principle, one can calculate the joint angles subtended by various body parts about the joints to which they are attached. This would be an instance of the inverse-kinematics problem that arises within the field of robotics, for which standard solution approaches exist. The advantage of maintaining joint angles is that it facilitates human body tracking in the image sequence, which in turn facilitates incremental updates of the pose once initialization has been done. Maintaining such a 3D *Euclidean* or *metric* representation of pose is relevant and necessary if visual motion capture is the final goal. In this case, it is not unrealistic to require manual input of the 2D joint positions during the initialization step. A user would pick joints in the image and feed the image coordinates to a motion-capture system which would initialize a model of the subject based on the image coordinates. After this bootstrapping, the system would update its model from frame to frame based on image data (e.g. [9]). For other applications where this may be unrealistic, an automatic initialization mechanism is necessary. Mapping the 2D joint locations in the image to a 3D Euclidean representation of pose is a difficult problem. The non-rigidity of the human body and its large number of degrees of freedom, make searching for solutions in the space of degrees of freedom computationally challenging. In the case of a single uncalibrated camera, the lack of depth information and perspective effects make the problem even more challenging. If the final goal is human action recognition, a complete 3D Euclidean representation of pose is not necessary. 2D stick figures, body silhouettes or 2D blobs/regions computed from the image sequences alone, may suffice. For example, in this thesis, we show that it is sufficient to maintain 2D and 3D *projective* representations of a pose and use them for action recognition.

For human action recognition, the manner in which body-poses (and actions) are represented determines the exent of applicability of an approach. There are infinite viewpoints from which a body in a given pose can be viewed, each leading to a different appearance of the body. While quantizing the space of viewpoints possible, leading to several view-dependent representations of a single body pose, this approach is cumbersome. A view-invariant body-pose representation provides the desired economy and elegance.

### 1.2.3 Action-Representation and Recognition

Johannson's [35] experiments with MLDs (Moving Light Displays) have been widely cited in the literature on visual human motion analysis. They involved subjects, outfitted in black in a dark background with lights attached to their joints, performing a variety of actions. When presented with an image of a subject showing a fixed set of lights, observers were not able to discern the presence of a human, but when presented with an image sequence showing the lights in motion, observers were able to realize that the lights came from a human and they were able to recognize the action as well as the gender of the subject using only image information. This result could be interpreted in two ways [13] : that the observers relied purely on the motion information to recognize the action, or that they reconstructed a model from the 2D data unconsciously and then used that to recognize the action. Research has followed both these interpretations in devising useful ways for representing human action.

A human action can be thought of in terms of a starting pose $\mathcal{P}_s$, an ending pose $\mathcal{P}_e$, and a sequence of continuous transitions that take the body from pose $\mathcal{P}_s$ at time $t = 0$ to pose $\mathcal{P}_e$ at time $t = T$. We can think of a *phase* of an action, $t$ that takes on a value in the interval $[0, T]$. A phase value maps to a body pose $\mathcal{P}(t)$. An action then becomes the function $\mathcal{P}(t), t \in [0, T]$. Representation of an action is tied to the representation of the pose $\mathcal{P}$. An important issue to deal with is the speed of the action or equivalently, the frame rate of the video. The action representation should be independent of these quantities. Another issue involves deciding what parts of the body are abstracted by the pose representation. Based on the application domain, for some actions, it may not be necessary for the entire body to be abstracted into the pose representation. For example, whether or not a walking person swings his/her arms, the person could be considered to be performing the action *walking*. Incorporating the arms into the representation of the pose will bring in irrelevant detail. On the other hand, if the application requires distinguishing between the actions *walking* and *walking while carrying a brief-case*, the arm position becomes relevant, and the pose representation would need to incorporate it. Yet another issue is the concurrency of body part movements. The above formalism essentially abstracts an action as a continuous state machine - the body is considered to be in one state at a given instant. In reality, human body parts move concurrently with possible periodic synchronization. For instance, it would be very difficult to represent the action *human juggling three balls* with the above formalism because while juggling, the right and left hands move independently - from the perspective of each hand, the ball is thrown immediately after it is caught. Hence, it is impossible to assign a single state to the entire body at a given time. A formalism involving Coupled Hidden Markov Models [8] or Petri Nets [18] would be better suited for representing such actions. Another challenge arises due to variability in actions: the same action executed multiple times by the same person, or by different persons will exhibit variation because humans are not consistent when they perform a given action. The action representation and recognition system should be able to account for this variability and be robust to it.

## 1.3 Prior Work

In this section we describe prior work done for each problem area we discussed in section 1.2. We follow the same order as in section 1.2. We describe different approaches for solving problems in the area in terms of their key ideas, and cite and review representative publications that illustrate the key ideas.

### 1.3.1 Figure Ground Segmentation

There are three main approaches for separating the human body from the background from a sequence of images and these are discussed below.

**Background Modeling and Differencing**

By far, a majority of methods developed for solving the figure-ground-segmentation problem have been for (a) fixed camera(s) which enable(s) the use of *background differencing*, a popular technique, for its solution. Background differencing is the process of estimating and maintaining pixel values of the background and performing a difference operation between the background and input images. The straightforward process is error-prone because in addition to fast variations in the backgrounds as described in section 1.2, slow long-term variations (moving clouds, diurnal changes etc.) pose an additional challenge. Image noise contributes to errors in the straightforward process of differencing as well. Many heuristics have been proposed for robust background maintenance using pixel statistics. Haritaoglu et al [29] maintain the minimum, maximum and maximum per-frame change in intensity at each pixel to determine if a pixel is part of the background or foreground. Stauffer and Grimson [67] model a pixel intensity as a random process arising from a mixture of adaptive Gaussians. A mixture of Gaussians is used to capture multimodality (e.g. flickering screens, specularities on water surfaces etc.) and the adaptive nature models temporal background variations. A similar example is the *Pfinder* (short for 'Person Finder') system [80] which maintains the YUV mean and covariance matrices at each pixel and allows for these to be updated recursively as the scene changes slowly.

### Temporal Differencing

Another technique for detecting moving areas in the scene is temporal differencing, which is the image difference operation applied to successive frames. The resulting image reveals areas of motion. By its very nature, the method works well only if the motion is small and it can only detect the outlines of moving objects if they exhibit little internal texture or intensity variation. An improvement of the basic two-frame differencing operation is the use of three consecutive frames [38]. The *Visual Surveillance and Monitoring* (VSAM) work done at Carnegie Mellon University combined background-differencing and temporal-differencing to detect moving objects in a relatively static background and good results were reported using the combined approach [15].

### Optical Flow

Optical flow based approaches form a third way for detecting moving regions. Assume that intensity changes between consecutive frames are caused only by moving regions and that the image regions are of smoothly varying intensity. Let $I$ denote the pixel intensity at location $(x, y)$ at time $t$. Let us say that at time $t + dt$, the region at $(x, y)$

undergoes a small translation, $(dx, dy)$. The location $(x, y)$ will now be occupied by the pixel that occupied $(x - dx, y - dy)$ at time $t$. The intensity of that pixel is

$$I - \frac{\partial I}{\partial x} dx - \frac{\partial I}{\partial y} dy$$

The rate of change of intensity at $(x, y)$ will then be given by

$$\frac{\partial I}{\partial t} = -\frac{\partial I}{\partial x}\frac{dx}{dt} - \frac{\partial I}{\partial y}\frac{dy}{dt} \tag{1.1}$$

Denoting the *flow*, i.e. $(dx/dt, dy/dt)$ as $(u, v)$ we have the *brightness constraint* which is an equation that the components of the flow satisfy at each pixel:

$$\frac{\partial I}{\partial t} + \frac{\partial I}{\partial x} u + \frac{\partial I}{\partial y} v = 0 \tag{1.2}$$

This is an ill-posed problem because $(u, v)$ are both unknown but we only have one equation that they satisfy (note that the partial derivatives can be approximately calculated from the image using finite differencing). One way is to assume a parametric form for optic flow in a region, which will provide an overdetermined set of equations in the parameter values (e.g. [65]). For human body segmentation, these regions correspond to moving body parts. The parametric form for flow is dictated by the human kinematic model. Ju et al [37] describe a 2D approach where major body parts are approximated as planar patches connected to each other at joints. Bregler and Malik [9] employ a more elaborate 3D kinematic model of the body and use optical flow to track the human body. Both approaches require manual initialization after which the optical flow based approach tracks the body.

### 1.3.2 Human Body Detection

Virtually all systems for which this is a necessary problem to solve, simplify the problem by designing a view based approach. Mohan et al in [47] use Haar wavelets and

Support Vector Machines to detect the human body and the head, legs, and arm locations from the front, side and rear views. They use a two stage approach: the first stage involves testing small windows throughout the image for the presence of each considered body part while the second stage involves combining the results to determine if there is a human in the image. Collins et al [15] report a view dependent neural network based classifier and a Linear Discriminant Analysis [19] based classfier to distinguish between the classes *human*, *human group*, and *vehicle*. In contrast to these shape based classfication methods, motion based methods have also been reported. Lipton in [41] defines *residual flow* at a pixel as the difference between the overall body (or ‘blob’) displacement and the optic flow. He uses the heuristic that the residual flow will be higher for a human body due to its articulation, than for a rigid body, to perform classification. Similarly, Stauffer in [66] uses binary motion silhouettes to discriminate between a vehicle and a human being.

### 1.3.3 Part Detection

One of the earliest attempts to detect body parts from an image sequence of a moving human in a general scenario was by Leung and Yang [40]. They first detected ‘2D ribbons’ in each image, which were calculated using a robust edge detector and which corresponded to the limbs, torso, and head of the body. Following this, a series of heuristics were applied to the 2D ribbons to detect the head, torso and limbs. *Ghost* [28] was a system developed by Haritaoglu et al for labeling a human body silhouette with locations of the body parts and joints. By applying a series of heuristics on recursive convex hull computations on the body silhouette, they were able to identify several body parts and joints. The Pfinder system [80] mentioned in section 1.3.1 is able to detect and track the head and hands of the human body using flesh-color as a prior. Rosales and Sclaroff in [60] describe an interesting neural-networks based approach for mapping a 2D silhouette to a set of 2D joint locations. A 2D silhouette in their case is a binary image (i.e. the intensity is 1 inside the silhouette and 0 outside) as obtained by figure-ground segmentation. The silhouette is represented by Hu moments [32] which are quantities derived from moments of inertia of a figure (which in this case is the silhouette). An appealing quality of Hu moments is their invariance to 2D translation, rotation and scaling. The training phase involved synthesizing a number of silhouettes of an average human with average clothing, from a number of views, from motion capture data. These are grouped into a fixed number of clusters using unsupervised learning. A neural network is trained to map Hu moments from the silhouettes of each cluster to the 2D positions, which are known. Then given an unknown image sequence, figure-ground segmentation is applied and the neural network is used to map the binary silhouette to the 2D joint positions.

If available, multiple cameras provide more information for the detection of body parts and localization of joints. For example, Gavrila and Davis in [24] describe a system for the determination of the 3D positions of various joints using multiple calibrated cameras. After figure-ground segmentation, principal components analysis (PCA) [36] is applied to the silhouette, which allows the determination of the head-torso axis in the image, from each view. Given that the cameras are calibrated, this allows the determination of the 3D location of the head-torso axis. Thereafter, through a search based procedure, various values for the 3D joint angles are hypothesized and verified by synthesizing the hypothesized appearance and verifying it against the true appearances from each view. Another representative paper using a multi-camera approach is that of Grauman et al [26] who recently presented a probabilistic approach for estimating the 3D coordinates of different joints of the body from images from multiple calibrated cameras, of a walking human. They first use an animation package to render various poses from various phases of a human walking to obtain silhouettes as would appear from the cameras in the real scenario. The silhouettes are represented by a vector of 2D coordinates of their contours. Since the 3D joint positions are known during the training phase, they augment the 2D contours with 3D joint positions. Using training data, they cluster the input space of multiple 2D contours and 3D joint positions. When presented with images from real cameras at the same locations, using a Bayesian approach, silhouettes from multiple views of a body in an unknown pose are mapped to the 2D contours and the associated 3D coordinates.

### 1.3.4 Tracking

As we mentioned earlier in this section, tracking aids in figure-ground segmentation, human body detection and body part detection because once these are initialized, tracking enables the incremental update of their outputs. Tracking at the level of body parts is helped by the use of a human body model, which encodes a state (or pose) of the human body. Given the state of the body model in an image, tracking then amounts to estimating the amount of change in state in the next image of the sequence. Given the noisy nature and uncertainties inherent in the body detection and state estimation process, probabilistic state estimation tools such as the Kalman filter [78], its variants, particle filters [4] and the related Condensation algorithm [34] have found good use in human body tracking. The model could be two dimensional, as in the use of 2D stick figures (e.g. [27]) and 2D contours (e.g. [40], [37]), or three dimensional. Three dimensional models are advantageous for their help in predicting and dealing with occlusions. An example is the work of Rohr [59] who modeled the body as a collection of fourteen elliptical cylinders. His work dealt with detecting the pose of a walking human as viewed from the side, using a calibrated camera. After searching in one dimension (the phase of the walking action), he used the Kalman filter to track the human and his pose. Bregler and Malik [9] used ideas from robotics and modeled the human body as a kinematic chain and modeled the flesh as ellipsoids. After initializing the model by hand they proceed to use the projection equations and optic flow to guide the incremental update of the model. Tracking approaches with less sophisticated or no explicit human models have also been reported. McKenna et al [44] model the body as a collection of regions with color distributions. The Pfinder [80] work discussed earlier models body parts as 2D blobs and tracks them separately using the Kalman filter. Fablet and Black [21] use a view based representation for human motion using optical flow. For training, they use motion capture data of subjects walking to render optic flow from several views and use PCA to bring down the dimensionality of the representation. When presented with a novel sequence, they calculate optic flow and calculate the posterior distribution of the model (i.e. action) parameters. They use particle filters to track the posterior probability.

### 1.3.5 Pose-Representation and Estimation

For visual motion capture applications, where it is necessary to recover the body motion in 3D, a metric representation of pose is necessary. When dealing with a monocular image sequence from an unknown camera (as would be the case for archived video), the input, as we described in section 1.2.2, is typically a set of image locations of the body, from which an initial estimate of the pose is required. To deal with the large number of degrees of freedom of the human body, researchers have resorted to assuming a scaled orthographic camera or requiring an unrealistic number of point correspondences. For example, Lee and Chen in [39] work with a single camera. However, they assume that a minimum of six point correspondences of the head be known (e.g. both eyes, both ears, neck and nose), which allows them to recover the transformation from a head-centered coordinate system to the image plane. While it is possible to obtain these correspondences for motion capture based input (which was the only modality of input that they tested their approach on), it is unrealistic to assume that all of these head features will be visible in a real image of a human body. In [71], Taylor uses a scaled orthographic uncalibrated camera model and relates the foreshortening in the image plane of a body limb to its 3D depth from the camera. The projection model relies on an unknown scale factor which is fixed to an arbitrary value. After fixing the scale factor, he shows how one can calculate the relative depths of all the joints of the body. Even so, there are an exponential number (in the number of limbs) of possible solutions because of the forwards-backwards flipping ambiguity, where given that a limb is foreshortened it is not clear which joint is closer to the camera (see figure 1.1). These ambiguities are resolved by additional user-input. In [6] Barron and Kakadiaris use the scaled orthographic camera assumption and user supplied image joint locations to estimate anthropometry and pose from a single uncalibrated image.

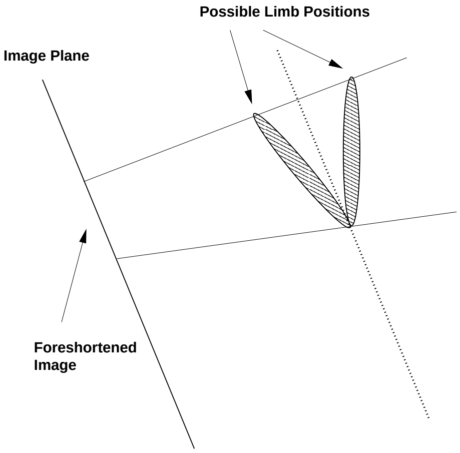


Figure 1.1: Foreshortening of a Limb in the Image

In order for the scaled orthographic assumption to apply, they require atleast two limbs to be almost parallel to the image plane and require the user to specify the limbs. They calculate the most plausible limb length ratios that explain the image using a search based procedure, constrained by anthropometric statistics. Bregler and Malik's work [9], cited earlier in this chapter, also has an initialization step where the joint angles of the body are calculated. They use the scaled orthographic projection approximation and also rely on manual input of joint locations in the image and report a search based procedure to recover the body lengths and joint angles. Recently, Sminchisescu and Triggs [64], reported on an approach for monocular 3D human tracking assuming that camera parameters are known. Manually input joint positions of the image are used to search for the set of 3D joint angles that best explain the image.

For action recognition applications, a complete 3D reconstruction of the body is helpful if possible, but not necessary. In the case of multiple calibrated cameras where the body and 3D part locations are recovered as part of the segmentation stage, this stage is not necessary. For the single camera case, pose can be represented as a quantity derived from the image domain. Fujiyoshi and Lipton [23] represent pose by way of a 'star-skeleton' of the extracted silhouette which is the figure obtained by connecting the centroid to the local maxima of the boundary distance from the centroid (see figure 1.2). Assuming that the human is in the upright posture, the bottom maxima are taken to be the feet and the top maximum is taken to be the head. They were able to distinguish between the actions *walking* and *running* from the side-view, based on the angle of inclination of the head to the vertical (which is higher for running than for walking because the head lunges forward more during running) and the frequency of the cyclic motion of the feet (which is higher for running). Ali and Aggarwal [3] also use a skeleton based representation of pose for action recognition from the side-view,

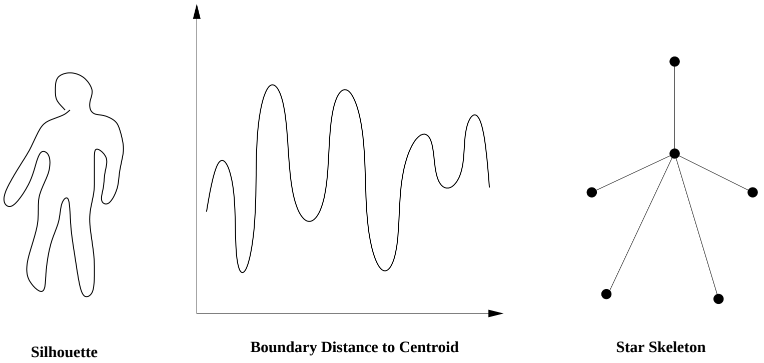


Figure 1.2: Fujiyoshi and Lipton's 'star-skeleton'

although their skeleton is the standard medial axis based skeleton which is obtained by joining points that are equidistant from the boundary.

### 1.3.6 Action-Representation and Recognition

An action representation needs to encode the evolution of pose with time and deal with spatiotemporal variability (recall the discussion in section 1.2.3). Temporal variability can be modeled by *dynamic time warping* (DTW) which has found good use in speech recognition [51]. It involves the stretching or shrinking of time to match a given input signal to a reference signal. The search for the best set of stretches and shrinkages of time is found using dynamic programming [16]. Hidden Markov Models (HMMs) are more sophisticated, in that they can encode spatial (i.e. pose) variability as well by incorporating probability distributions of the pose attributes. Rabiner's tutorial [57] provides a good description of the theory and practical issues involved in the implementation of HMMs. While HMMs abstract the system as a finite state machine, Petri nets provide more sophistication in being able to model the system as a set of concurrently evolving sub-states. With the formalism of Petri nets, one can model concurrency as well as synchronization between various parts of the system. David and

Alla provide a good introduction to Petri nets in [18]. Coupled HMMs [8] are another choice for modeling concurrency because they combine HMMs and allow transitions of one HMM to be conditioned on the states of other HMMs that are coupled to it.

Yacoob and Black [81] build on their parametric optical flow method [37] (recall our discussion of [37] in section 1.3.1) and model actions as view-based temporal trajectories of the optical flow parameters of the arm, torso, thigh, calf and foot. After manual initialization of the body, their tracking module recovers and tracks the body and optical flow of each part. They model and recover temporal variations in the actions by an 'affine' transformation of time, $t^{'} = \alpha t + L$, where $\alpha$ is the temporal scaling parameter that can model speed-ups and slow-downs and $L$ accounts for changes in the starting time of the action. Brand et al [8] describe a coupled HMM for representing and recognizing Tai-Chi gestures made by two hands. In the Tai-Chi sequences the authors considered, the two hands can move independently as well as in synchrony with each other which makes the actions suitable for being modeled by a coupled HMM. They work with stereo, which allows them to obtain the 3D coordinates of the left and right hands, which are used for training their coupled HMM. An example of the use of Petri nets for modeling concurrency and synchrony in human action is the work by Nam et al who describe a gesture recognition system in [52]. The subject wears a special glove whose signals form the input to their system. Their gesture database consists of several colored Petri nets corresponding to each distinct gesture that the hand can perform. Unknown gestures are matched against each gesture in the database.

### Direct recognition from Image Features

Several approaches have avoided reconstructing or representing the pose of the human for action representation and recognition and sought a mapping from low level image features directly to an action. Chomat and Crowley [14] use Gabor filters tuned to specific orientations and frequencies to capture the spatiotemporal dynamics of actions. When presented with an unknown image sequence, their system computes the Gabor filter responses and uses the Bayes rule to calculate the action probabilities. Their approach is sensitive to temporal and viewpoint variations. In fact, the *walking* action as seen from the front, back, left and right views is considered as four completely different actions. Yamato et al in [82] describe an HMM based approach for classifying various tennis strokes when viewed from the side. The human is extracted and the image is binarized such that the human is represented in white and the background is represented in black. They divide each image into pixel-blocks of configurable size and calculate the fraction of black pixels in each block. This becomes their feature vector which is used in the modeling of HMMs and in the recognition of actions. Polana and Nelson [56] describe a way to detect a periodic action based on its spatiotemporal appearance. The moving human is isolated into a bounding box and flow is computed in pixel-blocks. If it is determined that the motion is periodic, template matching is performed on the bounding boxes to classify the action. Bobick and Davis present an interesting approach in [7] that uses motion cues to represent and recognize actions in a view-dependent manner. They perform background differencing for each image and create two ‘meta-images’ per image in the sequence: the motion energy image (MEI) which is a binary image showing pixels that have had a difference and the motion history image (MHI) where each pixel has intensity proportional to the recency of motion at the pixel. Hu moments [32] of the MEIs and MHIs form the representation of an action.

### View-Invariant Action Representation and Recognition

Generally speaking, aside from multicamera approaches where the issue of viewpoint-invariance is not of particular concern because the action can be represented and recognized in 3D directly, prior work has largely ignored the viewpoint-invariance issue, especially where it has mattered the most - where the input is monocular video with unknown camera parameters. Nevertheless, some prior work has addressed the issue directly. Seitz and Dyer in [63] have described an approach to detect cyclic motion that is affine invariant assuming that feature correspondence between successive frames is known. Their goal of cyclic motion detection requires the comparison of two images for similarity, so that a repeating phase of an action can be detected. They generalize the Tomasi and Kanade rank theorem [72] for the affine case, and use it to verify if there are repeating shapes in the scene. Rao, Yilmaz and Shah [58] consider the problem of learning and recognizing actions performed by a human hand. They target affine invariance and apply their method on real image sequences, using skin tone to segment the hand. They characterize an action using *dynamic instants*, which they define as maxima in the spatio-temporal curvature of the hand trajectory which are preserved from 3D to 2D. Their approach does not require a model and builds up its own model database based on input actions. Syeda-Mahmood et. al. [69] represent actions as ‘generalized cylinders’, formulating the action recognition problem as a joint action-recognition/fundamental-matrix recovery problem. Their core recognition module requires that the starting and ending positions of an action are known, and various starting and ending positions are hypothesized exhaustively. Further, their approach is sensitive to variations in the speed of execution of the action as well as the frame rate. Campbell et. al. in [12] consider the problem of extracting view invariant features for 3D gesture recognition. The input is stereo data which allows them to work on the 3D coordinates of the two hands directly. They consider various possible features for representing Tai Chi moves using a Hidden Markov Model and report on their performances.

## 1.4 Contributions of this Thesis

This thesis makes contributions to the last two of the three problem areas described above in section 1.2. The contributions are driven by three new ideas which are described below:

1. A new approach for human action representation and recognition has been devised that is based on 2D projective invariance theory. Static and temporally varying 2D projective invariants are used to represent the spatiotemporal dynamics of human action to enable its recognition in a quasi-view-invariant manner. We present a detailed analysis of the approach which reveals weaknesses inherent in a straightforward implementation of the key ideas. We propose heuristics designed to surmount these weaknesses and improve its robustness. What results is an action representation and recognition approach that is not only resistant to changes in viewpoint, but is robust enough to handle different speeds of action (and hence frame-rate), different subjects and minor variabilities in the action. We present results on 2D projections of motion capture sequences as well as manually segmented real-image sequences.

2. A 3D approach to the human action representation and recognition problem that is more general than the pure-2D approach outlined above has been also been devised. Prior work done in the area of mutual invariants for object recognition is used as the foundation for the derivation of two new approaches for action

representation and recognition. We call these the *restricted-3D* approach and the *full-3D* approach. The theory and implementation of the restricted-3D approach are simple and efficient but the approach is applicable only to a restricted class of human action. On the other hand, the theory and implementation of the full-3D approach are more complex but the approach can be applied to any general human action. A detailed analysis of the two representations is presented which, like in the pure-2D case, reveals weaknesses which make them unlikely to work well in the general case. Heuristics similar to those in the pure-2D case are designed to improve the robustness of the method. Results on 2D projections of human motion capture and on manually segmented real image sequences demonstrate the effectiveness of the approach.

3. The final idea is a contribution to the area of visual human motion capture, and enables the recovery of the 3D coordinates of various body joints in a canonically scaled body-centric coordinate system, given simply the locations of those joints in a single perspective image taken from an uncalibrated camera. This problem has previously not been addressed in a general way and can form the initialization step of a visual human motion capture system. We make the (usually plausible) assumption that torso twist is negligible, which allows us to apply the restricted-3D theory (outlined above) to the problem. We first recover the head orientation by setting up a simple system of polynomials in terms of the head rotation angles. Once the head orientation is recovered, the systematic recovery of all of the other body joint coordinates becomes possible. We perform an empirical analysis of the sensitivity of the results to model and image noise. We also present results of the approach on real images.

## 1.5 Organization

Chapter 2 presents a brief review of the fundamentals of viewpoint invariance specifically as it relates to the thesis. For this reason, the focus is on projective view invariance rather than invariance for other camera models. Chapter 3 describes a 2D approach to the human action representation and recognition problem based on 2D projective invariants. In chapter 4, we describe 3D approaches to the problem based on mutual invariants. Chapter 5 describes an application of mutual invariants for the recover of the the human body pose from a single uncalibrated image taken from a perspective camera. We conclude and discuss extensions to the three contributions in chapter 6.

# Chapter 2

# View Invariance

## 2.1 Introduction

The viewpoint is a variable not particular to any human action that may be taking place in a scene being observed by a camera. It is desirable, as we observed in chapter 1, to make the action representation and recognition system independent of the viewpoint and the camera. In many applications of human motion analysis, such as content based video retrieval and visual motion capture, the viewpoint and camera information are typically not available. For such applications, viewpoint invariance becomes a good property to seek. The question we seek to answer is : *Given image sequences observing the same human action from different viewpoints and different cameras, do quantities exist that can be computed from the image sequences and whose values are preserved across all viewpoints?* In this chapter, we briefly review the necessary background required to answer the question which in turn provides a foundation for the algorithms introduced in this thesis and described in later chapters.

## 2.2 Projective Geometry

Perspective projection enables the modeling of the image formation process in the most general way, although approximate camera models suitable for certain situations have also been devised and used. These have included the *orthographic*, *weak perspective* and *affine* camera models. A question similar to what we posed in the previous section, has been pursued in the domain of single-image unarticulated object recognition where several well-founded algorithms have been devised in the past. The general foundation for these algorithms, and for the study of perspective effects, is the field of projective geometry. Besides [49], [50] and [48], which provide good starting points for understanding various techniques for achieving viewpoint invariant object recognition, Horadam [31] provides an accessible introduction to the mathematics of projective and related geometries. Whicher [79] provides a complementary non-algebraic and purely pictorial description of the main ideas and results of projective geometry.

A key feature of projective geometry is the use of homogenous coordinates, which enables the representation of points that are an infinite distance away. This is necessary because points at 'infinity' can map (i.e. project) to finite locations in their image: a familiar example is that of an image of a road where the two parallel sides appear to meet at a finite location in the image [48]. Similarly, finite points can map to infinity as well. The homogenous representation of a point in $n$ dimensional projective space involves using a vector of dimension $n+1$ where one of the elements (typically the last one) represents a scale factor. In other words, $\mathbf{X}$ and $\lambda\mathbf{X}$ ($\lambda \neq 0$) represent the same point. The 'true' coordinates of the point are obtained by dividing each homogenous coordinate by the scale factor. For example, given the homogenous coordinates of a pixel as $(x, y, t)$, the true pixel location is given by $(x/t, y/t)$. Points at infinity correspond to $t = 0$. With this representation, points at infinity are represented in the same way as points not at infinity - a key departure from Euclidean and affine geometries.

## 2.3 Mapping the World to an Image

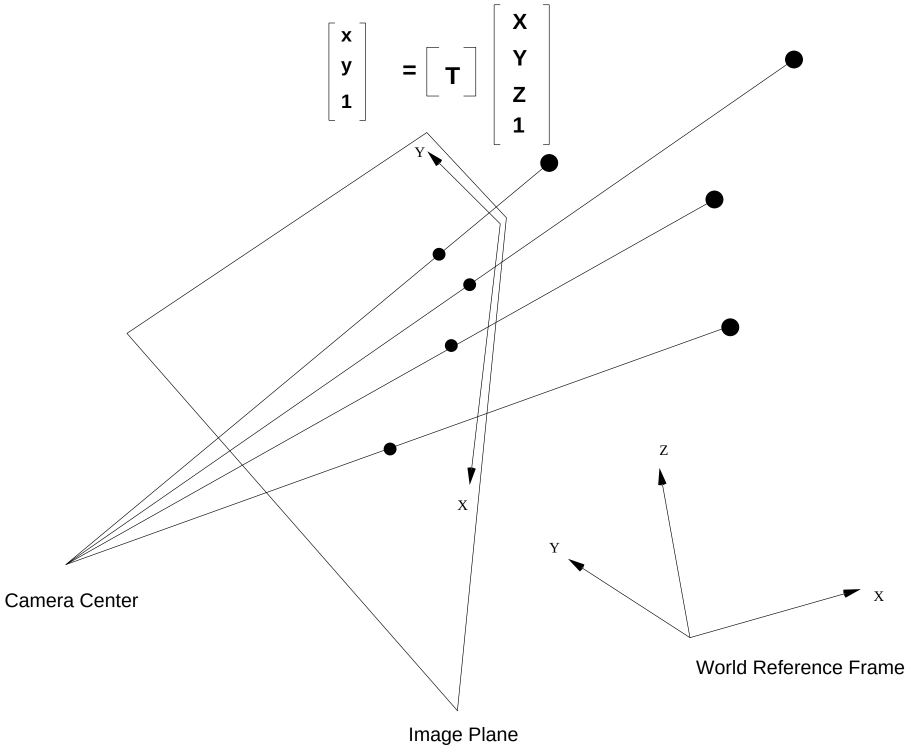


Figure 2.1: Transformation from World to Image

Throughout this thesis, $\mathbf{X}$ denotes the vector 3D coordinates of a point in the world and $\mathbf{x}$, the vector 2D coordinates of a point in the image. $X$, $Y$, $Z$ denote the components of $\mathbf{X}$ while $x$, $y$ denote the components of $\mathbf{x}$. $\mathbf{X}$ and $\mathbf{x}$ will represent homogenous or non-homogenous coordinates depending upon the context of discussion.

The projection of a 3D world onto the 2D image plane (figure 2.1) can be described by a projective transformation $\mathbf{T}$ which maps a 3D world point $\mathbf{X}$ to a 2D image point $\mathbf{x}$. Taking both, $\mathbf{X}$ and $\mathbf{x}$ as homogenous coordinates here, this makes $\mathbf{X}$ a 4-vector and $\mathbf{x}$ a 3-vector. Given this, the transformation $\mathbf{T}$ is necessarily a $3 \times 4$ matrix. $\mathbf{T}$ has eleven parameters in the general case. We are particularly interested in cases where we do not know beforehand the objects in the scene, and the properties, position and orientation of the camera in the world. This means that $\mathbf{T}$ is unknown and will assume different values for different viewpoints and cameras observing the same world point $\mathbf{X}$. Transformations between spaces of unequal dimensions, such as $\mathbf{T}$, are difficult to work with because of their non-invertibility and the resulting inherent loss of information in the transformation process. On the other hand, projective transformations between spaces of equal dimensions are easier to study because there is no such loss of information. Given this, it is not surprising to note that the literature on the calculation of projectively invariant quantities that arise in 2D-2D transformations (figure 2.2) is rich (e.g. [61], [49]). Given a set of world points in homogenous coordinates

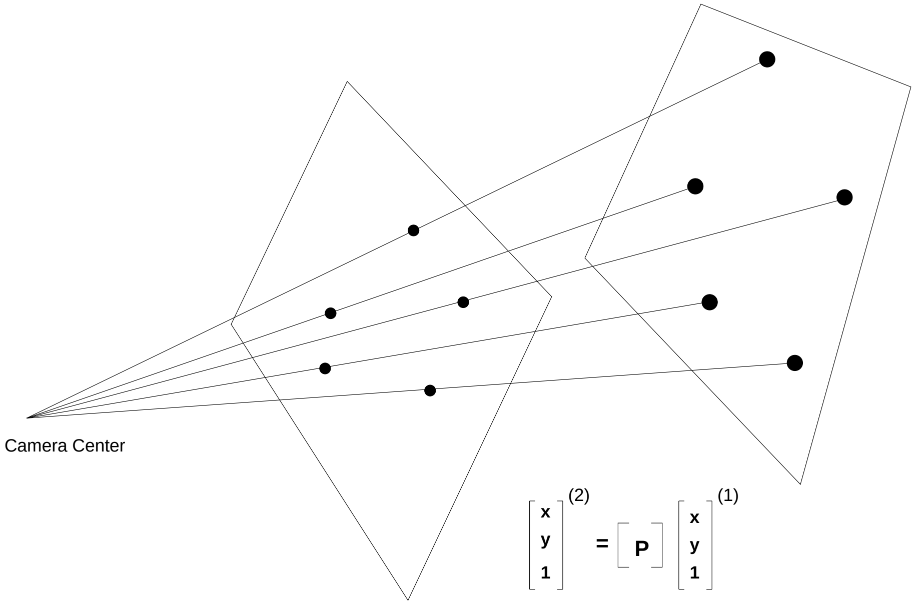

Camera Center
$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}^{(2)} = \begin{bmatrix} P \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}^{(1)}$

Figure 2.2: Transformation Between Two Planes

$\mathbf{X}_i$, $i = 1, 2...N$ that lie on a plane, and given the homogenous image-coordinates of the points from two different views, namely $\mathbf{x}_i^{(1)}$ and $\mathbf{x}_i^{(2)}$, it is well-known that the two views are related by a transformation $P$ such that

$$\mathbf{x}_i^{(2)} = P\mathbf{x}_i^{(1)}$$

$P$ is a $3 \times 3$ matrix with eight degrees of freedom, also called the *planar homography* (Appendix A provides a proof of this). Transformations between spaces of equal dimension like $\mathbf{P}$ can be generalized to $n$ dimensions, where the transformation matrix is of size $(n+1) \times (n+1)$ and is also called a *collineation*.

## 2.4 Invariants

The search for invariants becomes the search for quantities, given a specific configuration of points[1] that are preserved for all possible values of the transformation matrix, be it $\mathbf{T}$ or $\mathbf{P}$. Burns et al [10] answered the question in the negative for $\mathbf{T}$. In other words, they proved that there exist no non-trivial quantities that can be computed from an image of a set of 3D world points, that are invariant to $\mathbf{T}$. For collineations such as $\mathbf{P}$, however, there do exist invariants, a fact which has been known for centuries atleast for the case of 1D projective space. In terms of metric quantities, it can be shown that collineations do not preserve lengths, angles and ratios of lengths. However, ratios of ratios of specific quantities (also called *cross-ratios*) *are* preserved. The nature of the quantities whose cross-ratios are taken, depends upon the dimension of the underlying projective space. For 1D they are distances, for 2D they are areas, for 3D they are volumes while for higher dimensional space (which is not particularly relevant for computer vision), they are hypervolumes. Figure 2.3 shows the 1D cross-ratio

[1] We mention points here but the search for invariants is not exclusive to points - lines, curves and surfaces can and have been considered as well. We talk about points for ease of presentation.

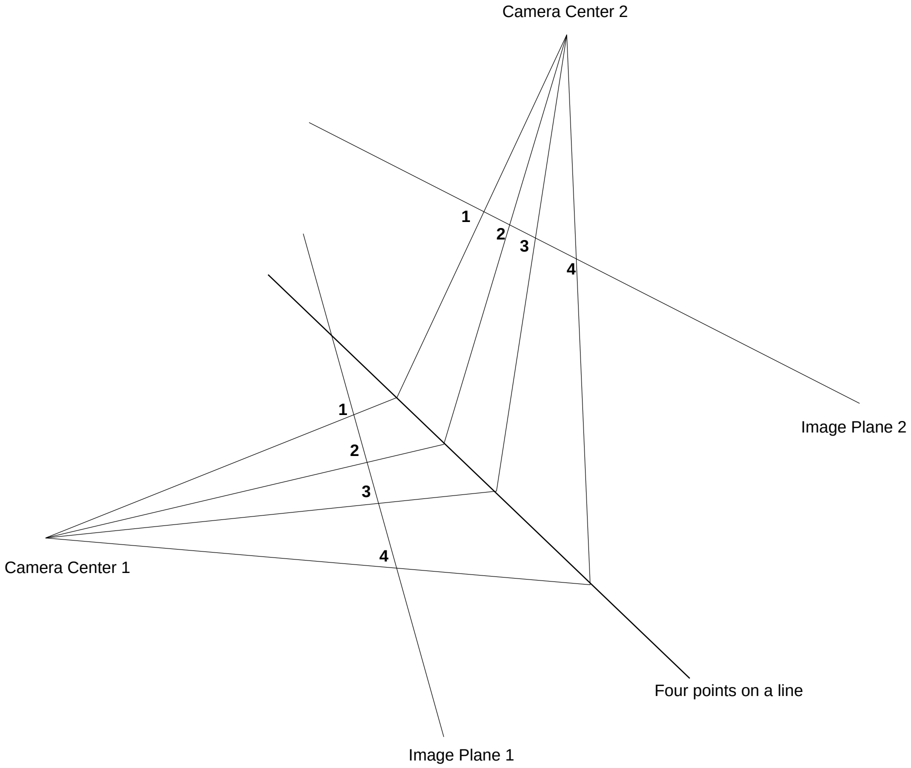


Figure 2.3: 1D Cross-Ratio requires four points.

construction which requires four points and produces one invariant which is given by:

$$C_{1d} = \frac{\overline{13}.\overline{42}}{\overline{23}.\overline{41}} \tag{2.1}$$

where $\overline{ij}$ is the signed distance between points $i$ and $j$. For a plane, five points are required which result in two invariants calculated as follows:

$$I_1 = \frac{M_{421}M_{532}}{M_{432}M_{521}} \quad , \quad I_2 = \frac{M_{421}M_{531}}{M_{431}M_{521}} \tag{2.2}$$

where $M_{ijk}$ is the signed area of points $i$, $j$, and $k$ given by the determinant[2] $|\mathbf{x_i} \;\; \mathbf{x_j} \;\; \mathbf{x_k}|$. A proof of this result can be found in Appendix A. In general, $n$ dimensional projective space requires $n+2$ points to form a projective basis and requires $n+3$ points to form invariants resulting in $n$ invariants.

[2] This requires that no three points are collinear.

As is speculated in [49], all types of invariants calculated in these spaces can perhaps ultimately be traced to some cross-ratio construction. Besides the above quantitative invariants, there exist qualitative invariants as well: $\mathbf{T}$ and $\mathbf{P}$ map points to points and lines to lines, preserving collinearity, incidence and concurrency, all of which can be used for viewpoint invariant representations of objects.

## 2.5 Viewpoint Invariant Representation and Recognition

A bulk of prior work in the object recognition field has concerned itself with 2D projective space using 2D invariants to represent objects (e.g. several papers in [49] and [50], [61]). Besides point based constructions that lead to two invariants for a five point tuple, invariance for 2D curves has also been studied. Four points define a 2D projective basis. Any four points on a plane can be chosen to form a projective basis and be mapped to a canonical coordinate system $(0, 0, 1), (1, 0, 1), (0, 1, 1), (1, 1, 1)$. The coordinates of any other point expressed in the transformed coordinate system will then be invariant to all collineations of the plane. This fact has been used by Rothwell et al in [62] to match non-algebraic planar curves. Given a curve, their points of bitangency are calculated first, (which are preserved by collineations). Tangents formed by rays cast from the two points onto the curve are calculated next to give four total points. These points are chosen to form a projective basis, into which the entire curve is mapped (see figure 2.4). By definition, all views of the curve will give rise to the same curve in the canonical frame, making the canonical frame curve an invariant. Rothwell in [61] describes a machine vision system called LEWIS where these ideas are put to use. Weiss [76] describes the construction of a different kind of canonical

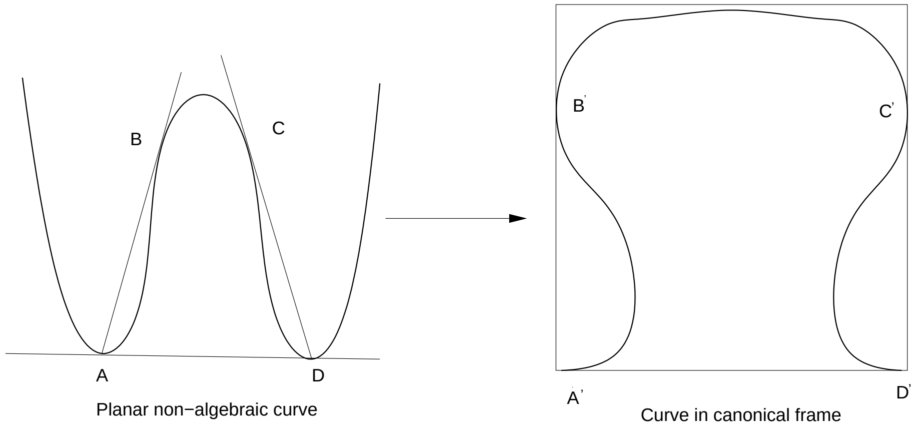


Figure 2.4: Rothwell et al's canonical frame construction using bitangents

reference system. He first fits a quartic to the given set of points on the planar curve to be represented or recognized. Next, an osculating curve[3] of known form is calculated at each point. A series of transformations (including projective transformations) is applied to the osculating curve so that it takes on a standard form. The same series of transformations is applied to the given curve. The coefficients of the transformed curve are by definition projectively invariant because the same coefficients will be arrived at, no matter which view of the curve we started with.

## 2.6 View invariant 3D object recognition

The negative result of Burns et al [10] for $\mathbf{T}$ does not necessarily imply that 3D object recognition cannot be done in a view-invariant way. *Model based invariants*, also called *mutual invariants* is a relatively newer area of study ([75], [68], [77]), where the objective is to develop compatibility relationships between quantities calculated from

[3] Osculating curves generalize the concept of tangency: an osculating curve to a given curve at a given point is that curve which has the same set of the first $k$ derivatives at the point as the given curve. $k$ is called the order of contact.

3D coordinates and quantities calculated from their image. Rather than the quantities themselves, it is the relationships between them that are invariant to viewpoint. These ideas extend the applicability of viewpoint invariant representation and recognition approaches from points, lines and curves on a plane, to 3D objects.

In this thesis, we extend and apply some of the ideas from these viewpoint invariant representation and recognition schemes to the task of human motion recovery. A unique challenge for human motion recovery is the requirement that the system deal with the non-rigidity and the dynamics of the human body and its movement.

# Chapter 3

# 2D Approaches for Action Representation and Recognition

## 3.1 Introduction

In this chapter, we show how ideas from prior work in 2D-2D projective invariants can be adapted and applied to our goal of representing and recognizing human action in a view-invariant manner. We observed in the previous chapter that based on the counting argument, for 2D projective transformations that have eight degrees of freedom, five points will give rise to two invariants. There are three different techniques to obtain the invariants: (1) 2D cross-ratios, (2) the canonical coordinate system technique where the first four points are mapped to known positions such that the two coordinates of fifth point in the system become invariant and (3) the cross-ratios of two pencils, where the first two points are connected to every other point thereby forming two pencils and the cross ratios of any two lines intersecting the pencils become the invariants. We choose the 2D cross-ratios because unlike the other two techniques, the associated $3\times3$ determinants can be calculated rather easily and the expressions (equation 3.1) have a simple geometric interpretation in terms of cross-ratios of areas. In our discussions, we will use the term ‘invariants’ for the above values when they refer to the model and the term ‘area-cross-ratio’ when they refer to the calculated quantities from the image. For convenience, we repeat the expressions from chapter 1 here:

$$I_1 = \frac{M_{421} M_{532}}{M_{432} M_{521}} \quad , \quad I_2 = \frac{M_{421} M_{531}}{M_{431} M_{521}} \tag{3.1}$$

where $M_{ijk}$ is the determinant $|\mathbf{x_i} \ \ \mathbf{x_j} \ \ \mathbf{x_k}|$.

## 3.2 Key Ideas

We can represent the action in a view independent way by choosing key poses such that atleast five body joints align themselves approximately in a plane (we say *approximate*, here because we can show (as we do in section 3.3.3) that we can achieve good results even with approximate rather than total planarity). We call such poses *canonical* poses. This is indeed possible for a variety of human actions. Let us consider the example of the action *walking*. During walking, the palms, feet and head of the person fall in a vertical plane periodically (e.g. first image in figure 3.2). Figure 3.1 shows the distances of the right-foot and the head to the plane formed by the right-shoulder, left-shoulder and the left-foot of a walking person from motion-capture data. The distances have been normalized by the head-to-toe length. A zero-crossing of two curves indicates that the five joints are on a plane. Let us denote this canonical pose for walking, where the palms, feet and head lie approximately on a plane as $C_1$. The detection of $C_1$ by itself provides us a strong cue that a ‘walk-like’ action is taking place and can become part of the action model for walking. In addition to the $C_1$ pose, each corresponding leg and hand of the body trace areas lying approximately on a plane. In other words, the left hand and leg as well as the right hand and leg lie approximately in a plane in each walk cycle. Further, there exist two poses where the body limbs swing

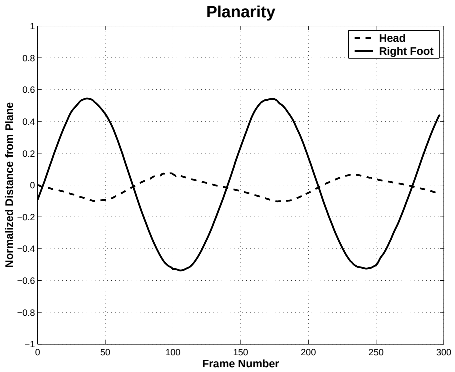


Figure 3.1: $C_1$ Planarity

to their maximal positions and the end effectors attain their maximum distances from the torso. Call these canonical poses $C_2$ and $C_3$ (see figure 3.2). There exist many joint combinations that could be used to represent $C_1$, $C_2$ and $C_3$. The requirement to be satisfied of a joint combination is that the joints lie approximately on a plane for the corresponding pose. We can pick tuples of five such joints and use (3.1) to precompute a pair of invariants for each tuple. Pairs of invariants for several five-tuples can be used to represent a particular body-pose which occurs at a single instant of an action. For the action *walking*, a simple action model would thus be the repeating sequence $C_1$, $C_2$, $C_1$, $C_3$. Clearly this type of abstraction of an action in terms of the

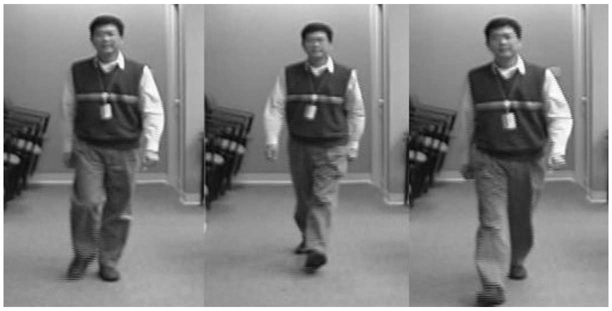


Figure 3.2: Canonical poses $C_1$, $C_2$ and $C_3$

invariants from a planar decomposition is possible only if we can locate a significant number of poses in the action where five or more joints lie approximately on a plane and is not a completely general method for representing any arbitrary action. However, in practice, this is possible for many other actions including jumping, waving, running, sitting-down etc. For achieving viewpoint invariance, besides canonical poses, we can also exploit stationarity of one or more joints across frames because stationarity is also viewpoint-invariant (a qualitative invariant similar to those discussed in chapter 1, section 2.4).

## 3.3 Analysis

Faugeras in [22], drawing on Marr and Nishihara's [42] three criteria for judging the effectiveness of a 3D shape representation, lists four criteria for judging the effectiveness of a geometric primitive representation: *minimalism, completeness, continuity* and *uniqueness*. We believe that in addition to being desirable qualities of a geometric primitive representation, these criteria can serve as qualitative benchmarks for an action representation as well. In the following sections we briefly consider each of the above criteria and see how our human body pose (and hence action) representation measures against them. We also consider the sources of variability in the 2D 'invariants' and the effectiveness of the use of invariants for classification.

### 3.3.1 Minimalism

A representation should be minimal; i.e. it should use no more than the required number of parameters. A human body pose is completely described by the values of all joint angles of the body. But because this cannot be done in a view-invariant manner, our approach picks five joints that lie approximately on a plane and represents their relative configuration by use of two numbers, $I_1$ and $I_2$. Whether or not the five chosen joints sufficiently capture the pose of the body is a question answerable only relative to the specific pose and specific five-tuple of joints used. However, if the chosen joints do capture the pose effectively, two scalars are indeed the minimal number of parameters needed to represent the pose because there are two degrees of freedom for five points in 2D projective space.

### 3.3.2 Completeness

The representation should be complete; i.e. for each configuration, there should exist a representation. Clearly, our representation does not allow the representation of every pose of the body because not all poses will have a sufficient number of joints that lie close to a plane. Secondly, for degenerate positions where three joints are collinear, there does not exist a representation. These two problems are not of consequence for us because our choice of points specifically avoids them - we only pick canonical poses (i.e. those with a sufficient number of joints on a plane) and the set of joints in non-degenerate positions for our representation. For such poses, there is indeed a representation.

### 3.3.3 Continuity

Infinitesmal deviations in the represented entity should result in infinitesmal deviations in the representation. This is a critical requirement for us to satisfy because we have been only insisting on the approximate planarity of the joint combinations. When there are small deviations from a plane, we would like the calculated invariants to also differ only by a small amount.

We establish the continuity requirement for one invariant, $I_1$ and for a small deviation in one point. Establishing the requirement for $I_2$ and for deviations in the other points can be done in the same fashion. We assume that the points are in general position and the invariants are finite to begin with. Let $\mathbf{X_i}$ denote the world coordinates of the $i$th point. Let us assume that the fifth point, $\mathbf{X_5}$ has deviated from a plane by a small amount, $\mathbf{\Delta X_5}$. Let $I_1$ be the value of the invariant with the fifth point $\mathbf{X_5}'$ taken as the point of normal on the plane spanned by the other four points from $\mathbf{X_5}$. Let $\mathbf{T}$ be the world-to-image transform such that $\mathbf{x_5} = \mathbf{TX_5}$. Setting $\mathbf{X_5}$ as $\mathbf{X_5}' + \mathbf{\Delta X_5}$, after algebraic manipulation and dropping higher order terms, we obtain the deviation in $\mathbf{x_5}$:

$$\Delta x_5 = \left(\frac{1}{\alpha}\right)(\epsilon_x - X_5 \Delta\alpha)$$

$$\Delta y_5 = \left(\frac{1}{\alpha}\right)(\epsilon_y - Y_5 \Delta\alpha)$$

where

$$\epsilon_x = T_{11}\Delta X_5 + T_{12}\Delta Y_5 + T_{13}\Delta Z_5,$$
$$\epsilon_y = T_{21}\Delta X_5 + T_{22}\Delta Y_5 + T_{23}\Delta Z_5,$$
$$\Delta\alpha = T_{31}\Delta X_5 + T_{32}\Delta Y_5 + T_{33}\Delta Z_5,$$
$$\alpha = T_{31}X_5 + T_{32}Y_5 + T_{33}Z_5$$

Hence, assuming that the coordinate positions are finite to begin with, it can be seen that the deviation in the fifth image point is small for a small deviation of the fifth world point.

If $\Delta I_1$ is the deviation in $I_1$, we have:

$$\Delta I_1 = \frac{\partial I_1}{\partial x_5}\Delta x_5 + \frac{\partial I_1}{\partial y_5}\Delta y_5$$

Substituting $\partial M_{ijk}/\partial x_i = y_j - y_k$ and $\partial M_{ijk}/\partial y_i = -x_j + x_k$ and using the chain rule, after simplification, we obtain the following:

$$\Delta I_1 = I_1 \left[ \left( \frac{y_3 - y_2}{M_{532}} + \frac{y_1 - y_2}{M_{521}} \right) \Delta x_5 + \left( \frac{x_2 - x_3}{M_{532}} + \frac{x_1 - x_2}{M_{521}} \right) \Delta y_5 \right]$$

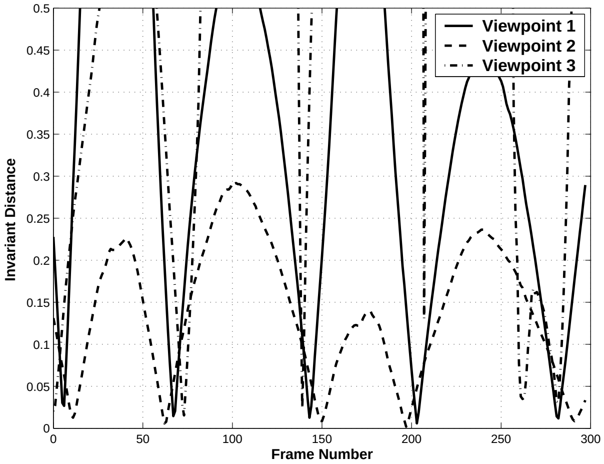


Figure 3.3: Euclidean Norm of Invariant Distances for $C_1$

Hence, if the invariant value is finite to begin with, we see that it varies infinitesimally for small deviations from a plane. This result justifies the use of thresholding for determining if the body is in a particular canonical pose. However, this is a qualified justification because although infinitesimal, the deviations in the invariant values $(\Delta I_1, \Delta I_2)$ are dependent upon the viewpoint $T$. In the extreme case, if the camera is very close to the subject, the denominator $\alpha$ will be small and the deviations will be large. However, for moderately distant camera positions, we are able to get reasonably good results with only a single threshold for all viewpoints. Figure 3.3 shows the Euclidean distances between the *walking* $C_1$ invariants and the area-cross-ratios against each frame for a walking subject. The distances have been plotted for three different viewpoints: a side-view, a front-view and a top-view (note that these are viewpoints 1, 3 and 5 in figure 3.15 which we shall describe later in this chapter). The data is for the same action sequence shown in figure 3.1. It can be observed from the data that while there is no correlation between the area-cross-ratios for frames where the body is not in the $C_1$ pose, all the curves meet when the body *is* in the $C_1$ pose. Note also that the correlating frames correspond to the zero crossings in figure 3.1. For all the viewpoints, the distance between the area-cross-ratios and the invariants is close to zero when the body is in the $C_1$ pose.

### 3.3.4 Uniqueness

A representation should be unique, i.e. one-to-one. Our representation does not satisfy the uniqueness requirement on two counts. Firstly, joint projections of different body configurations could potentially be identical if the respective joints happen to lie along the same line of sight from the camera. This may not always be possible because of integrity constraints on the body (e.g. fixed limb-lengths, joint angle limits etc.), but such ‘spurious’ poses can occur. Secondly, only five joints are used in representing each canonical pose. The other joints are effectively abstracted out and this could pose a problem because the body could be in a different pose of a different action than the one to which it is classified if the ‘abstracted-in’ joints are in the same pose while the ‘abstracted-out’ joints are in a different pose.

### 3.3.5 Fixing Non-Uniqueness

There are two strategies for reducing the probability that spurious poses will be detected. First, we can represent a canonical pose by more than one five-tuple of joints with the observation that the likelihood that two sets of five joints are both in spurious position simultaneously is small. Secondly, we can exploit planarity of joints spanning several frames - we already noted that certain limbs trace areas on a plane during several actions. We can consider the temporal evolution of the invariants formed by five such joints at each phase and make that part of the representation of an action, with the observation that while a different action can result in matches at a few phases, only the correct action will match at all phases. Finding five joints in a plane for all the phases of an action is quite difficult. Considering walking once again, the left/right arms and legs trace four different planes and on each plane, there are atmost three joints we can identify - shoulder-elbow-hand and hip-knee-foot. We solve this problem by first identifying a range of frames delineated by alternate occurrences of the canonical pose $C_1$ (see figure 3.4). Two joints (say the right-hip and right-foot) for the start and end frames give us four points. The two invariants formed by a fifth moving point (say the right-knee) will trace two trajectories which we call invariance space trajectories (ISTs). We thus obtain the five points necessary for computing two invariants at each phase and still need to identify only three joints in each phase. The use of ISTs is advantageous for another important aspect: Recall from our continuity analysis that the closer the joints are to a plane, the more accurate is the representation. The overall distances between the points for which we calculate invariants in an IST, are larger

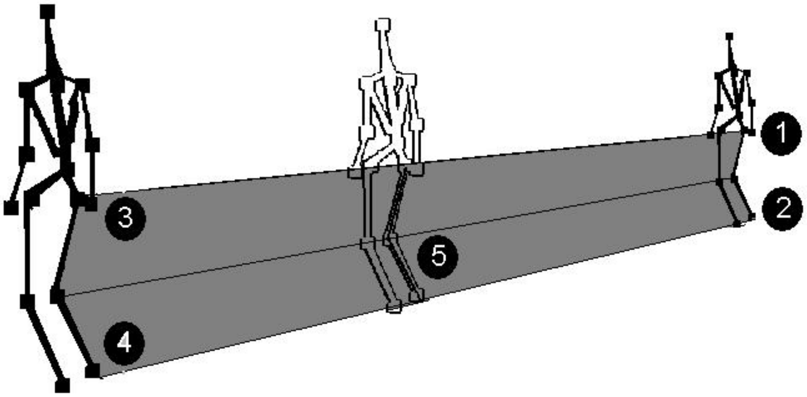


Figure 3.4: ISTs

than distances between points in a single frame and so the planarity of the five points in an IST is better. Other nice properties that ISTs buy us are independence from the frame rate and speed of the action, because the starting and ending instants (and hence, duration) of the ISTs are not fixed upfront. Rather, they are event driven (i.e. determined by the occurrence of specific canonical poses). Note that ISTs can be suitably employed for other actions such as running, sitting-down, waving, etc.

### 3.3.6 Variability

Zatsiorsky [83] describes different models of the human body suitable for geometric analysis. The simplest *isometry* model assumes that body parts of different subjects, when scaled by a reference body part (i.e. the relative body proportions) will be subject-invariant. The *allometry* model assumes that relative body proportions vary with overall body size and is considered a better approximation. For instance, children have a proportionally bigger head than adults. In addition the proportions vary by *affine* transformations [83]. Our representation $(I_1, I_2)$ is invariant to projective transformations of the considered joints (which includes affine transformations) but due to allometry, there will be variations in these values across subjects. For the same reason, ISTs will also exhibit some variability. These variabilities, though, are quite small and in section 3.5.3 we present empirical evidence of this fact. We deal with these variabilities by calculating the empirical medians $(\mu_{I_1}, \mu_{I_2})$ and standard deviations $(\sigma_{I_1}, \sigma_{I_2})$ of the invariants from motion capture data, and use a normalized distance measure as follows:

$$d = \sqrt{\left(\frac{I_1 - \mu_{I_1}}{\sigma_{I_1}}\right)^2 + \left(\frac{I_2 - \mu_{I_2}}{\sigma_{I_2}}\right)^2} \tag{3.2}$$

### 3.3.7 Using Invariants For Classification

Cross-ratios are not uniformly distributed over the line or the plane and there has been some research in understanding and formulating decision rules that account for the non-uniformity. Maybank in [43] and Astrom and Morin in [5] have independently derived expressions for the probability distribution of the 1D cross-ratio and shown that there are logarithmic singularities at 0 and 1. Astrom and Morin have suggested the use of the cumulative distribution to weight two invariants before comparing them to eliminate any bias introduced due to the non-uniform distribution of the cross-ratio. Maybank [43], suggests a simple classification rule based on one threshold $t$ and the standard deviation of image-space noise $n$:

$$\begin{aligned} \text{Accept if} \quad & \|\sigma - \tau\| > ntu \\ \text{Reject if} \quad & \|\sigma - \tau\| \leq ntu \end{aligned} \tag{3.3}$$

where $\sigma$ is the model cross-ratio, $\tau$ is the measured cross-ratio from the image and $u$ is the gradient of the cross-ratio w.r.t. the image coordinates. He shows that a straightforward application of this rule for object classification will likely fail because of the large number of false matches : the system will essentially 'hallucinate' the presence of an object given a set of random feature-points. This is indeed a significant limitation of a projective-invariance based approach which we need to work around, and the problem it presents is more acute for our case compared to Maybank's application domain of 1D object classification. In our case, we have additional problems caused by errors arising from the non-planarity of the body joints and non-uniqueness. We address this problem using the following three heuristics that impose spatial and temporal coherence constraints on a candidate action-instance, before it can be classified as a model action:

1. Clearly, a simple thresholding rule suggested by Maybank will not work because when the body is in a pose close to the canonical pose, the difference in the invariants will be small. For example, in figure 3.3, if one were to choose a threshold of 0.5, a large number of frames would satisfy the threshold. A straightforward way of dealing with this problem is to impose an additional constraint that the distance be a local minimum. With this constraint, in this example, we would perform non-minima suppression and pick only five frames which indeed correctly correspond to the $C_1$ pose.

2. The use of one or more ISTs provides for a temporal coherency constraint: while there may be random matches at a few frames of the IST, the chances that there is a random match for every frame in an IST will be small and this is indeed borne out by our experiments.

3. Representing a pose by the invariants of multiple five-joint sets provides for a measure of robustness against random mis-matches : while one five-joint tuple may match spuriously, the likelihood of all five-joint tuples matching is small.

### 3.3.8 Dynamic Programming on the ISTs

From an empirical analysis of motion capture sequences, we found that the ISTs exhibit minor variabilities when an action is performed repeatedly by the same subject and also when the same action is performed by different subjects. Figure 3.5 shows the $I_1$ walking IST for one subject performing the walk-cycle action five times while figure 3.6 shows the $I_1$ walking IST for five different subjects. It can be observed that the ISTs are fairly similar but there are minor variabilities, both, in the values of the invariant as well as its temporal characteristics. We employed dynamic programming

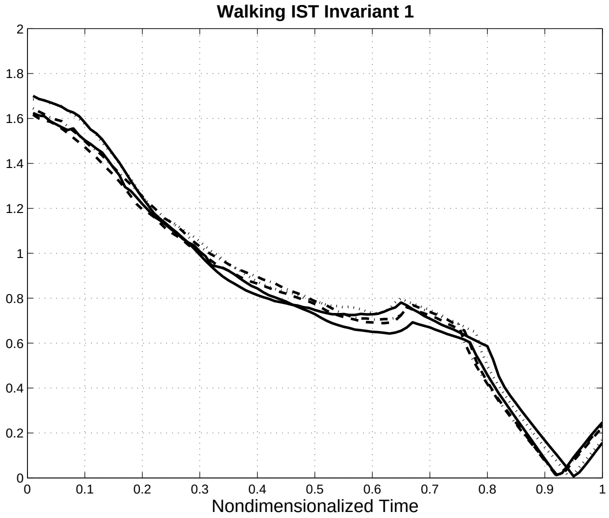
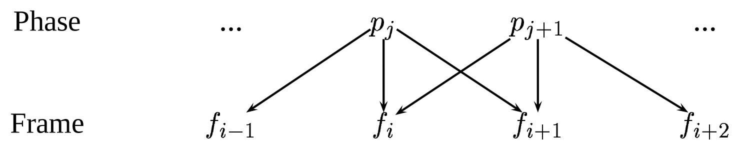


Figure 3.5: Walk-cycle IST, same subject, five instances

[16] for matching a given trajectory of invariants with an IST which we describe briefly below.

Let **F** be a sequence of frames containing area-cross-ratios which we would like to match with **P** the set of IST invariants. Let $f_i \in \mathbf{F}$ and $p_j \in \mathbf{P}$ where $i = 1..N$, $j = 1..M$. In other words, $N$ is the number of frames and $M$ is the number of phases. We seek the optimal mapping $\psi : \mathbf{P} \to \mathbf{F}$ such that the cumulative distance between the invariant values of each frame with the phase that it is mapped to, namely $\sum_{j=1}^{M} d(p_j, \psi(p_j))$ is minimized. We have the boundary condition $\psi(p_1) = f_1$ and $\psi(p_M) = f_N$ because the ISTs are delineated by canonical poses as explained earlier and the mappings of the ends are fixed.



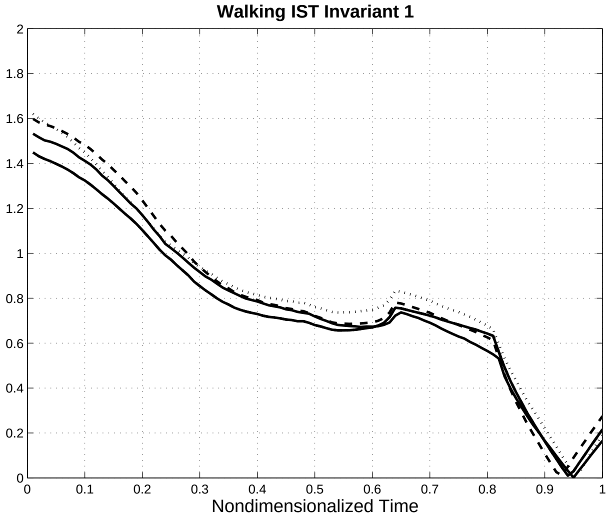


Figure 3.6: Walk-cycle IST, same action, five subjects

For notational convenience, we think of $\psi$ as mapping the phase number $j$ to a frame number $i$. We first calculate a linear form for $\psi$ (call it $\psi_0$):

$$\psi_0(j) = 1 + \left(\frac{N-1}{M-1}\right)(j-1) \tag{3.4}$$

Our use of dynamic programming now is to enable us to pick the optimal value within a window $w$ on either side of $\psi_0$. We have

$$\psi_{min}(j) \quad \leq \quad \psi \quad \leq \quad \psi_{max}(j) \tag{3.5}$$

where

$$\begin{aligned} \psi_{min}(j) &= max(2, (\psi_0(j) - wN)) \\ \psi_{max}(j) &= min(N-1, (\psi_0(j) + wN)) \end{aligned}$$

Proceeding from phase no. 1 towards M, let $D(j,i)$ denote the minimum cumulative distance of the mapping upto phase no. $j$ with phase no. $j$ mapped to frame no. $i$. We can then write down the following dynamic programming recurrence relation:

$$D(j+1,k) = d(j+1,k) + \begin{matrix} Min(D(j,m)) \\ [\psi_{min}(j) \leq m \leq min(\psi_{max}(j), k-1)] \end{matrix} \tag{3.6}$$

for all $k$ in the window:

$$\psi_{min}(j+1) \quad \leq \quad k \quad \leq \quad \psi_{max}(j+1)$$

Once the recursion is calculated and the optimizing frame number is found for the last phase, $j = M$, we backtrack to the first phase, $j = 1$ and read off the frame numbers at each phase $j$.

## 3.4 Algorithm Details

We first make precise a few terms that we discussed in earlier sections and which will be used in describing the algorithm. Following that, we describe the algorithm.

### 3.4.1 Definitions

A *canonical pose* consists of five ordered joint names, two invariant values and a threshold for matching purposes. The area-cross-ratios (3.1) that we compute from the image coordinates will be close to the two reference invariants when the body is in the canonical pose. We declare a match when the Euclidean norm of the distance between the invariants and the area-cross-ratios is below the threshold and is a local minimum. Besides the canonical pose, we also define a *stationary pose* which consists of one or more joint names, a window-size and a threshold with the following semantics: The stationary pose is declared as detected if the combined motion of the joints within a window of the specified size is below the threshold. We refer to the canonical pose and the stationary pose simply as *pose*. An *IST* consists of two delineating poses determining the start and end of the trajectory, three joint names and flags indicating whether the joints are fixed or moving (recall the discussion in section 3.3.5), two sequences of invariant values and a threshold for matching. We represent the two sequences at discrete points in non-dimensionalized time that runs from 0 to 1. Given a trajectory of observed area-cross-ratios, dynamic programming is carried out that maps each point in the trajectory to an optimum point on the IST and if the average invariant distance per point of the trajectory is smaller than the threshold, the trajectory is declared as matched to the IST. An *action model* consists of canonical poses or stationary poses and one or more ISTs.

### 3.4.2 Algorithm

The input to the algorithm is an action model $A$ and a sequence of frames $\mathbf{F}$. Rather than an image, in our case, each frame $f \in \mathbf{F}$ consists of body joint names and their 2D image coordinates. The output is a sequence of the structure $(s, e, d)$ with the following semantics: the action $A$ was found to occur starting at frame no. $s$ and ending at frame no. $e$ with an overall distance of $d$. The smaller $d$ is, the better the match. The following is the pseudocode for the action recognition algorithm:

**Recognize(Action $A$, Frame $\{\mathbf{F}\}$)**

```
Initialize list $L$ to $\{\}$
for each pose $p \in A$
  Initialize list $l$ to $\{\}$
  for each frame $f \in \{\mathbf{F}\}$
    Compute Euclidean norm $n$ of invariant distances
    if ($n$ < $p$.threshold)
      push ($f, n$) into $l$;
    end if
  end for
```

```
    smooth $l$ based on field $n$
    $l \leftarrow$ local minima of $l$ based on field $n$
end for
for frame numbers $(s,e) \in l$ where $s < e$
    boolean detected=TRUE
    $d \leftarrow 0$
    for each IST $i \in A$ with delineators $s$ and $e$
        calculate area-cross-ratios between $(s,e)$
        get optimal distance $n$ by dynamic programming
        if ($n$ > $i$.threshold)
            detected=FALSE;
            break;
        end if
        $d \leftarrow d + n$
    end for
    if (detected)
        Push ($s$,$e$,$d$) into $L$
    end if
end for
output $L$
```

## 3.5 Creating an Action Model Database

### 3.5.1 Data Acquisition

We obtained human motion capture data from public and commercial sources. The data sets included several subjects walking, running and sitting-down. These datasets were in different formats, including the BioVision Hierarchy (BVH) format (from Credo Interactive [33], a commercial source), Acclaim format (from the Carnegie Mellon University public domain motion capture repository [73]) and in raw format where the 3D coordinates of each joint were listed per frame (from the Georgia Institute of Technology public domain motion capture repository [53]). We decomposed each motion capture sequence into a person part and an action part. The person part consisted of the skeletal model of the person while the action part consisted of the joint angles of the action. This way, we were able to mix subjects and actions from different sources creating more motion capture sequences. The body model we employed consisted of fifteen joints (see figure 3.7) namely the head, hip, chest, shoulders, elbows, hands, hips, knees and feet.

### 3.5.2 Model Building

We built models for four actions: *walk-cycle*, *run-cycle*, *sit-down* and *forward-jump*. The walk-cycle and run-cycle actions are cyclical in that any particular phase of the actions can become the starting pose of the action. We arbitrarily define the starting (and ending) pose of the actions as that where the body is in the $C_1$ pose and where the left foot just passes by the right foot (see figure 3.8). For sit-down, we define the starting pose of the action where the feet become stationary in preparation for sitting down and the ending pose where the subject is finally seated. For forward-jump, we define

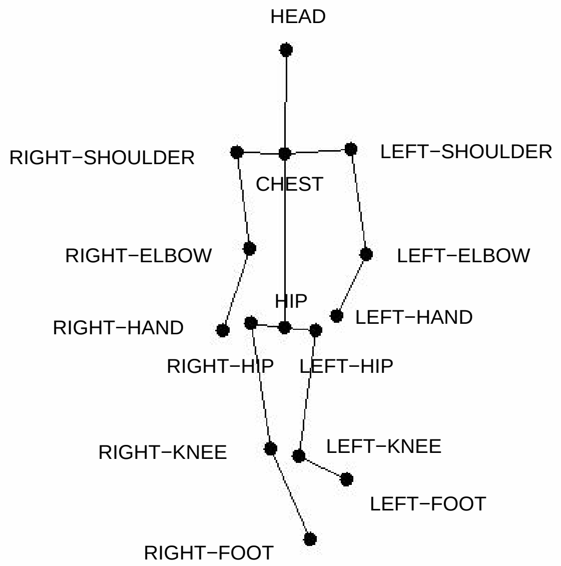


Figure 3.7: Body Model

the start of the action to be the pose when the feet are stationary and just about to move upwards and the end of the action when the feet become stationary again (see figure 3.9). Model building was done in an automatic way with manual supervision only to verify correctness of the model. For walking and running, a planarity assessment program was run through one instance of each subject (12 subjects total) giving frames where the joints of interest for $C_1$ were on a plane. In our case, specifically, these joints were the head, right-shoulder, left-shoulder, left-foot, and right-foot for both walking and running. An invariant calculation program was run on those specific frames where the joints were closest to being planar, and the average invariant values were obtained. These frames became delineators for the ISTs. Next, invariance space trajectories were extracted for frames between these delineator frames and median values of the invariants for each intervening frame were obtained. We used two ISTs each to model

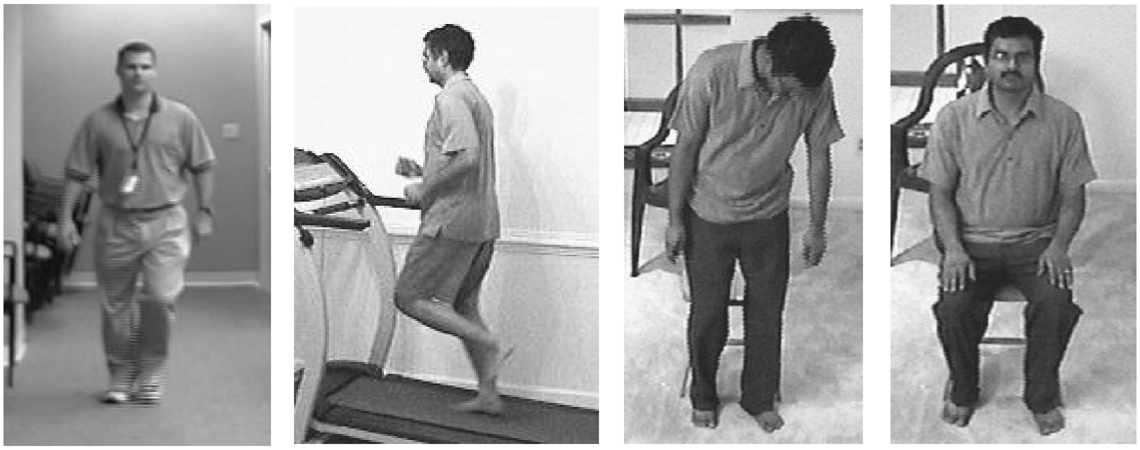


Figure 3.8: Start pose for walk-cycle, run-cycle, sit-down and end-pose for sit-down

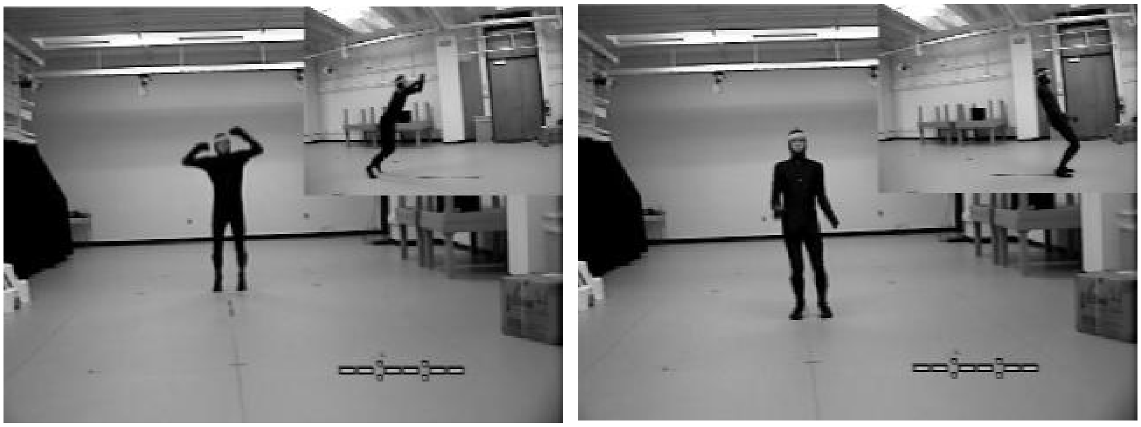


Figure 3.9: Forward-jump starting and ending poses, images from [73]

walking and running. The specific joints used were the same for both the actions and they were {left-hip, left-foot, left-knee} and {right-hip, right-foot, right-knee}. Our choice of using the leg joints for the ISTs for walking and running was motivated by the fact that the motion of the legs exhibited more planarity and less variability than the motion of the arms. The two ISTs along with the $C_1$ invariant values for each action completed our action-model for the walking and running actions ( Figure 3.10 shows the ISTs). For sit-down a stationarity assessment program was run first only on the feet and then on the feet and hip. These helped us to determine the delineator poses for the action. We used two ISTs for representing the sit-down action. The specific joints chosen were $\{$left-foot, hip, head$\}$ and $\{$right-foot, hip, head$\}$. We found that the trajectories formed by the elbows and hands exhibited a lot more variation than those formed by the knees and feet: at the end of the sit-down action, some subjects chose to rest their arms on the handles of the chair while others rested their hands on their upper legs. By using the feet, hip and the head for the IST, we effectively abstract out these variations. Figure 3.11 shows the left-side IST for the sit-down action.

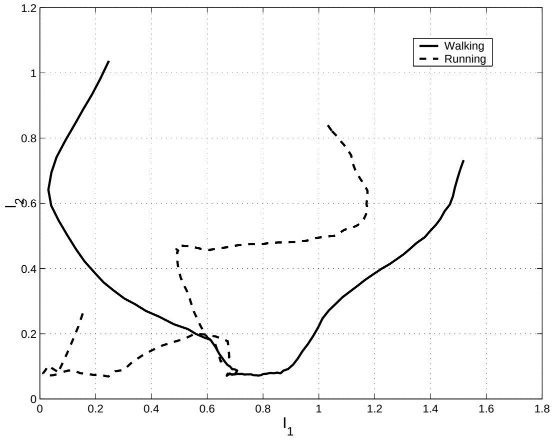


Figure 3.10: Median ISTs for Walk-cycle and Run-cycle

For forward-jump, we ran a stationarity assessment program for both the feet and obtained the start and end as consecutive minima. For the ISTs, we used the same set of joints used for walking and running. The left and right sides are symmetric and yield the same IST. However, both ISTs are used to increase robustness. Figure 3.12 shows the IST. The slope-discontinuity seen at approximately $(0.1, 0.05)$ corresponds to the pose where the body lands and the feet begin to stabilize. Note that $t = 0$ corresponds to approximately $(1.66, 0.42)$.

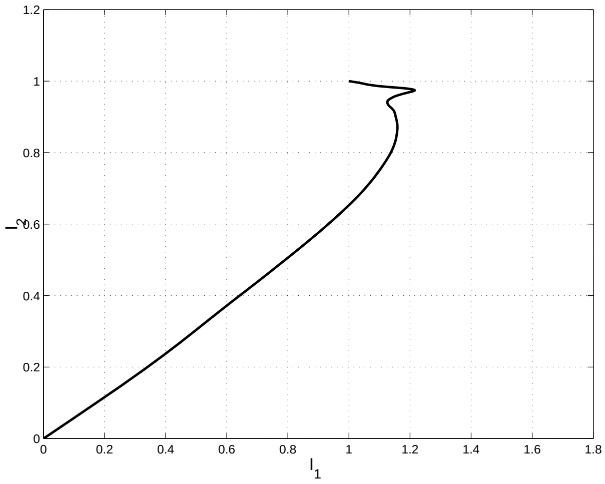


Figure 3.11: IST for Sit-Down

### 3.5.3 Variability and Distinguishability of the Invariant Values

For the specific joint combinations that we used to model actions, the inter and intra-subject variability in the invariants was quite small. Figures 3.13 and 3.14 show the probability distribution for the walking and running $C_1$ invariant values for 1745 walk-cycles and 1980 run-cycles respectively from 12 different subjects and 5 different viewpoints each. The variability was also an artifact resulting from inconsistencies in the placement of sensors on the motion capture subjects because these were obtained from different sources. For example from one source, the 'shoulder' sensor was called 'upper arm' and was placed slightly below the shoulder position. Similarly the 'knee' sensor was placed slightly below the real knee and was called 'lower leg'. We compensated for these inconsistencies somewhat by adding or subtracting vectors of small magnitudes from the given positions and visually verifying the positions by rendering the data. Inspite of this, it was inevitable that some artificial variation still remained. Nevertheless, the probability distribution shows that the overall variability in the invariants is quite small. We found that the median values of $I_1$ and $I_2$ for the

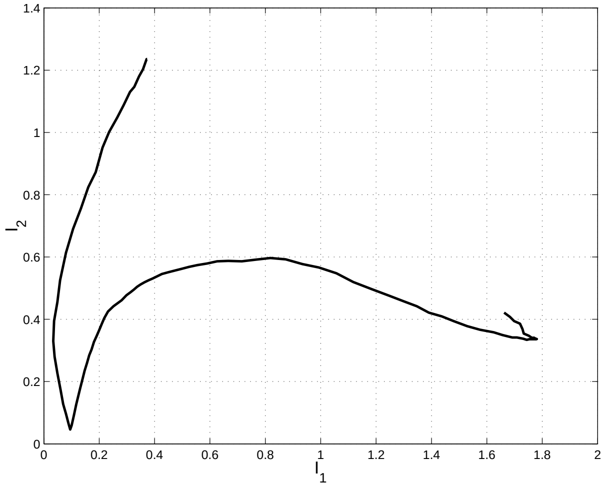


Figure 3.12: IST for Forward-Jump

$C_1$ pose for walking were $1.0489$ and $1.1005$. The corresponding values for running were $1.0701$ and $1.1049$. Clearly, the differences between the values for walking and running are not different enough to enable reliable distinction between the two actions. However, the ISTs were markedly different. Figure 3.10 shows the the median left side ISTs for walking and running (the two invariants formed by the moving left-knee are plotted against each other) and both are sufficiently different to enable reliable distinction between the two actions.

## 3.6 Results

We evaluated our algorithm on two modalities of input : arbitrary projections of motion capture data and manually segmented real image sequences. A set of unknown actions observed from different viewpoints and performed by different subjects at different speeds were input to the action recognition system. These actions were continuous, meaning that each sequence had one or more instances of an action being performed, and the starting and ending times of the action were unknown (to the system). The

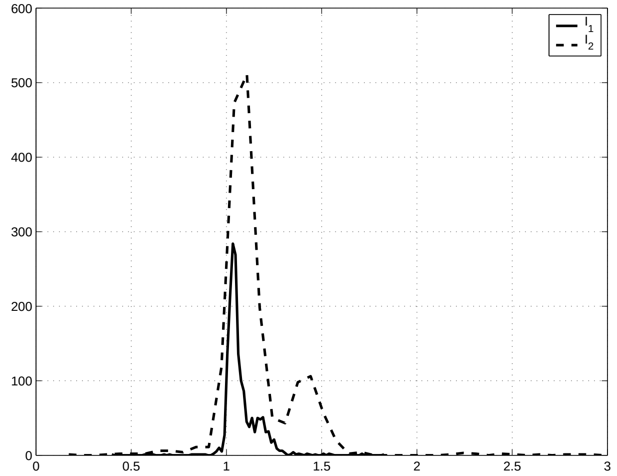


Figure 3.13: Probability Distribution of $C_1$ invariants for Walking

system not only had to correctly classify the actions but also detect where in the sequence each action was found to occur. There were three metrics that we evaluated the algorithm against for both modalities:

1. *The true detection rate* defined as the ratio of correct detections to the expected number of detections. An action is deemed correctly detected if the start and ending positions are detected within 25% of the correct starting and ending positions (as a fraction of the ground-truthed action length).

2. *The false alarm rate* defined as the ratio of falsely detected actions to the total number of detected actions.

3. *The misclassification rate* defined as the ratio of incorrectly classified actions (e.g. a walk-cycle was classified as a run-cycle) to the total number of detections.

The same three action models (i.e. the invariants and thresholds) were used for the motion capture sequences as well as the real image sequences. We present results on these two modalities in this section.

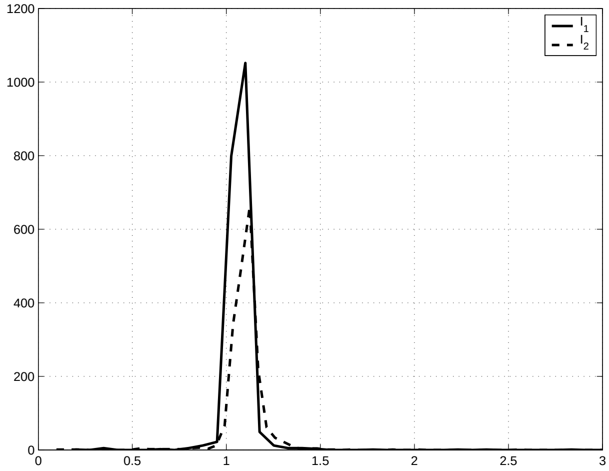


Figure 3.14: Probability Distribution of $C_1$ invariants for Running

### 3.6.1 Results on 2d Projections of Motion Capture Data

The dataset included 25 walking sequences, 23 running sequences, 18 sit-down sequences and 8 forward-jump sequences performed by 12 different subjects. Each walking and running sequence included one to three complete cycles of the respective action. Each sit-down and forward-jump sequence had one instance of each action. The sequences collectively made up a total of 1200 action instances that were to be detected by the algorithm for each viewpoint. We chose five different viewpoints to

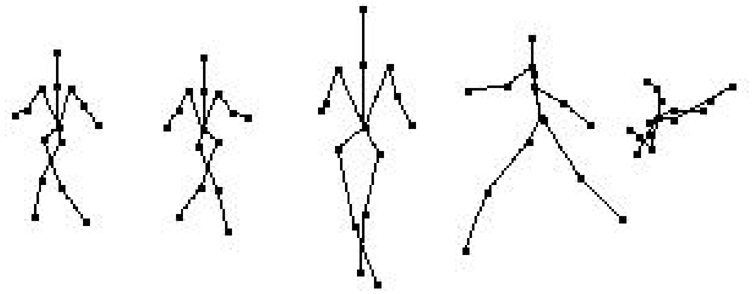


Figure 3.15: Walk seq. 20, frame no. 100, all viewpoints

test the performance of the algorithm on 2d projections of motion capture data. Figure 3.15 shows the same walking frame as seen from each viewpoint. Note that there was only one action model employed for all the viewpoints. For each action instance, the algorithm first computed the matching score (distance) for each of the four action-models and chose the model that resulted in the best score (minimum distance). This was evaluated against the known ground-truth. Table 3.1 summarizes the action classification results. It can be seen that the performance of the algorithm for viewpoints $3$ and $5$ are the worse than the other viewpoints. Viewpoint 3 is a frontal view of the subjects while viewpoint 5 is a top view. From both views, several key joints of the subject are coincident or very close to each other. ISTs happen to represent the sideways movement of the body for all four actions. Although the representation is view-invariant, the frontal and side views are near 'end-on' views of the IST, making them more susceptible to error. Further, from the frontal viewpoint, the area-cross-ratios for the $C_1$ pose of walking and running do not exhibit much temporal variation because the foot does not rise much above the floor. In many instances, this resulted

Table 3.1: Motion Capture Results, 1200 Total Actions

| Metric | Viewpt. 1 | Viewpt. 2 | Viewpt. 3 | Viewpt. 4 | Viewpt. 5 |
|---|---:|---:|---:|---:|---:|
| True Detections | 1145 | 1126 | 1057 | 1105 | 1012 |
| False alarms | 73 | 75 | 138 | 42 | 68 |
| Misclass. | 145 | 108 | 171 | 154 | 185 |
| True Det. % | 95.42 | 93.83 | 88.08 | 92.08 | 84.33 |
| False Alarm % | 5.35 | 5.72 | 10.04 | 3.22 | 5.37 |
| Misclass. % | 10.64 | 8.24 | 12.44 | 11.84 | 14.62 |

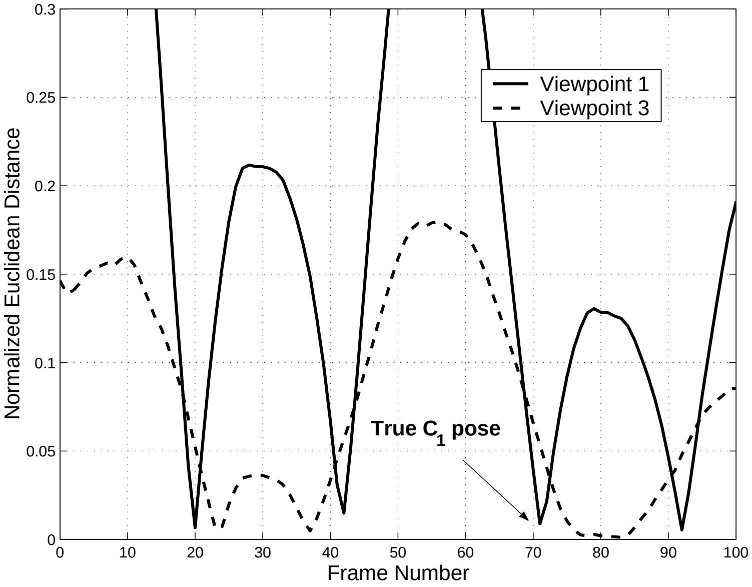


Figure 3.16: Incorrect detection of $C_1$ pose for a walking sequence from viewpoint 3

in the $C_1$ pose being detected at spurious locations which increased the misses as well as false alarms Figure 3.16 shows an example of this: the figure shows the normalized Euclidean distance between the area-cross-ratios and the reference invariants for the $C_1$ pose for a walking sequence as seen from viewpoints 1 and 3. From the ground truth, a walk-cycle is expected to be detected between frames 18 and 68. While the start is detected reasonably accurately from both viewpoints, the ending is detected correctly only from viewpoint 1. The ending is detected at a spurious location from viewpoint 3. Another particular problem was that for the forward-jump action, the subjects crouched before jumping, mimicking a sit-down action. Similarly, upon landing after the jumping, they assumed a similar crouching posture. Both of these were detected as sit-down actions by the system, increasing the false alarm rate. Yet another particular problem with the approach was the performance when the person is relatively still as was the case at the end of all of the forward jump and sit-down actions. Still, there are small joint motions that trigger the algorithm to detect local minima at spurious locations, inspite of temporal smoothing. Adding a heuristic stipulation that the body translate a minimum amount for the motion to be considered as a candidate action, while not a theoretically correct one because the translation threshold will depend upon the viewpoint, was nevertheless incorporated into the recognition system to weed out many false matches. Overall, it can be observed that the detection results do not vary too substantially with viewpoint and that the results are quite good.

### 3.6.2 Real Image Sequences

We obtained videos of four different subjects walking, running and sitting down. These videos were shot from the front, sides and the top. In all, we chose a total of 40 action instances for the evaluation. A small graphical user interface program allowed the manual location of the 15 body joints for any given frame. It was not necessary to mark the locations in every frame. Rather, every alternate frame was marked and cubic spline interpolation was used to map the entire sequence into the interval $[0, 100]$. There were self-occlusions in almost every sequence and a guess was made as to the occluded joint positions wherever possible based on their past and future trajectories. Thus, it was not possible to be very accurate in the picking of joint locations, with the result that they exhibited significant spatial and temporal jitter. This somewhat simulated (albeit incidentally and not rigorously) small errors in the joint locations. Temporal smoothing was employed to reduce the effect of noise.

#### Sample frames and Invariant Distances

Figure 3.17 shows a sample frame of a walking sequence containing one walk-cycle, shot from a front view along with the marked joints and skeleton. Beside the frame is a plot showing the distance between the area-cross-ratios and the invariants for the

$C_1$ pose. Note that similar to the motion capture sequences, the invariant distance approaches zero thrice, reflecting the fact that the body assumes the $C_1$ pose thrice in a walk-cycle. Similarly, figures 3.18 and 3.19 show a sample frame and $C_1$ invariant norms for a side view and top-view walking sequence. With regard to the top-view, although the curves exhibit several spurious local minima, the ISTs were able to eliminate all of them except the correct local minima corresponding to the frames where the $C_1$ pose was truly attained. This is a good example of how the first and third heuristics outlined in section 3.3.7 can be used to achieve some robustness in the solution.

Figures 3.20, 3.21 and 3.22 show a sample frame and $C_1$ invariant norms for a sample side-view, front-view and top-view running sequence. As in the walking sequences, the invariant norms approach zero periodically reflecting the fact that the body assumes the $C_1$ pose periodically. Figure 3.23 shows a sample frame of a top-view and side-view sit-down sequence that were used for the evaluation.

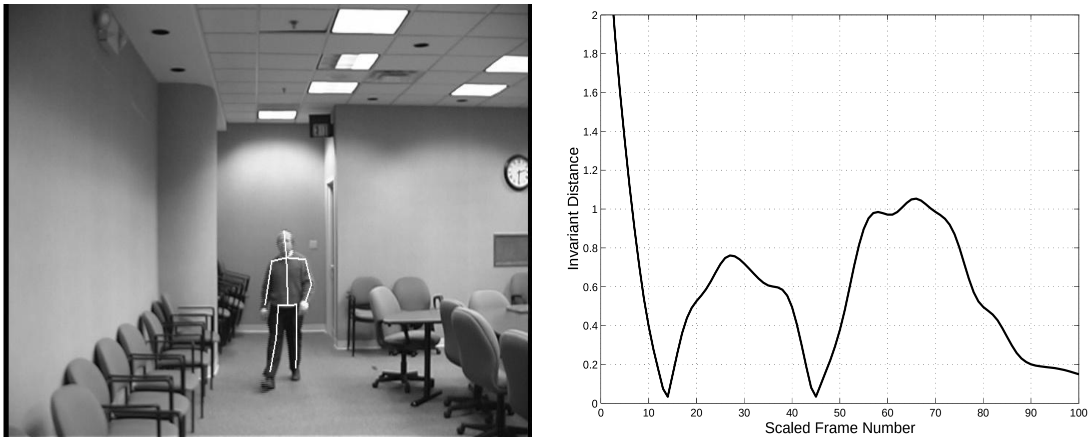


Figure 3.17: Front-view walking sequence and $C_1$ invariant norm

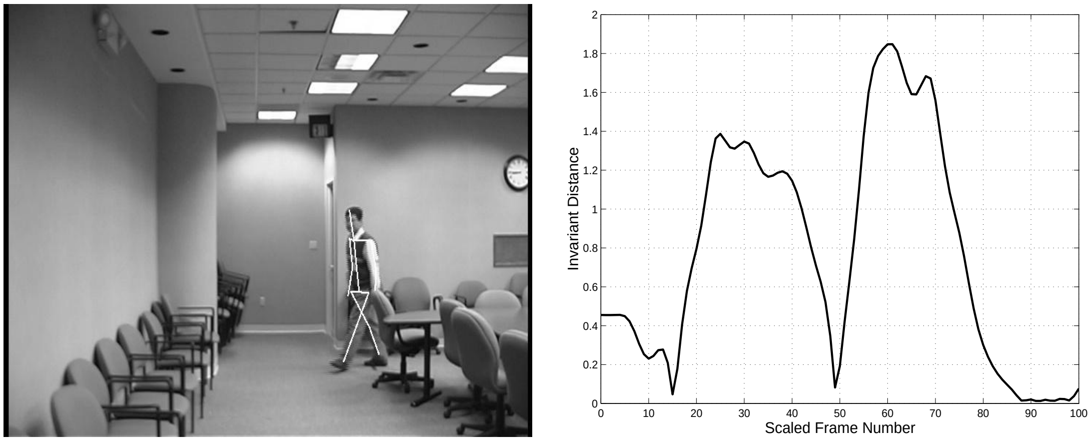


Figure 3.18: Side-view walking sequence and $C_1$ invariant norm

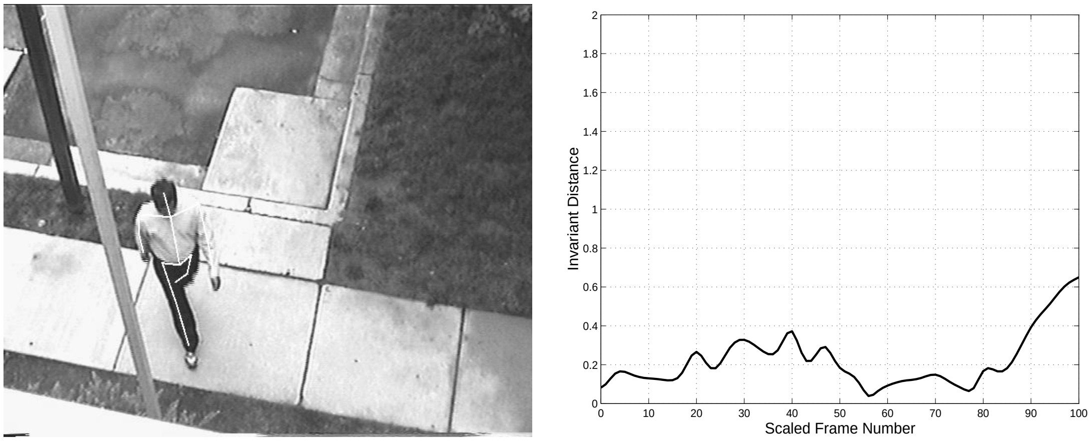


Figure 3.19: Top-view walking sequence and $C_1$ invariant norm

**Summary of Results**

The table below shows a summary of results. Intuitively, for the real image sequences, one would expect that the results would be somewhat inferior to those obtained from the motion capture sequences because of additional errors in the image locations of the joints. While this was found to be true for the front and side views, surprisingly, the top-view sequences were detected correctly. It is to be noted that while it was easy enough to simulate a viewpoint directly above the subject for the motion capture sequences (viewpoint no. 5), it was not possible to shoot a similar real image

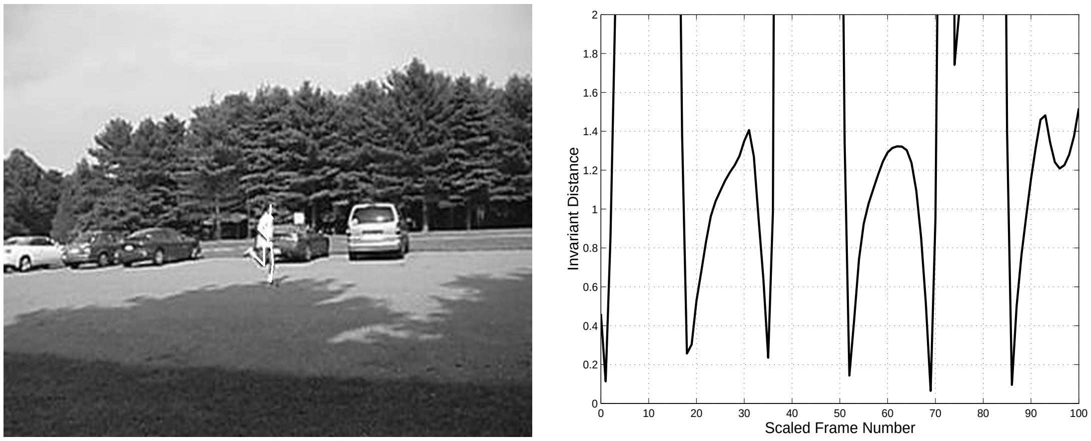


Figure 3.20: Side-view running sequence and $C_1$ invariant norm

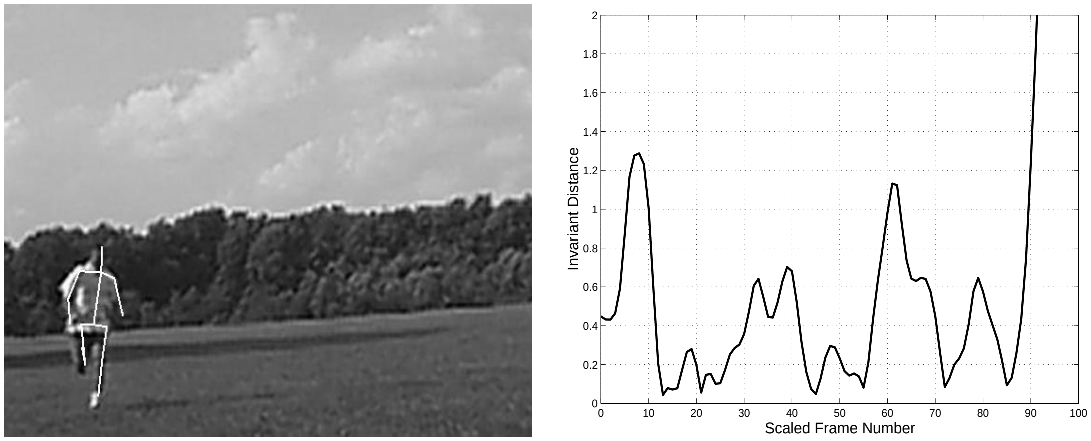


Figure 3.21: Front-view running sequence and $C_1$ invariant norm

sequence. Hence the ‘top’ view for the real image sequence was slightly different than the top view for the motion capture sequences. The front view had more misclassifications than the other viewpoints and they occured because two walking sequences were classified as running. The number of real image sequences used for the performance evaluation is not high enough to enable us to draw definite conclusions about the nature of the failures and the differences in performance between different viewpoints. The primary purpose of the study was to demonstrate the applicability of the approach to real image sequences and it can be seen that the overall performance is quite good

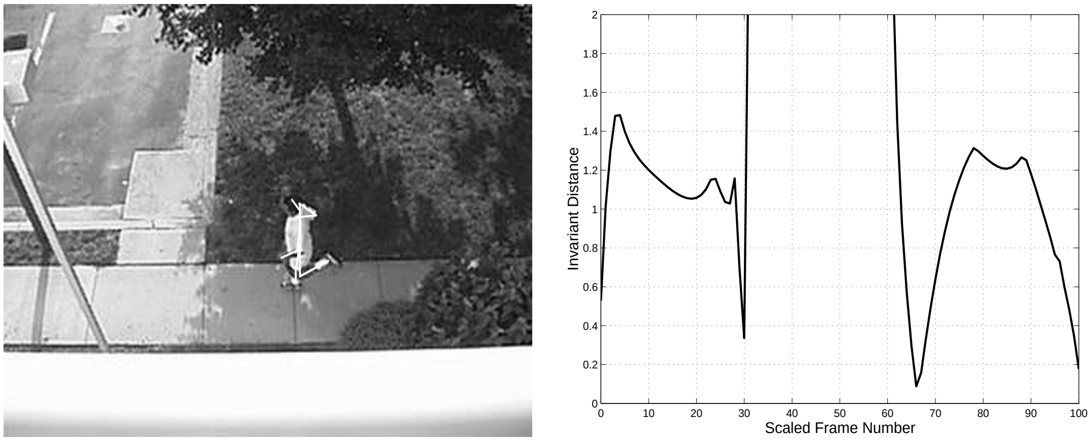


Figure 3.22: Top-view running sequence and $C_1$ invariant norm

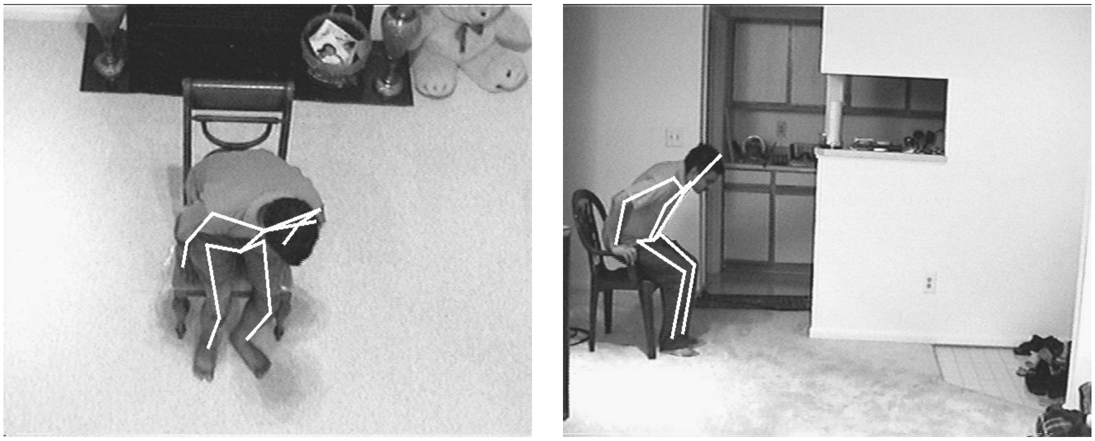


Figure 3.23: Sample frames for top-view and side-view sit-down sequences

inspite of the noisy input.

## 3.7 Summary

The key idea presented in this chapter is the exploitation of the geometry of a human action and the use of 2D invariance theory for view-invariant representation and recognition. We modeled actions as static canonical poses and dynamic trajectories in 2D

Table 3.2: Real Image Sequences, 40 Total Actions

| Metric | Front-View | Side-View | Top-View |
|---|---:|---:|---:|
| No. of Sequences | 13 | 14 | 13 |
| True Detections | 11 | 12 | 13 |
| False alarms | 0 | 1 | 0 |
| Misclass. | 2 | 1 | 0 |

invariance space. We evaluated the representation scheme theoretically and showed why a straightforward application of the idea will generate many false alarms. We showed how we could enforce spatial and temporal coherency constraints on the solution to bring down the false alarm rate without sacrificing the detection rate. We evaluated the approach on arbitrary projections of motion capture data and on real image sequences arising from different subjects performing different actions at different speeds and from different viewpoints. Our use of the two modalities of input - 2D projections of motion capture data and manually segmented real image sequences effectively simulated for us, the output of a body-joint detection and tracking module. We demonstrated that a single view-independent representation of an action was sufficient to recognize and distinguish it from other actions with good success on both input modalities and different viewpoints. A weakness of the 2D approach we presented is that while it can be applied successfully on a variety of actions, it is not a completely general solution and will not work for those actions that cannot be decomposed into approximately planar patches. To complement the 2D approach and relax this requirement, we present 3D approaches based on model-based invariants, in the following chapter.

# Chapter 4

# 3D Approaches using Mutual Invariants

## 4.1 Introduction

In this chapter, we overcome a fundamental limitation of 2D approaches and propose, analyze, and evaluate the performance of, two variants of a 3D approach based on mutual invariants. The lack of general case view-invariants for an arbitrary 3D point-cloud was proved in [10]. However, this does not imply that recognizing a 3D object in a view-independent manner is not possible. Rather than searching for quantities computed from image sequences of a human action that are preserved across different views of the action, the mutual invariants approach searches for relationships between quantities derived based on the 3D representation of the action and those derived from image sequences of the action, that are preserved across different views. We start off by describing the theory of mutual invariants in section 4.2 and show how a human action can be modeled using mutual invariants in section 4.3. In section 4.4, we analyze the representation in relation to the same qualitative benchmarks that we used for analyzing our 2D approach from the previous chapter. Model building is discussed in chapter 4.5. In section 4.6, we present results of the approach on the same data that was used to evaluate the 2D approach presented in the previous chapter.

## 4.2 Mutual Invariants

Mutual invariants are quantities derived from a 3D object and its image that satisfy compatibility relationships. Such invariants and their 3D-2D relationships have been derived and used for view-independent 3D object recognition in the past. These are derived by eliminating the unknown parameters relating the world-to-image transform.

Given five world points, $i = 0..4$, we can set $\mathbf{X}_0 = \mathbf{0}$ and $\mathbf{x}_0 = \mathbf{0}$ without loss of generality. For scaled orthographic projection, applicable where the depths of the objects along the line of sight is much smaller than their distances to the camera,

$$\begin{pmatrix} x_i \\ y_i \end{pmatrix} = s \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{11} & r_{12} & r_{13} \\ r_{11} & r_{12} & r_{13} \end{pmatrix} \begin{pmatrix} X_i \\ Y_i \\ Z_i \end{pmatrix} \tag{4.1}$$

where $s$ is the scale and $r_{jk}$ are the elements of the 3D rotation matrix $R$. Weinshall ([75]) showed that by setting

$$\begin{aligned} A &= \begin{pmatrix} X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \\ X_3 & Y_3 & Z_3 \end{pmatrix}, \ \mathbf{p_x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}, \ \mathbf{p_y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} \\ B &= (AA^T)^{-1} \end{aligned}$$

and eliminating the unknown rotation parameters and the scaling parameter produces the following relationship:

$$f_B(\mathbf{p_x}, \mathbf{p_y}) = \mid \mathbf{p_x}^T B \mathbf{p_y} \mid + \mid \mathbf{p_x}^T B \mathbf{p_x} - \mathbf{p_y}^T B \mathbf{p_y} \mid = \ 0 \tag{4.2}$$

that holds regardless of the viewpoint. The method was extended to the affine case to obtain a different relationship.

Similarly, in [68], Stiller et. al. derived camera-parameter independent relationships among five world points on a rigid object and their imaged coordinates for an affine camera. Weiss and Ray in [77] simplified and extended this result to the full-projective case showing that there exists one equation relating six world points and their image coordinates. We proceed along the lines of Weiss and Ray and derive a simpler relationship than theirs after making an approximation. Five points ($\mathbf{X_i}, i = 1..5$ in homogenous coordinates) in 3D projective space cannot be linearly independent. Assuming that the first four points are not all coplanar they will make a 3D projective basis. We can write the 3D coordinates of the fifth point in this basis as follows:

$$\lambda_5 \mathbf{X_5} = a_5 \lambda_1 \mathbf{X_1} + b_5 \lambda_2 \mathbf{X_2} + c_5 \lambda_3 \mathbf{X_3} + d_5 \lambda_4 \mathbf{X_4} \tag{4.3}$$

The $\lambda_i$ are the unknown projective scale factors and $a_5, b_5, c_5, d_5$ are the unknown projective coordinates of the point $\mathbf{X_5}$ in the basis of the first four points.

### 4.2.1 Restricted 3D Case

We would like to model a point configuration where four points lie on the same plane. Given that we need the first four form a basis, we can choose a labeling such that points 1,2,4 and 5 form a plane while points 3 and 6 lie outside this plane[1]. We call this configuration a *restricted 3D* configuration because of the restriction that four points lie on a plane. In this configuration, point 3 doesn't contribute to point 5's coordinates, making $c_5$ zero.

We have for $\mathbf{X_6}$:

$$\lambda_6 \mathbf{X_6} = a_6 \lambda_1 \mathbf{X_1} + b_6 \lambda_2 \mathbf{X_2} + c_6 \lambda_3 \mathbf{X_3} + d_6 \lambda_4 \mathbf{X_4} \tag{4.4}$$

Here $a_6$, $b_6$, $c_6$ and $d_6$ are the basis coordinates for $\mathbf{X_6}$.

[1] We assume the points are in general position and for now, ignore degenerate cases where points 3 or 6 lie on the plane

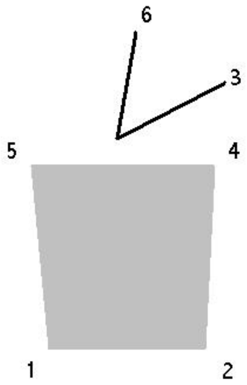


Figure 4.1: Six point configuration used for analysis. Points $1, 2, 4, 5$ form a plane and $3, 6$ lie outside this plane

If $\mathbf{T}$ is the world to image transform such that $\mathbf{x_i} = \lambda_i' \mathbf{T} \mathbf{X_i}$, where $\lambda_i'$ is an unknown scale factor, the image coordinate for the fifth point, $\mathbf{x_5}$ is given by:

$$\begin{aligned}
\mathbf{x_5} &= \lambda_5' \mathbf{T} \mathbf{X_5} \\
&= (\lambda_5'/\lambda_5)\mathbf{T}(a_5\lambda_1\mathbf{X_1} + b_5\lambda_2\mathbf{X_2} + d_5\lambda_4\mathbf{X_4}) \qquad (4.5) \\
&= (\lambda_5'/\lambda_5)(a_5\lambda_1/\lambda_1'\mathbf{x_1} + b_5\lambda_2/\lambda_2'\mathbf{x_2} + d_5\lambda_4/\lambda_4'\mathbf{x_4})
\end{aligned}$$

Writing $\lambda_i/\lambda_i'$ as $\alpha_i$, and repeating the same algebra for point 6, we have the following two equations relating the image coordinates:

$$\begin{aligned}
\alpha_5\mathbf{x_5} &= a_5\alpha_1\mathbf{x_1} + b_5\alpha_2\mathbf{x_2} + d_5\alpha_4\mathbf{x_4} \\
\alpha_6\mathbf{x_6} &= a_6\alpha_1\mathbf{x_1} + b_6\alpha_2\mathbf{x_2} + c_6\alpha_3\mathbf{x_3} + d_6\alpha_4\mathbf{x_4} \qquad (4.6)
\end{aligned}$$

We would like to eliminate the projective coordinates and the scale factors. Let $M_1$ denote the determinant $[\mathbf{X_2}\mathbf{X_3}\mathbf{X_4}\mathbf{X_5}]$ (the notation is such that we index $M$ by the point left out from the five-point set $(\mathbf{X_1}, \mathbf{X_2}, \mathbf{X_3}, \mathbf{X_4}, \mathbf{X_5})$. Substituting for $\mathbf{X_5}$ from

(4.3) and noting that determinants with two equal columns vanish we have:

$$M_1 = |\ \mathbf{X_2 X_3 X_4}(a_5\lambda_1\mathbf{X_1} + b_5\lambda_2\mathbf{X_2} + d_5\lambda_4\mathbf{X_4})/\lambda_5\ |$$
$$= -a_5\frac{\lambda_1}{\lambda_5}M_5 \tag{4.7}$$

We similarly have the following two relationships as well:

$$M_2 = b_5\frac{\lambda_2}{\lambda_5}M_5\ ,\ M_4 = d_5\frac{\lambda_4}{\lambda_5}M_5$$

For the point $\mathbf{X_6}$ we use $M_1'$ to denote the determinant $[\mathbf{X_2X_3X_4X_6}]$. The notation is such that the index is that of the point left out in the five-point set $(\mathbf{X_1}, \mathbf{X_2}, \mathbf{X_3}, \mathbf{X_4}, \mathbf{X_6})$. We obtain:

$$M_1' = -a_6\frac{\lambda_1}{\lambda_6}M_5\ ,\ M_2' = b_6\frac{\lambda_2}{\lambda_6}M_5\ ,\ M_4' = d_6\frac{\lambda_4}{\lambda_6}M_5$$

The projective coordinates and scale factors can be eliminated by taking cross ratios to obtain two 3D invariants (as opposed to three in [77]):

$$I_1 = \frac{a_5b_6}{a_6b_5} = \frac{M_1M_2'}{M_1'M_2}\ ,\ I_2 = \frac{a_5d_6}{a_6d_5} = \frac{M_1M_4'}{M_1'M_4}$$

Any projective transformation applied to the six-point configuration cancels out in the expressions for $I_1$ and $I_2$ and so these are invariant to 3D projective transformations. Note that this subsumes 3D translations, rotations, anisotropic scaling and affine transformations applied to the configuration.

For the image coordinates, we follow the same approach of taking determinants and their cross-ratios. Using the 'points-left-out' notation as in the 3D case, let $m_{12}$ denote the determinant $|\mathbf{x_3x_4x_5}|$:

$$m_{12} = |\mathbf{x_3x_4x_5}|$$
$$= |\ \mathbf{x_3x_4}(a_5\alpha_1\mathbf{x_1} + b_5\alpha_2\mathbf{x_2} + d_5\alpha_4\mathbf{x_4})/\alpha_5\ |$$
$$= (a_5\alpha_1/\alpha_5)m_{25} + (b_5\alpha_2/\alpha_5)m_{15} \tag{4.8}$$

Letting $m'_{12}$ denote the determinant $|\mathbf{x_3 x_4 x_6}|$, we similarly obtain:

$$\begin{aligned} m'_{12} &= |\mathbf{x_3 x_4 x_6}| \\ &= |\ \mathbf{x_3 x_4}(a_6\alpha_1\mathbf{x_1} + b_6\alpha_2\mathbf{x_2} + c_6\alpha_3\mathbf{x_3} + d_6\alpha_4\mathbf{x_4})/\alpha_6\ | \\ &= (a_6\alpha_1/\alpha_6)m_{25} + (b_6\alpha_2/\alpha_6)m_{15} \end{aligned} \tag{4.9}$$

Similar to the 3D case, we obtain expressions for the other determinants $m_{13}$, $m'_{13}$, $m_{14}$, $m'_{14}$, and calculate cross ratios $(a_5b_6)/(a_6b_5)$ and $(a_5d_6)/(a_6d_5)$ in terms of them, equate them to $I_1$ and $I_2$ respectively to obtain the following final relationship:

$$\begin{aligned} (m_{13}m_{45}m'_{12} - m_{13}m_{25}m'_{14}) \,+\, I_1(m_{13}m_{25}m'_{14} - m_{12}m_{35}m'_{14}) \\ +\, I_2(m_{14}m_{35}m'_{12} - m_{13}m_{45}m'_{12}) = 0 \end{aligned} \tag{4.10}$$

Note that the $m_{ij}$ are quantities computed from image coordinates and we can rewrite the above equations in terms of coefficients, $a_r, b_r, c_r$ as

$$a_r \;+\; b_rI_1 \;+\; c_rI_2 \;\;=\;\; 0 \tag{4.11}$$

The $r$ subscript denotes the restricted 3D scenario. Denote by $\alpha_r$ the vector $(a_r\ b_r\ c_r)$ and by $\mathcal{I}_r$ the vector $(I_1\ \ I_2)$. Equation (4.11) expresses a view-invariant compatibility relationship between solely the 3D coordinates and their 2D image positions for the six points shown in figure 4.1. The six-point configuration is effectively represented by two scalars, $I_1$ and $I_2$ in a 3D-invariant way, and (4.11) describes the mutual-invariant relationship satisfied by any image of the configuration. The advantage of the restricted-3D formulation is that the compatibility equation is linear in the 3D invariants, a fact that we will use later. The obvious disadvantage, of course, is that it can model only a restricted class of 3D objects.

### 4.2.2 Full 3D Case

If point 5 lies outside of the plane spanned by points 1,2 and 4 we have the *full 3D* case considered in [77], where instead of just two invariants, we have three invariants, because $c_5 \neq 0$:

$$I_1 = \frac{a_5 b_6}{a_6 b_5} = \frac{M_1 M_2'}{M_1' M_2}\ ,\ I_2 = \frac{a_5 d_6}{a_6 d_5} = \frac{M_1 M_4'}{M_1' M_4}\ ,\ I_3 = \frac{a_5 c_6}{a_6 c_5} = \frac{M_1 M_3'}{M_1' M_3} \tag{4.12}$$

The compatibility equation is a quadric surface in $I_1, I_2, I_3$ space:

$$a_f I_1 I_2 \ +\ b_f I_2 I_3 \ +\ c_f I_1 I_3 \ +\ d_f I_1 \ +\ e_f I_2 \ +\ f_f I_3 = \ 0 \tag{4.13}$$

The subscript $f$ denotes the full-3D scenario. It can be verified that as point 5 approaches the plane spanned by points 1, 2 and 4, (4.13) approaches (4.11). Thus, if the points 1, 2, 4 and 5 lie *approximately* on a plane the left hand side of (4.11) will be close to zero. Denote the vector $(a_f\ b_f\ c_f\ d_f\ e_f\ f_f)$ by $\alpha_{\mathbf{f}}$ and the vector $(I_1\ I_2\ I_3)$ by $\mathcal{I}_f$.

## 4.3 Key Ideas

In this section, we show how we can apply the results of the previous section to the representation and recognition of human action.

### 4.3.1 Action Modeling

Recall from our discussion in section 1.2.3, that a human action can be thought of in terms of a starting pose $\mathcal{P}_s$, an ending pose $\mathcal{P}_e$, and a sequence of continuous transitions that take the body from pose $\mathcal{P}_s$ at time $t = 0$ to pose $\mathcal{P}_e$ at time $t = T$. We can eliminate rate variations by non-dimensionalizing time such that the action occurs from $t = 0$ to $t = 1$. The *phase* of an action can be represented by $t$ that takes on a value in the interval $[0, 1]$. A phase value maps to a body pose $\mathcal{P}(t)$. An action then becomes the function $\mathcal{P}(t), t \in [0, 1]$. In order to arrive at a view invariant representation of the pose $\mathcal{P}(t)$ , we can choose any six joints of the body and calculate their 3D invariants at each phase of the action. Considering for the moment the full-3D case, an action can be modeled in terms of three temporally varying 3D invariants, $\mathcal{I}_f(t) = (I_1(t), I_2(t), I_3(t))$. The action representation is a parametric curve in 3D invariance space, parameterized by time $t$: each point on the curve corresponds to a phase of the action. For cyclical actions like walking or running, this is a closed curve, while for non-cyclical actions like sitting-down, this is an open curve. The action curve (denote by $\mathcal{A}$) can be discretized and represented at some resolution: $\mathcal{A} = \{\mathcal{I}_f^{(i)}\}$, $i = 1..N$.

### 4.3.2 Action Recognition

Given an image sequence where the joints of the body have been estimated and tracked in each image, the image based quantities $\alpha_{\mathbf{f}}$ can be calculated. The body pose, $\mathcal{I}_f$ is unknown but satisfies the compatibility (4.13). The compatibility equation, which is a quadric surface in $I_1, I_2, I_3$ space, will potentially intersect several action curves at several points, as shown in figure 4.2. The points of intersection are hypotheses of candidate poses among candidate actions. As successive frames are processed, a sequence of quadrics will intersect several action curves at several phases. However, only the true action curve would be intersected in sequence. Hence, the action recognition algorithm involves calculating the points of intersection of each phase $\mathcal{I}_f^{(i)}$ of the action, with the quadric surface determined by $\alpha_{\mathbf{f}}$. In practice the surface may not intersect the action curve and in that case, we find the point on the surface, closest to the phase

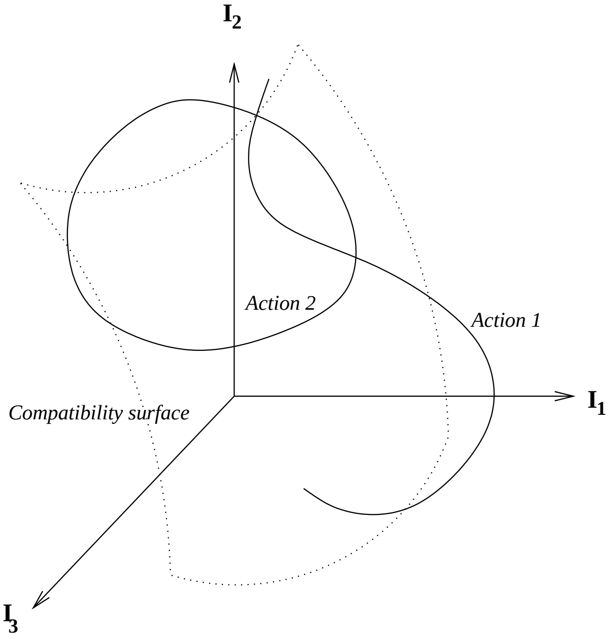


Figure 4.2: Action-curves and action-recognition

using straightforward minimization with Lagrange multipliers. We minimize:

$$F(I_1, I_2,\ I_3\ , \lambda) = (I_1 - I_1^{(i)})^2 + (I_2 - I_2^{(i)})^2 + (I_3 - I_3^{(i)})^2 \\ +\ \lambda\,(a_f I_1 I_2\ +\ b_f I_2 I_3\ +\ c_f I_1 I_3\ +\ d_f I_1\ +\ e_f I_2\ +\ f_f I_3) \quad (4.14)$$

Equating the partial derivatives of $F$ with respect to $I_1^{(t)}, I_2^{(t)}, I_3^{(t)}$ and eliminating them in terms of $\lambda$ gives a sixth degree polynomial in $\lambda$ which can be solved to yield upto six stationary points. Direct substitution of the solutions back into (4.14) will enable calculation of the point of closest approach, $(I_1^*, I_2^*, I_3^*)$. We can then calculate a matching score or *distance, d* as

$$d^2 = (I_1^* - I_1^{(i)})^2 + (I_2^* - I_2^{(i)})^2 + (I_3^* - I_3^{(i)})^2 \quad (4.15)$$

As an example, figure 4.3 shows the distance for phase 0 of the walk-cycle action computed from an arbitrary viewpoint, for a subject walking continuously. The minima of the curve are the frames where the subject is actually at phase 0 of the action.

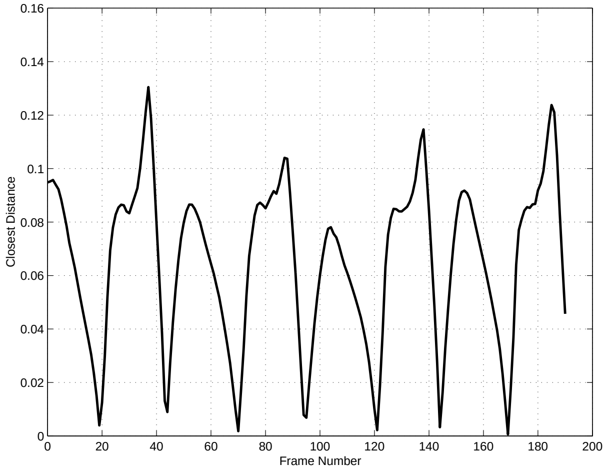


Figure 4.3: $d$ for a subject walking, phase=0, viewpoint=1

The action modeling and recognition are simpler in the restricted 3D case because there are only two invariants. An action can be modeled as a curve in 2D space (see figure 4.4). Furthermore, the compatibility (4.11) is linear, which makes action recognition far simpler than in the full-3D case.

### 4.3.3 Temporally Distributed Joints

It is rarely the case that the action curves as described above are bounded. If a determinant in the denominator of any component of $\mathcal{I}_f$ becomes zero, as would be the case when the associated quadruplet of points is coplanar, the curve passes thru infinity - this is not a theoretical problem because a point at infinity is a valid point in 3D projective space (recall our discussion in chapter 2). However, this poses practical problems for implementation. One way to deal with such cases is to define the reciprocal invariant, $I_i' = 1/I_i$ if $|I_i| > 1$ etc. and obtain an equivalent compatibility equation in the reciprocal invariant. In such a case, we would tag points on the action curve with boolean variables indicating whether the compatibility at that point refers

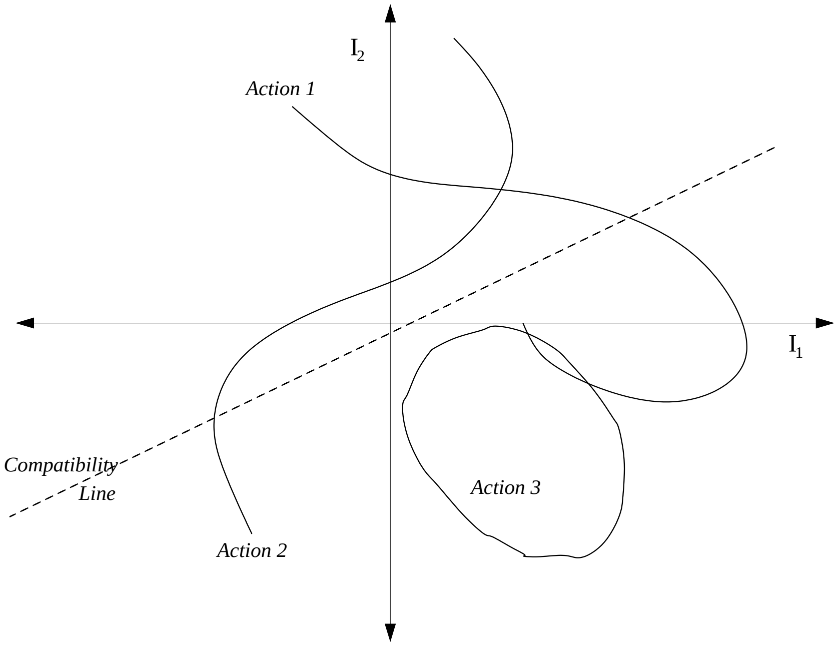


Figure 4.4: Restricted-3D case: Action-curves and action-recognition

to one or more reciprocal invariants. The advantage is that the action curves would all be bounded within a cube of side 2 in 3D projective space. However, this increases the effective degree of the compatibility equation, and straightforward Lagrange minimization leads to a polynomial system of equations that requires the heavy machinery of Grobner bases [17] for a solution. Rather than follow this path, we can instead exploit temporal aspects of the action: One can detect the starting and ending of an action using the approach of the previous section (choosing joint combinations where the $\mathcal{I}_f$ for the start and end poses do not contain infinities). Rather than restricting the six-joint set to one particular pose, we can distribute them across multiple poses between the starting and ending pose such that at all times during the action, none of the quadruplets [2346], [1345], [1245], [1235] are coplanar, ensuring that infinities are avoided. This method of distributing points across different frames parallels the concept of an invariance space trajectory (IST) that was introduced for the 2D case

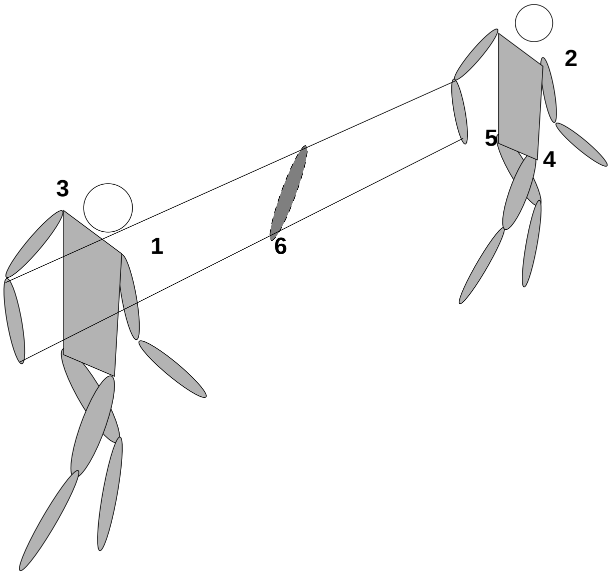


Figure 4.5: Temporally distributed joints for the walk-cycle action

in the previous chapter. For the walk-cycle action, figure 4.5 shows one particular distribution of joints such that infinities are avoided, and the resulting action-curves are bounded. Note that in this case, some joints are always held fixed at the start or end of the action whereas others may be moving. We would thus tag each joint used with the symbols $s$ for start, $e$ for end and $m$ for moving. For example, in figure 4.5, joint 6 would be tagged with the symbol $m$. Note that such a representation provides for robustness against rate variations in the action since we are modeling curves delineated by the starting and ending poses of the action. Delineation of the action is ‘event-driven’, i.e. determined by the occurrences of the starting and ending poses of the action and not by the passage of a fixed amount of time.

### 4.3.4 Fixed Camera

In the case of a fixed camera, an action can be verified rather easily given a hypothesis of the starting and ending of the action [55]. The approach would be as follows: Along with the invariant representation of the starting and ending poses of an action, we also store the Euclidean representation of the action in an arbitrary world frame of reference. Then, when a hypothesis is made of the starting and ending of an action, there are enough point correspondences to enable the calculation of the world-to-image transformation : six points for the starting pose and six for the ending pose. Collectively, this provides 24 equations in the 11 unknowns of the world-to-image transformation matrix $\mathbf{T}$. Once $\mathbf{T}$ is determined, it is a simple matter of projecting the Euclidean representation of the action from the world coordinates to the image coordinates and verifying the projection. Note that for this fixed camera case, the 2D approach of the previous chapter can also benefit from this idea, except that there will be 10 points, leading to 20 equations, still more than enough to recover $\mathbf{T}$.

If the camera is moving, $\mathbf{T}$ becomes a varying quantity, and we need to resort to a projective representation as we outline above.

## 4.4 Analysis of the Representation

A pose $\mathcal{P}$ of the human body is completely described by the joint angles of every joint of the body. Rather than joint angles, we are modeling a pose by three invariants $\mathcal{I}_f$ formed by six joints of the body. Similar to what we did in our 2D approach described in section 3.3, we analyze this representation against *minimalism, completeness, continuity* and *uniqueness*. We likewise consider the sources that contribute to variabilities in the 3D ‘invariants’ and the effectiveness of the use of invariants to represent a point configuration. We consider the full-3D case.

### 4.4.1 Minimalism

Six joints cannot be modeled in a projectively invariant manner using two invariants because there are indeed three degrees of freedom. Hence, we satisfy the minimalism requirement rather easily.

### 4.4.2 Completeness

The completeness requirement states that every entity being represented should have a representation. While six joints in general position have a representation, there exist degenerate cases, where the first four lie on a plane, that do not result in a representation. Secondly, when specific joints fall on a plane, the denominators in the expressions in (4.12) become zero, sending the invariants to infinity. Indeed, these were some of the problems we faced in the implementation and we addressed them by choosing joints where such degeneracies are avoided (as described section 4.3.3).

### 4.4.3 Continuity

Small changes in the represented entity should result in small changes in the representation. This requirement is easily met because it can be seen that the components of $\mathcal{I}_f$ are bilinear or biquadric polynomials in the coordinates which makes them continuous.

### 4.4.4 Uniqueness

There should be a one-one mapping between the represented entity and the representation. There are two aspects of uniqueness to consider: (1) whether the mapping between a pose $\mathcal{P}$ and the resulting representation $\mathcal{I}_f$ is unique, and (2) whether the mapping between $\mathcal{P}$ and the satisfiability of (4.13) is unique. Consider (1): A particular pose $\mathcal{P}$ will always result in a unique value for $\mathcal{I}_f$, making the mapping from $\mathcal{P}$ to $\mathcal{I}_f$ trivially unique. However, if the associated six joints are irrelevant to the pose, the same value of $\mathcal{I}_f$ will occur for different $\mathcal{P}$. To give a concrete example, consider two poses of a person: (a) the person is sitting down (b) the person has the left hand raised while sitting down. If the feet, knees and right hand joints are used to model the pose, $\mathcal{I}_f$ will be the same for both poses. Thus, the mapping from $\mathcal{I}_f$ to $\mathcal{P}$ will be non-unique if the six joints do not capture the body pose. Furthermore, if two different poses of the body are such that the six joints are in projectively equivalent configurations, the same value of $\mathcal{I}_f$ will occur. Now consider (2): The mapping from a pose to the satisfiability of (4.13) is trivially unique but the converse is not true: The image based quantities, $\alpha_{\mathbf{f}}$, for two different poses of the body that project to the same positions will be the same resulting in the satisfiability of (4.13) for two different poses. Hence, uniqueness is a problem on both counts.

The way we mitigate the representability problem due to (1) is a judicious choice of joints for the modeling, as we did for the 2D approach. We mitigate the other problem due to (1) (where two truly different poses are in 3D projectively equivalent configurations resulting in the same values of $\mathcal{I}_f$), and the problem due to (2), is to use multiple six-joint sets to model a pose : While spurious poses may occur resulting in matches for one set, it is unlikely that all six-joint sets will be matched. Besides this spatial redundancy strategy, temporal redundancy (as we described section 4.3.3) can be used to further mitigate spurious matches.

### 4.4.5 Variability

Our representation, $\mathcal{I}_f$, is invariant to 3D projective, affine and Euclidean transformations of the body. However, due to allometry, it can be expected that it will not be subject-invariant. Figure 4.6 shows the 217-sample histogram of $I_1$ for a particular pose in the walk-cycle action taken from motion capture data of 12 subjects. Besides

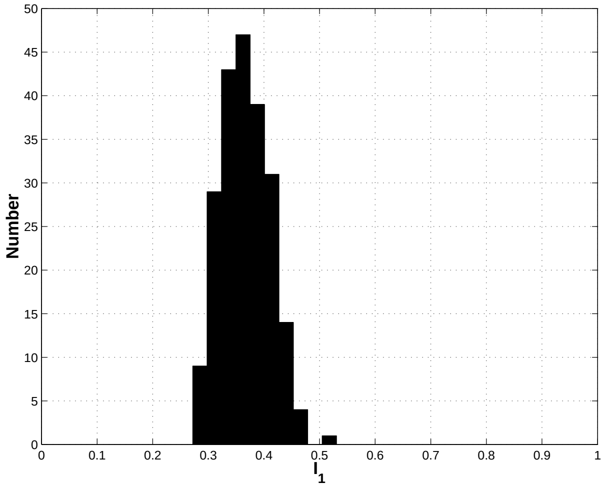


Figure 4.6: $I_1$ histogram, action='walk-cycle', phase=0.4, sample-size=217

spatial variability, there also is temporal variability to account for because the same action, repeated multiple times by a single subject or when performed by different subjects, will result in slightly different action curves. We deal with these variabilities by maintaining the empirically derived median $\mu$ and standard deviation $\sigma$ for each invariant value and store it along with the invariants themselves. Rather than minimizing (4.14), we minimize:

$$F(I_1, I_2,\ I_3\ , \lambda) = \left(\frac{I_1 - \mu_{I_1}^{(i)}}{\sigma_{I_1}^{(i)}}\right)^2 + \left(\frac{I_2 - \mu_{I_2}^{(i)}}{\sigma_{I_2}^{(i)}}\right)^2 + \left(\frac{I_3 - \mu_{I_3}^{(i)}}{\sigma_{I_3}^{(i)}}\right)^2 \\ + \lambda\left(a_f I_1 I_2\ +\ b_f I_2 I_3\ +\ c_f I_1 I_3\ +\ d_f I_1\ +\ e_f I_2\ +\ f_f I_3\right) \quad (4.16)$$

The distance in (4.3.2) is redefined as:

$$d^2 = \left(\frac{I_1^* - \mu_{I_1}^{(i)}}{\sigma_{I_1}^{(i)}}\right)^2 + \left(\frac{I_2^* - \mu_{I_2}^{(i)}}{\sigma_{I_2}^{(i)}}\right)^2 + \left(\frac{I_3^* - \mu_{I_3}^{(i)}}{\sigma_{I_3}^{(i)}}\right)^2 \quad (4.17)$$

where $(I_1^*, I_2^*, I_3^*)$ minimize (4.16).

### 4.4.6 Represention using Invariants

As we observed in the previous chapter, Astrom and Morin [5], and Maybank [43] independently showed that the probability density function for the 1D cross-ratio exhibits logarithmic singularities at 0 and 1. The 3D invariants, $\mathcal{I}_f$, are not uniformly distributed across the 3D space either, as we found by Monte Carlo simulation. Not surprisingly, the distributions for $I_1$, $I_2$ and $I_3$ were identical. Figure 4.7 shows the experimentally computed distribution for the 3D case using six 3D points whose coordinates are uniformly distributed in $[0,1]$. Also shown is the analytically derived pdf for the 1D cross-ratio using the formulas in [5]. It can be seen that there are singularities at 0 and 1 for the 3D case as well and that the curves are very similar. One could, in principle, proceed along the lines of [5] and extend the 1D cross-ratio analysis to the 3D case and derive analytical expressions for the 3D invariants. For our purposes, however, it is sufficient to make the observation that pdfs for the 3D case are similar to the 1D case, and that problems introduced due to the non-uniformity of the pdf in the 1D case will also apply to our 3D case.

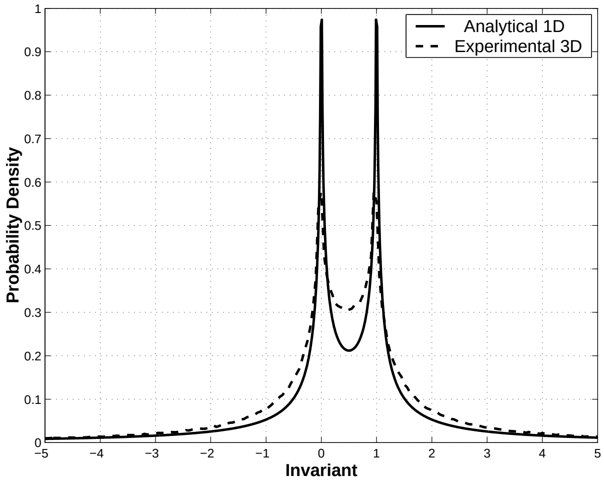


Figure 4.7: Probability Distributions of Invariants

Noting that these are the very same problems that we encountered for our 2D approach of the previous chapter which we mitigated by using spatial and temporal coherency constraints, we use the same approach here:

1. While deciding whether a body is in a given phase of a given action, we not only threshold the distance $d$ (given by equaton 4.16) but also stipulate that the distance be a local minimum in time.

2. Representing a pose by the invariants of multiple six-joint sets provides for a measure of robustness against random mis-matches : while one six-joint set may match spuriously, the likelihood of all six-joint sets matching spuriously is small.

3. The use of action curves provides for temporal redundancy: while there may be spurious matches at a few poses of the curves the chances that there is a spurious match for every pose of the curve will be small.

## 4.5 Model Building

Model-building (i.e. estimation of the action curves for specific actions) was done empirically, using motion-capture data which provide the 3D coordinates for selected body joints for each frame of a human action. The body model we employed consisted of fifteen joints: (see figure 3.7) namely the head, hip, chest, shoulders, elbows, hands, hips, knees and feet. We tried several joint combinations that would achieve high inter-class and small intra-class variation in the action curves and chose the following joint combinations:

### 4.5.1 Full-3D

For walking and running, the joint combination {Left-hand, Right-hand, Left-knee, Right-knee, Left-foot, Right-foot} was used for detecting the starting and ending of the action. We modeled the temporal aspects of the actions by using two action curves made up of the following joints (recall section 4.3.3 where the tags $s$, $e$ and $m$ are explained).

1. Right-shoulder (s), Left-shoulder (e), Left-shoulder (s), Right-hip (s), Left-hip (e), Left-hand (m)

2. Right-hip (s), Right-hip (e), Left-hip (s), Right-foot (s), Left-foot (e), Left-foot (m)

For sitting down, the start of the action was modeled using relative stationarity of the feet (which is preserved from 3D to 2D, although only for a fixed camera). The end was modeled using the following joints: Left-knee, Left-hip, Left-shoulder, Right-shoulder, Right-foot, Left-foot. The temporal aspects were modeled with the following joint combination: Right-shoulder (e), Left-shoulder (e), Left-shoulder (s), Right-hip (e), Right-knee (e), Left-hip (m).

For forward-jump, the subject's feet are stationary relative to the other joints at the start and the end of the jump. This relative stationarity was used to detect the starting and ending of the action. For the temporal aspects, we used the following two joint combinations:

1. Right-shoulder (s) Left-shoulder (s) Right-knee (s) Right-shoulder (e) Left-hip (e) Left-knee (m).

2. Left-shoulder (s) Right-shoulder (s) Left-knee (s) Left-shoulder (e) Right-hip (e) Right-knee (m).

Figure 4.8 shows the action curves (the curve arising from the first set of joints is shown for walking and running).

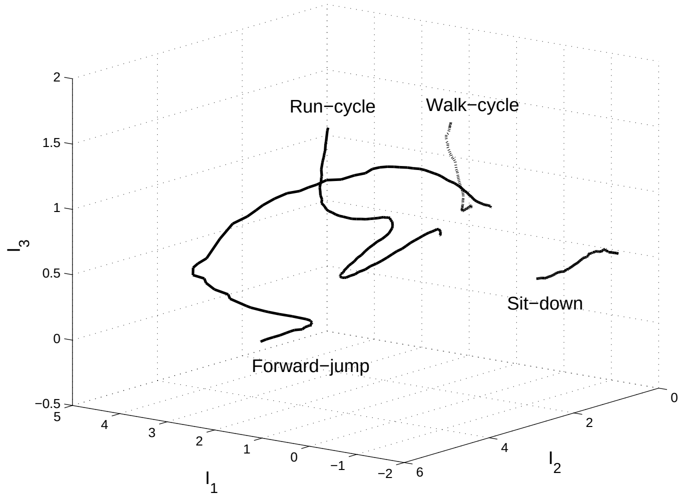


Figure 4.8: Full-3D Median Action Curves. Note that only walk-cycle and run-cycle deserve to be compared because they correspond to the same set of joints. Sit-down and forward-jump use different joint sets and are shown for illustrative purposes only. The fact that they are different than the other curves in the figure is not significant.

### Restricted 3D

For walking and running, for detecting the starting and ending of the action, the set of joints used were the same as in the full-3D case. However, for the temporal component of the actions, restricted-3D action curves arising from the following two sets of joints were used:

1. Left-shoulder (s), Right-shoulder (s), Hip (m), Right-shoulder (e), Left-shoulder (e), Left-hand (m)

2. Left-shoulder (s), Right-shoulder (e), Right-foot (s), Right-shoulder (s), Left-shoulder (e), Left-hand (m)

Note that points 1,2,4 and 5 of the above two joint combinations, which correspond to the shoulder positions at the start and end of the action, lie approximately on a plane for walking and running.

For sit-down, for detecting the starting and ending poses, the set of joints used were the same as in the full-3D case. For the temporal aspect, we used one action curve arising from the joint combination: $\{$Right-shoulder (e), Right-hip (e), Left-hip (e), Right-knee (s), Right-shoulder (s), Left-shoulder (m)$\}$. As in the walking/running case, points 1,2,4 and 5 of the joint combination lie approximately on a plane, allowing us to use the restricted-3D formulation.

For forward-jump, the starting and ending of the action were modeled in the same way as in the full-3d case. The temporal aspects of the action were modeled using action curves arising from the following two sets of joints:

1. Left-shoulder (s) Right-shoulder (s) Right-hip (e) Right-shoulder (e) Left-shoulder (e) Left-hip (m)

2. Right-shoulder (s) Left-shoulder (s) Left-hip (e) Left-shoulder (e) Right-shoulder (e) Right-hip (m)

Figure 4.9 shows the action curves (the curve arising from the first set of joints is shown for *walk-cycle, run-cycle* and *forward-jump*).

In all, 217 walk-cycles, 204 run-cycles, 108 sit-down actions and 96 forward-jump instances were used to build the action curves. The model stored the median action curves as well as their standard deviations. As in the 2D approach, we observe here that there was artificial variability introduced due to inconsistencies in the placement of sensors on the motion capture subjects because these were obtained from different sources. These inconsistencies were partially compensated by adding or subtracting

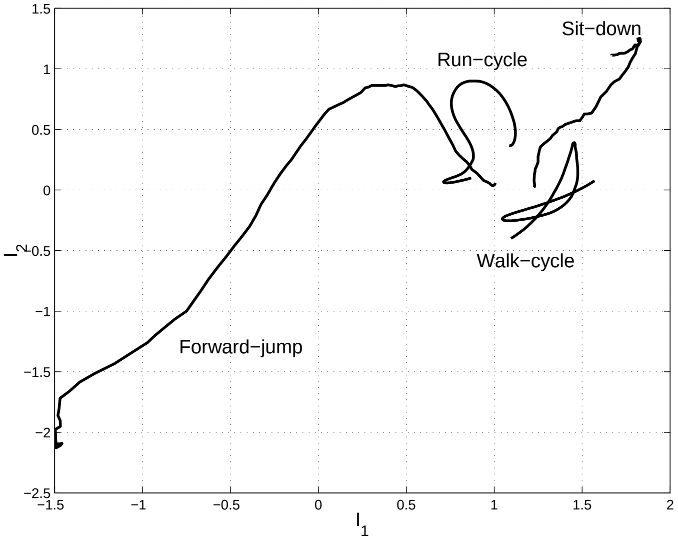


Figure 4.9: Restricted-3D Median Action Curves. Only walk-cycle and run-cycle use the same set of joints and can be compared.

vectors of small magnitudes from the given positions and visually verifying the positions by rendering the data. Inspite of this, it was inevitable that some artificial variation still remained.

## 4.6 Results

We tested the approaches on the same data as for the 2D case, so we could draw comparisons between the 2D and 3D approaches. We also used the same metrics *true-detection-rate, false-alarm-rate*, and *misclassification-rate* (as described in chapter 3 section 3.6).

### 4.6.1 Motion Capture

Recall that the data set included 1200 instances of actions for detection. Table 4.1 shows the results for the full-3D approach while table 4.2 shows the results for the restricted-3D approach.

Table 4.1: Full-3D Results, Motion Capture Sequences, 1200 Total Actions

| Metric | Viewpt. 1 | Viewpt. 2 | Viewpt. 3 | Viewpt. 4 | Viewpt. 5 |
|---|---:|---:|---:|---:|---:|
| True Detections | 1096 | 1077 | 1039 | 996 | 1012 |
| False alarms | 74 | 94 | 124 | 36 | 42 |
| Misclass. | 131 | 113 | 151 | 159 | 124 |
| True Det. % | 91.33 | 89.75 | 86.58 | 83.00 | 84.33 |
| False Alarm % | 5.69 | 7.32 | 9.43 | 3.02 | 3.57 |
| Misclass. % | 10.07 | 8.80 | 11.49 | 13.35 | 10.53 |

The numbers raise several questions:

1. Why do we not get 100% detection, 0% false alarms and 0% misclassification?

2. Why is there viewpoint dependency on the results?

3. Why are some viewpoints better than others?

To answer these questions, we need to consider the various sources of error.

Firstly, there is the issue of distinguishability. We noted that there is intra and inter-subject variability in the action such that the ‘same’ action will result in slightly different curves in invariance space. We addressed this by choosing the median curve and stored the standard deviations of the invariant values at each phase. We calculate

Table 4.2: Restricted-3D Results, Motion Capture Sequences, 1200 Total Actions

| Metric | Viewpt. 1 | Viewpt. 2 | Viewpt. 3 | Viewpt. 4 | Viewpt. 5 |
|---|---:|---:|---:|---:|---:|
| True Detections | 973 | 1038 | 999 | 1033 | 1067 |
| False alarms | 77 | 91 | 71 | 47 | 42 |
| Misclass. | 211 | 65 | 130 | 87 | 96 |
| True Det. % | 81.08 | 86.5 | 83.25 | 86.08 | 88.92 |
| False Alarm % | 6.11 | 7.62 | 5.92 | 4.03 | 3.46 |
| Misclass. % | 16.73 | 5.44 | 10.83 | 7.46 | 7.91 |

weighted distances to this median curve and classify an unknown instance of action based on distance to this median curve. Given this, it is possible that an instance of action happens to be closer to a different action model, leading to an incorrect classification, affecting both, the detection as well as the misclassfication rates.

Secondly, the view invariant properties captured by (4.11) and (4.13) imply only that the compatibility lines and surfaces intersect the model at a given phase for all viewpoints. However note that these lines and surfaces are derived from image quantities $\alpha_r$ and $\alpha_f$ and are hence dependent upon viewpoint. For the full-3D case, some of the joints chosen project very close to each other from the top view (viewpoint no. 5). For instance, the shoulder and hip, which are joints used to obtain the action curve, will project very close to each other from this view, giving rise to small values for certain components of $\alpha_f$, which in turn lead to to errors in classification.

Thirdly, there is the issue of validity of the underlying four-point planarity assumption in the restricted-3D case. If points 1,2,4 and 5 are not close to being planar, the restricted 3D compatibility (4.11) will cease to hold. In fact, the right hand side of the equation will no longer be zero but a quantity that is dependent not only upon the degree of non-planarity but also upon the viewpoint. This was one of the reasons why viewpoint 1 had a particularly high misclassification rate. These observations are illustrated for the restricted-3D case in figures 4.10 and 4.11. Figure 4.10 shows the

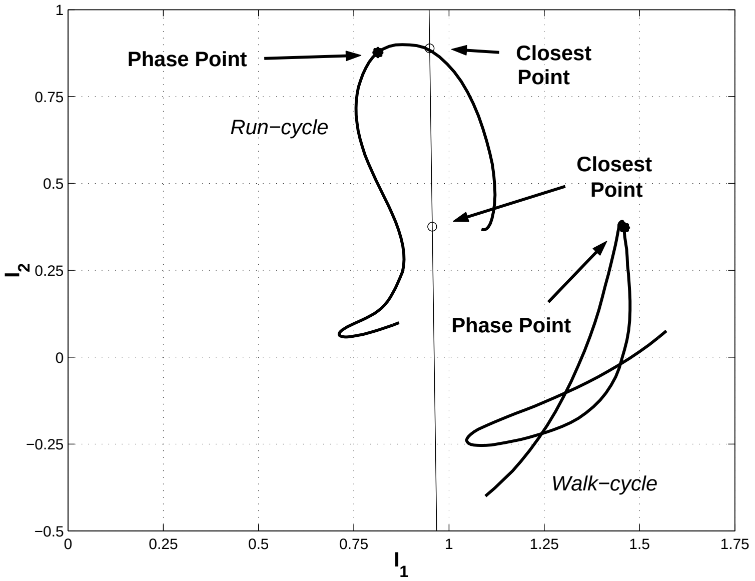


Figure 4.10: Viewpoint 1 - Misclassification of Walk-cycle as Run-cycle

walk-cycle and run-cycle models along with the compatibility line for a subject walking at phase 0.7 as seen from viewpoint 1. The figure also shows the point of closest approach to each of the curves, calculated as described in section 4.3.2. It can be seen that the distance to the run-cycle is smaller. A similar behavior was seen for almost all phases, leading to an incorrect classification. In contrast, figure 4.11 shows the same quantities from viewpoint 5. Note that the compatibility line is different and that the point of closest approach for walk-cycle is much closer to the expected point. At the same time, the point of closest approach for run-cycle lies outside the plot. A similar behavior was seen for all phases, leading to a correct classification.

Fourthly, for the forward-jump action, the subjects crouched before jumping, mim-

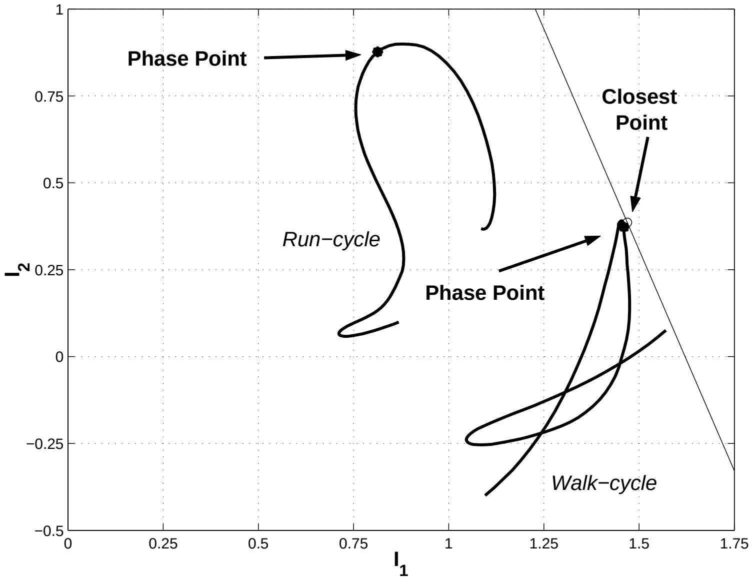


Figure 4.11: Viewpoint 5 - Correct classification of Walk-cycle

icking a sit-down action. Similarly, upon landing after the jumping, they assumed a similar crouching posture. Both of these were detected as sit-down actions by the system, increasing the false alarm rate.

Finally, though we experimented with several joint combinations, we did not carry out extensive searches for joint combinations that would maximize inter-class variation while minimizing intra-class variation. Indeed, there may be particular joint combinations and thresholds that would result in better receiver operating characteristics than what we report here. Our goal was to demonstrate the effectiveness of our approach at recognizing an action from any viewpoint, given a single model. The results show that the overall performance is quite good for a diverse set of viewpoints.

### 4.6.2 Real Image Sequences

We ran the full-3D and restricted-3D approaches on the same real image sequences as we did for the 2D approach. The models used for detecting actions for these real image sequences were the same as those used for the motion capture sequences. The only difference was that we needed to perform more temporal-smoothing of the data than for the motion capture sequences. Tables 4.3 and 4.4 show the results for the full-3D case and the restricted-3D case respectively. One can expect that the results will be somewhat inferior to those on motion capture sequences because of errors in the image joint positions. The observations we made in the previous section about sources of error and distinguishability of the models apply for the real image sequences as well. However, as we observed in chapter 3, the number of sequences used do not permit us to draw definite conclusions regarding the modes of failure, or the reasons for differences in results across viewpoints. Nevertheless, one conclusion that can be drawn is that the results show good performance from the three views for the full-3D and restricted-3D approaches inspite of noisy image data.

Table 4.3: Full-3D Results, Real Image Sequences, 40 Total Actions

| Metric | Front-View | Side-View | Top-View |
|---|---:|---:|---:|
| No. of Sequences | 13 | 14 | 13 |
| True Detections | 9 | 13 | 9 |
| False alarms | 0 | 2 | 0 |
| Misclass. | 3 | 1 | 4 |

Table 4.4: Restricted-3D Results, Real Image Sequences, 40 Total Actions

| Metric | Front-View | Side-View | Top-View |
|---|---:|---:|---:|
| No. of Sequences | 13 | 14 | 13 |
| True Detections | 10 | 11 | 12 |
| False alarms | 0 | 2 | 0 |
| Misclass. | 2 | 3 | 0 |

## 4.7 Summary

We presented approaches for developing a high-level representation and an effective recognition algorithm for human action that is resistant to variations in viewpoint, speed of the action and to minor inconsistencies in the action when performed repeatedly by the same subject as well as when performed by different subjects. We showed that we could achieve these objectives by representing actions as curves in spaces arising from 3D mutual invariants. Recognition of actions amounts to keeping track of intersections of these curves by compatibility lines and surfaces calculated from the input image sequences. We presented two approaches - the restricted-3D approach and the full-3D approach. The restricted-3D approach is less generally applicable because it requires the presence of four joints that lie approximately on a plane. However, actions can be modeled as curves in a 2D space and action recognition requires calculating the intersections of these curves by straight lines. In comparison, the full-3D approach is completely general but results in action curves that reside in a 3D space with action recognition amounting to calculating the intersections of these curves by quadric surfaces. A detailed analysis of the approaches revealed inherent difficulties with using them. However, heuristics that enforced spatial and temporal coherency constraints were proposed to surmount these difficulties. We evaluated the approach on four actions - walk-cycle, run-cycle, sit-down and forward-jump from a variety of views on two modalities of input - 2D projections of human motion capture data and manually segmented real video sequences. Our use of these two modalities of input effectively simulated for us the output of a body-joint detection and tracking module. We demonstrated that a single view-independent representation of an action was sufficient to recognize and distinguish it from other actions from a variety of viewpoints with good success on both input modalities.

# Chapter 5

# Pose Estimation

## 5.1 Introduction

In contrast to the *representation* of pose which formed a major part of our primary focus on human action representation and recognition in chapters 3 and 4, we deal with the *estimation* of pose in this chapter. By estimation, we mean the complete recovery of the 3D positions of all the major joints of the body in a body centric coordinate system. This type of Euclidean estimation is in contrast to the 2D and 3D projective representations of pose. While a projective representation suffices for the problem of action recognition, Euclidean estimation of pose is necessary for the problem of visual motion capture from archived video. A typical application would involve a user specifying the positions in the image of several body joints, while the system estimates the initial pose of the body and proceeds to track these from frame to frame. The final output of the system would be the temporal evolution of the joint angles of the body : a capture of the motion. Working with calibrated image/video data and/or multiple cameras is possible only in restricted application domains. Most archived videos are monocular with unknown camera parameters (intrinsic and extrinsic). The scaled orthographic assumption, which has been used by previous researchers (e.g.

[9], [71], [6]), is too restrictive for many cases where perspective effects are strong. A full-perspective solution to the problem will increase the applicability of good tracking algorithms such as Bregler's [9] because in addition to providing a more accurate initial estimate, in the fixed-camera case, one can recover the perspective 3D to 2D transform of the camera, making it possible to carry out full-perspective tracking of the human body. In this chapter, we aim for such a solution and seek to estimate the 3D positions of various body landmarks in a body-centric coordinate system. Using the restricted-3D formulation derived in chapter 4, we set up a simple polynomial system of equations in the unknown variables, for which analytical solutions exist. In cases where no solutions exist, an approximate solution to the polynomials is found. Recovery of the 3D joint angles, which are helpful for tracking, then becomes possible by way of inverse kinematics on the limbs.

## 5.2 Problem Statement

We employ a simplified human body model of fourteen joints and four face landmarks as shown in figure 5.1 The fourteen body joints are - two feet, two knees, two hips (about which the upper-legs rotate), pelvis, upper-neck (about which the head rotates), two shoulders, two elbows and two hands. The facial landmarks correspond to the forehead, nose, chin and (right or left) ear. The hip joints constitute a rigid body. Choosing the pelvis as the origin, we can define the X axis as the line passing through the pelvis and the two hips. The line joining the base of the neck with the pelvis can be taken as the positive Y axis. The Z axis then becomes the line perpendicular to the XY plane and pointing in the forward direction to make a right handed coordinate system. We call the XY plane the torso plane. We scale the coordinate system such that the

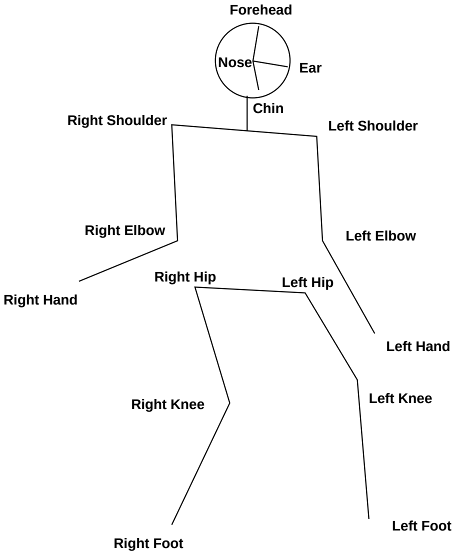


Figure 5.1: Body Model

head-to-chin distance is unity.

With respect to the input and output, the problem we seek to solve in this paper is similar to those addressed previously (e.g. [71], [6]): Given an image with the location in the image of the body landmarks and the relative body lengths, recover their body-centric coordinates.

We make use of two assumptions described below:

1. We use the isometry approximation where all subjects are assumed to have the same body part lengths when scaled. As we described in chapter 3, the allometry approximation, where the proportions are dependent on body size is better. Given that the input is manual, it may be possible to choose an appropriate body model reflecting the age and body size of the human in the image, rather than a one-size-fits-all body model. However, as for the 3D action representation case, the pose-estimation algorithm we describe below is invariant to full-body 3D projective transformations.

2. The torso twist is small such that the shoulders take on fixed coordinates in the body-centered coordinate system. Except for the case where the subject twists the shoulder-line relative to the hip-line by a large angle, this assumption is usually applicable. Further, since our algorithm relies on manual input, it is easy to tell if this assumption is violated. This assumption allows us to apply the restricted-3D equation we derived in chapter 4, to this problem where the points on the shoulder and hip will form the required plane.

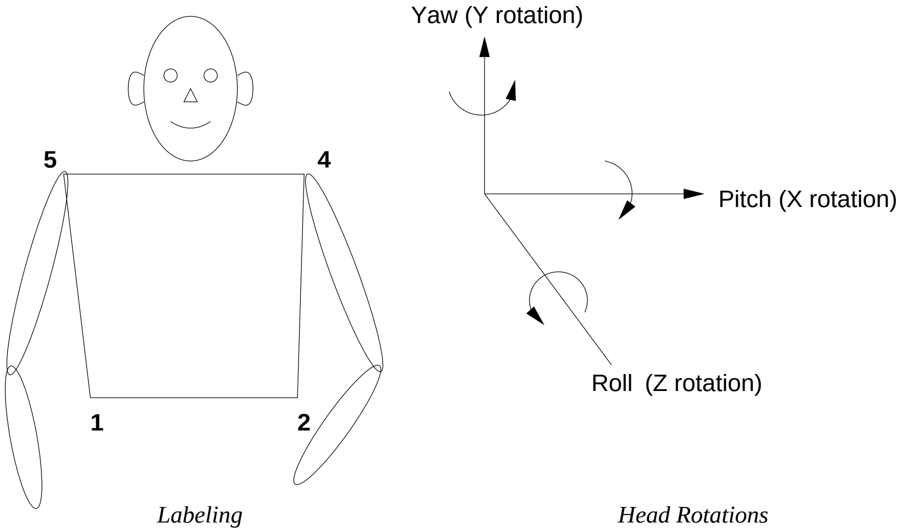
We will first show how to recover the three angles of rotations of the head in the body-centric coordinate system, given the image locations of the body landmarks. From the recovered head orientation, we next show how the 3D coordinates of the remaining joints can be recovered. Recovery of these quantities also allows us to determine the epipolar geometry of the camera.

## 5.3 Recovering the Head Orientation

Points 1,2,4 and 5 (refer to figure 4.1 from chapter 4) will be the two shoulders and the two hips, which, given our small torso-twist approximation, lie approximately on a plane. Recall the derivation of the (4.11) from chapter 4 which we repeat below and drop the $r$ subscript for convenience.

$$aM_1'M_2M_4 \;+\; bM_1M_2'M_4 \;+\; cM_1M_2M_4' \;=\; 0 \tag{5.1}$$

Equation 5.1 expresses a view-invariant relationship between solely the 3D coordinates and their 2D image positions for the six points shown in figure 4.1. If we choose the following labeling of points (see figure 5.2): right-hip(1), left-hip(2), left-shoulder(4), right-shoulder(5) and allow points 3 and 6 to be any two head features in (say forehead and chin), the only unknowns in the equation are the coordinates $\mathbf{X_3}$ and $\mathbf{X_6}$. Being



*Labeling* *Head Rotations*

Figure 5.2: Point labeling for recovering head orientation

positions on the head, which is a rigid body that rotates about the upper-neck, in effect, there are only 3 scalar unknowns corresponding to a rotation matrix $\mathbf{R}$. If we use Euler angles and denote the rotation about the X, Y, and Z axes as $\theta_1$, $\theta_2$ and $\theta_3$ respectively, we can write:

$$\begin{aligned}\mathbf{X_3} &= \mathbf{R}(\theta_1, \theta_2, \theta_3)\mathbf{X_{30}} \\ \mathbf{X_6} &= \mathbf{R}(\theta_1, \theta_2, \theta_3)\mathbf{X_{60}}\end{aligned}$$

where $\mathbf{X_{30}}$ and $\mathbf{X_{60}}$ are known forehead and chin coordinates corresponding to a reference 'neutral' position.

**Theorem 1.** *There are upto eight possible head orientations that explain the image formed by a head-torso combination with zero torso-twist.*

*Proof.* Writing

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} = \begin{pmatrix} \mathbf{r_1}^T \\ \mathbf{r_2}^T \\ \mathbf{r_3}^T \end{pmatrix}$$

where $\mathbf{r_i}^T = (r_{11}\ r_{12}\ r_{13})$.

Observing that the third elements of $\mathbf{X_2}, \mathbf{X_4}, \mathbf{X_5}$ are zero,

$$\begin{aligned} M_1 &= [\mathbf{X_2}\ \mathbf{X_3}\ \mathbf{X_4}\ \mathbf{X_5}] \\ &= [\mathbf{X_2}\ (\mathbf{RX_{30}})\ \mathbf{X_4}\ \mathbf{X_5}] \\ &= p_{245}\mathbf{r_3^T X_{30}} \end{aligned}$$

where $p_{245}$ is the signed area of points $\mathbf{X_2}, \mathbf{X_4}, \mathbf{X_5}$, a known constant. Similarly, we can write

$$\begin{aligned} M_2 &= p_{145}\mathbf{r_3^T X_{30}} \\ M_4 &= -p_{125}\mathbf{r_3^T X_{30}} \end{aligned}$$

When the above are substituted into (5.1), the scalar $\mathbf{r_3^T X_{30}}$ cancels out and we obtain

$$a'M_1' \ + \ b'M_2' \ + \ c'M_4' \ \ = \ \ 0 \tag{5.2}$$

where

$$a' = -ap_{145}p_{125} \tag{5.3}$$

$$b' = -bp_{245}p_{125} \tag{5.4}$$

$$c' = cp_{245}p_{145} \tag{5.5}$$

We now write expressions for $M_i'$:

$$\begin{aligned} M_1' &= [\mathbf{X_2}\ \mathbf{X_3}\ \mathbf{X_4}\ \mathbf{X_6}] \\ &= [\mathbf{X_2}\ (\mathbf{RX_{30}})\ \mathbf{X_4}\ (\mathbf{RX_{60}})] \end{aligned} \tag{5.6}$$

Expanding R in terms of the Euler angles $\theta_1, \theta_2, \theta_3$, and substituting it in the expressions for the determinants $M_i'$, (5.2) becomes a 13 term transcendental equation in the

Euler angles. Given the point correspondences of two more head features, say the nose and either ear, we will have three equations in the three unknown Euler angles. The equations depend on the neutral position of the head reflected in $X_{30}$ and $X_{60}$. Choosing a neutral position where the head points forward with no yaw or roll, the $x$ coordinates are zero for the forehead, nose and chin and two of the equations become four term transcendental equations. We then have:

$$a_1 \sin\theta_1 + a_2 \cos\theta_1 + a_3 \sin\theta_3 + a_4 \cos\theta_3 = 0 \tag{5.7}$$

$$b_1 \sin\theta_1 + b_2 \cos\theta_1 + b_3 \sin\theta_3 + b_4 \cos\theta_3 = 0 \tag{5.8}$$

$$\begin{aligned} d_1 \sin\theta_3 \cos\theta_2 + d_2 \sin\theta_2 \cos\theta_1 \sin\theta_3 + d_3 \sin\theta_3 \cos\theta_1 \, + \\ d_4 \cos\theta_1 \cos\theta_3 + d_5 \sin\theta_2 \cos\theta_1 + d_6 \cos\theta_2 \cos\theta_3 \, + \\ d_7 \cos\theta_1 \cos\theta_2 + d_8 \sin\theta_2 + d_9 \sin\theta_1 \sin\theta_2 \cos\theta_3 \, + \\ d_{10} \sin\theta_1 \sin\theta_2 \sin\theta_3 + d_{11} \sin\theta_1 \cos\theta_3 + d_{12} \sin\theta_1 \cos\theta_2 \, + \\ d_{13} \sin\theta_1 \sin\theta_3 = 0 \end{aligned} \tag{5.9}$$

Interestingly, (5.7) and (5.8) are independent of $\theta_2$ and can be solved rather trivially using
$\sin^2\theta_1 + \cos\theta_1^2 = 1$ and $\sin^2\theta_3 + \cos\theta_3^2 = 1$. We obtain a quadratic equation in $\sin^2\theta_1$:

$$h_1 \sin^4\theta_1 + h_2 \sin^2\theta_1 + h_3 = 0 \tag{5.10}$$

where the $h_i$ can be written in terms of $a_i$ and $b_i$. Hence there are upto four solutions for $\theta_1$ and $\theta_3$. When these are substituted into (5.9), we obtain a simple equation in $\theta_2$:

$$j_1 \sin\theta_2 + j_2 \cos\theta_2 + j_3 = 0 \tag{5.11}$$

where the $j_i$ can be written in terms of $d_i$, $\theta_1$ and $\theta_3$. With $\sin^2\theta_2 + \cos\theta_2^2 = 1$, we obtain two solutions for $\theta_2$. Collectively, we then obtain upto eight solutions for the angles. □

The angle solutions represent head orientations that produce the image. At this stage, we could do some rather basic anthropometric filtering by observing that the pitch angle cannot be so large that the chin penetrates the torso. Similarly, we could also impose constraints on the roll and yaw angles. The valid solutions can then be presented to the user from which one will be selected.

### 5.3.1 Recovering the Epipolar Geometry

Recall that $\mathbf{T}$ projects points from the body-centered coordinate system to the image plane. Given the calculated head orientation, we can recover $\mathbf{T}$, which has eleven unknowns. From the eight point correspondences at our disposal (four head plus four torso), we have an overdetermined set of sixteen equations in the elements of $\mathbf{T}$ which we solve for in a least squares sense using singular value decomposition.

The matrix $\mathbf{T}$ contains all information necessary to retrieve the epipole. $\mathbf{T}$ can be written in the form $(\mathbf{M} \mid \mathbf{c}) = (\mathbf{M} \mid -\mathbf{M}\mathbf{t})$ where $\mathbf{t}$ is the epipole [30]. Given this, $\mathbf{t}$ can be recovered as $-\mathbf{M}^{-\mathbf{1}}\mathbf{c}$.

### 5.3.2 Recovering Body Joint Coordinates

Consider any unknown world point $\mathbf{X} = [X \;\; Y \;\; Z \;\; 1]^T$ with known image point $\mathbf{x}$. Inverting the relationship $\mathbf{x} = \mathbf{T}\mathbf{X}$ (note that $\mathbf{T}$ is known to us now), we obtain a set of solutions for $\mathbf{X}$ parametrized by the unknown $Z$. This is simply the epipolar line of the image point in the body-centered coordinate system.

$$X = a + bZ \tag{5.12}$$

$$Y = c + dZ \tag{5.13}$$

where $a, b, c, d$ can easily be calculated in terms of elements of $\mathbf{T}$ and $\mathbf{x}$. Let $\mathbf{X}$ represent the right elbow which is connected to the right shoulder with known world coordinates $\mathbf{X_5} = [X_5\ Y_5\ Z_5\ 1]^T$. We also know the upper arm length, $L_{ua}$ from our model. We then have the following constraint:

$$(X - X_5)^2 + (Y - Y_5)^2 + (Z - Z_5)^2 = L_{ua}^2$$

$$= (a + bZ - X_5)^2 + (c + dZ - Y_5)^2 + (Z - Z_5)^2 \qquad (5.14)$$

which is a quadratic in $Z$, representing the two points of intersection of the epipolar line with the sphere of possible right elbow positions (see figure 5.3).

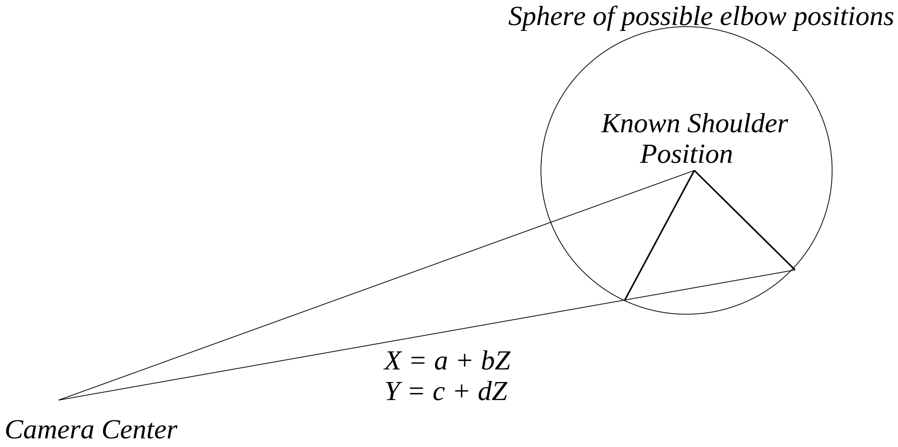


Figure 5.3: Possible Elbow Positions

These two solutions for the elbow represent the unavoidable forward/backward flipping ambiguity inherent in the problem. Once the correct right elbow position is found, the right hand can be found in the same manner. Similarly, we can obtain the 3D coordinates of all the other joints of the body. The interactivity in this solution process can be eliminated by having the user pre-specify the relative depths of the joints. In other words, before the solution process starts, each joint is assigned a boolean variable that specifies whether that joint is closer to the camera than its parent. Given that the user is specifying the point correspondences of body landmarks, this input imposes trivial additional burden. This idea is also used in [71]. Since we have already calculated the epipole location, we are able to calculate these distances readily.

### 5.3.3 Dealing with Unsolvable Cases

Computation of the head-orientation as well as the limb 3D locations involves the solution of quadratic equations. During experiments on real images and noisy synthetic images, in several cases, there were no solutions to one or more quadratics. This required a search for approximate solutions to the head orientation angles and to the limb positions. Both of these cases are described below.

#### Unsolvable Head Orientation Equations

Three different strategies were explored for arriving at an approximate solution to the head orientation angles. The correct and straightforward approach is to formulate a Lagrange optimization problem with the objective function being sum of squares of (5.7), (5.8) and (5.9) plus Lagrange multiples of the trigonometric identity constraints. However, proceeding in this manner results in a non-trivial system of polynomial equations in the sines and cosines of the three angles. A standard technique to solve such a system is to use results from polynomial ideal theory (i.e. Grobner basis [17]) to convert the system to triangular form which can then be solved easily. However, the equations overwhelmed all of the Grobner basis computation tools that were tried.

A second approach is to exclude (5.9) from the objective function and only retain (5.7) and (5.8) because $\theta_2$ does not occur in either of the two equations. The calculation of a Grobner basis for this formulation was found to be tractable and hence the finding of all the local minima of the equations was guaranteed. However, the Grobner basis computations were rather heavy and the performance was poor.

A third approach exploits the fact that the $(\theta_1, \theta_2, \theta_3)$ space is bounded by head-reachability constraints that could be set to anthropometrically accurate values. Nevertheless, a conservative approximation would be $[-\pi/2, \pi/2]$ for each angle. The fact that the angle space is bounded affords a brute-force search for $\theta_1$ and $\theta_3$ that produce minima for absolute values of the left hand sides of (5.7) and (5.8). We used $5^0 \times 5^0$ bins and calculated all local bin-wise minima. This approach was found to be much faster and produced a good approximate solution most of the time. However there is the possibility that some local minima would be missed by the search if they occur with other competing local minima in the same bin.

### Unsolvable Limb Positions

This case is easier to deal with than the head orientation case because of the absence of coupled transcendental equations. We considered two different approaches for this case (equation 5.14). One approach was to use Lagrange multipliers and find the optimal $X, Y, Z$ that satisfied the equation as closely as possible keeping the limb length fixed. A second approach was to use a scale $k$ such that the scaled limb-length ($kL_{ua}$ in this case) made the discriminant positive. We chose the latter approach because the scale factor effectively accounted for variations in the assumed and actual limb lengths.

## 5.4 Results

We evaluated the approach on synthetic and real images, the results of which we present below.

### 5.4.1 Synthetic Images

In the synthetic case, given that the error is zero for a perfect model and perfect image correspondences, we focussed on empirical error analysis. There are two sources of error: (1) differences between the assumed model and the imaged subject and (2) inaccuracies in the image correspondences. For five different viewpoints, and 500 random unknown poses per noise-level, we calculated the average error in full-body reconstruction (sum of squares of the difference between real and recovered 3D coordinates scaled by the head-to-foot distance) for Gaussian noise of zero mean and unit standard deviation and increasing noise intensities. The interactivity of the algorithm was eliminated by the evaluation program automatically choosing the head-orientation with minumum error among the solutions. There are three cases: noisy-model, noisy-image, and noisy-model with noisy-image. For image noise, we perturbed the image coordinates with the noise, scaled by the image dimensions which were taken to be

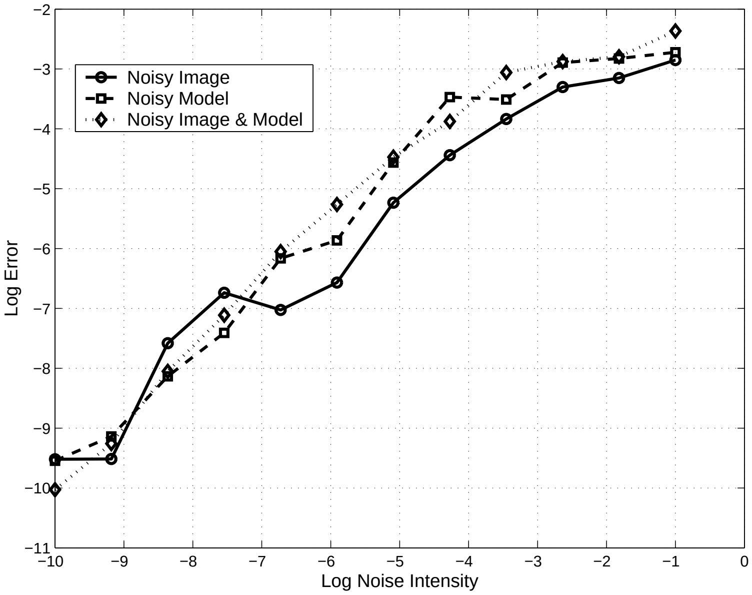


Figure 5.4: Error dependency on noise

those of the bounding box of the imaged body. For model-noise, the scale was the head-to-foot distance. Figure 5.4 shows the dependency. An important observation is that the reconstruction is more sensitive to errors in the model than in the image point correspondences. Interestingly, the curve for noisy-model with noisy-image error is almost the same as the noisy-model curve. We believe that this is because the model error swamps out image errors which are much smaller, especially at higher noise levels. Further, since the model and image errors are independent, errors cancel out in some cases. Nevertheless it can be seen that small errors in the model and image only produce small errors in the final reconstruction.

### 5.4.2 Real Images

We evaluated the qualitative performance of the approach on real images by using 3D graphics to render the reconstructed body pose and epipolar geometry. We used a 3D model derived photogrammetrically from front and side views of one subject and used the same 3D model for all images. There were two important problems we encountered with real images which we describe below:

#### Clothing

Clothing obscures the location of the shoulders and hips, the accuracy of which affects the head orientation computation. We addressed this problem with two strategies. First, given that the shoulders, hips and upper-neck form a planar homography we compute and use it: though we do not use the upper-neck as a feature point in (5.7), (5.8) and (5.9), we require the user to locate it. The homography is uniquely specified by four planar points. We use the five torso points to calculate the torso-plane-to-image homography in a least-squares sense, transform the torso-plane to the image using the

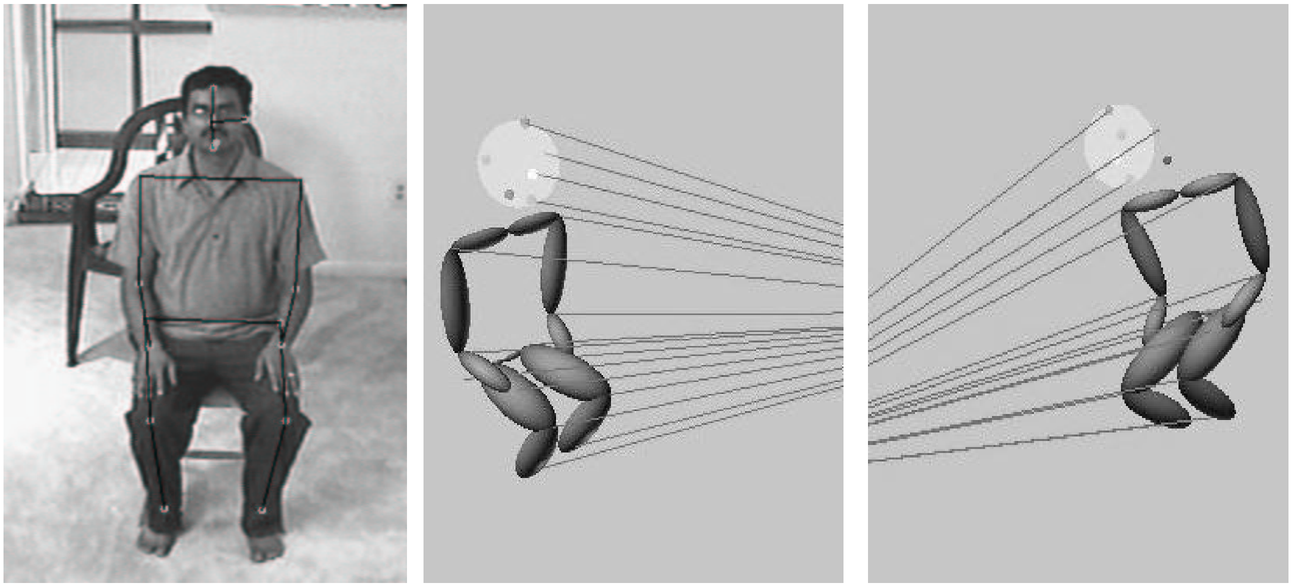


Figure 5.5: Person Sitting, Front-view

homography and use the transformed points as input rather than the user-specified points. Second, rather than requiring the user to locate the true right and left hip (about which the upper legs rotate), we just require their surface locations (i.e. ‘end-points’), which are easier to locate. The model stores the true centers of rotation of the legs as well as the surface locations.

### Upper-neck rigidity assumption

The skull rests on top of the cervical portion of the spinal cord. While we model this junction as a ball and socket joint, this effectively neglects the fact that the cervical vertebrae are free to rotate (although by a small amount and with a small radius) about the torso. To compensate for this, we take the skull center of rotation to be midway between the neck-base and upper-neck. This produced a significant improvement in the head-orientation recovery for cases where subjects lunged their head forward or backward in addition to rotating it.

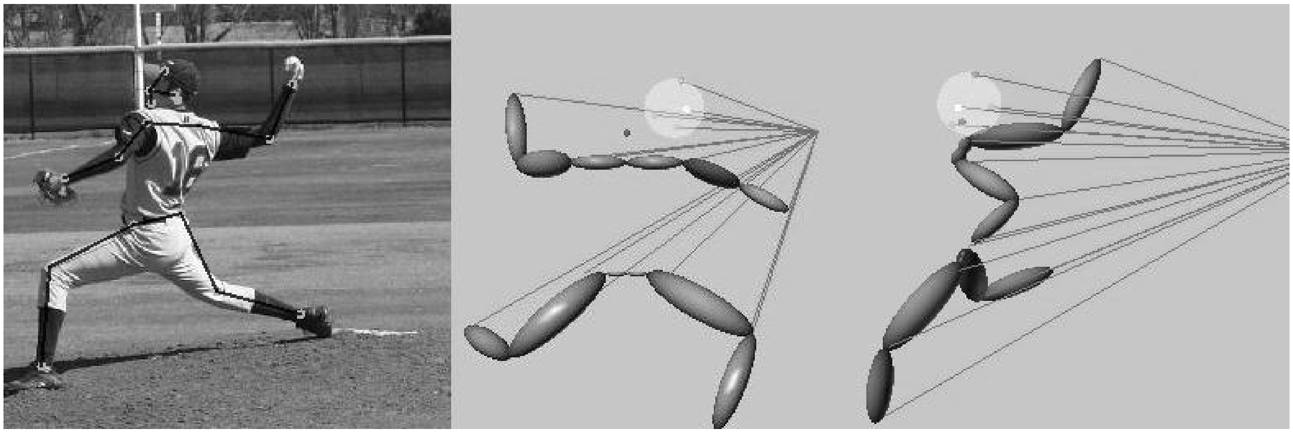


Figure 5.6: Baseball

For some images where these two effects were significant, we had to guess the true image coordinates three or four times before the algorithm returned realistic looking results. Figure 5.5 shows a subject sitting down and imaged from the front. Also shown in the image are user-input locations of various body landmarks. Beside the image are two rendered views of the reconstructed body pose and epipolar lines of the body landmarks as seen from views slightly oblique to the frontal view. The meeting of epipolar lines depicts the camera position which is accurately computed as lying in the front of the subject. Figure 5.6 shows a baseball pitcher and the reconstruction. Interestingly in this case, the camera is behind the torso of the subject and this fact is recovered well by the reconstruction. The algorithm is also able to handle a small amount of torso twist exhibited by the person. Figure 5.7 shows a subject sitting down with the hand pointed towards the camera, inducing strong perspective. This is an example of a viewpoint that cannot be handled by previous methods such as [6] and [71]. The reconstructed views correctly capture the fact that the camera is to the left and top of the subject. Figure 5.8 shows a subject skiing alongside two rendered views

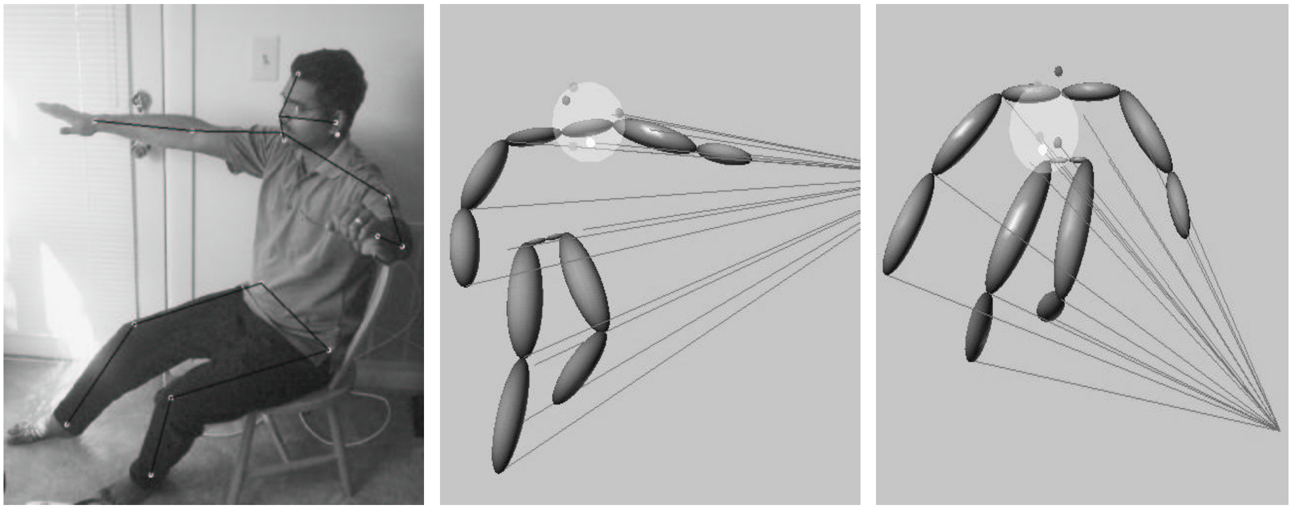


Figure 5.7: Person Sitting, Side-view

of the reconstructed body. The novel views of the reconstructions show that the body pose and camera position are both captured quite well.

There were several instances of real images where the algorithm failed to produce the correct reconstruction. Figure 5.9 shows a person riding a bicycle and two views of the reconstructed body. Though the recovery of the feet is reasonably correct, the overall pose of the hands is incorrect because the recovered pose shows a much larger distance between the hands than the true one. A more important failure is the fact that the epipole is wrongly detected to be to the right of the person when in fact it is to his left. One of the problems with this particular image is the near impossibility to reliably point out the persons hips (especially the right hip) due to clothing. Secondly the person's shoulders bend forward significantly and the torso rigidity assumption that the algorithm relies on, is violated. Figure 5.10 shows an even more drastic failure of the algorithm. Similar kinds of problems as in the previous example are encountered for this image. Notice that the subject's torso twists considerably in addition to a significant forward bend of the shoulders. The left hip position is obscured by the

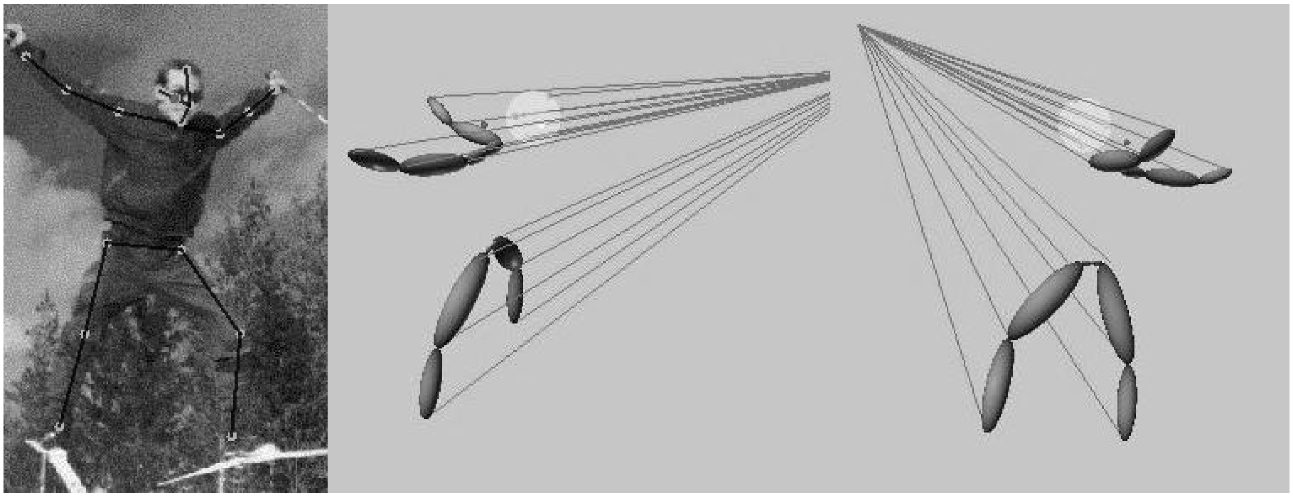


Figure 5.8: Person Skiing

hands as well. The pose as well as the epipole are both incorrectly calculated by the algorithm.

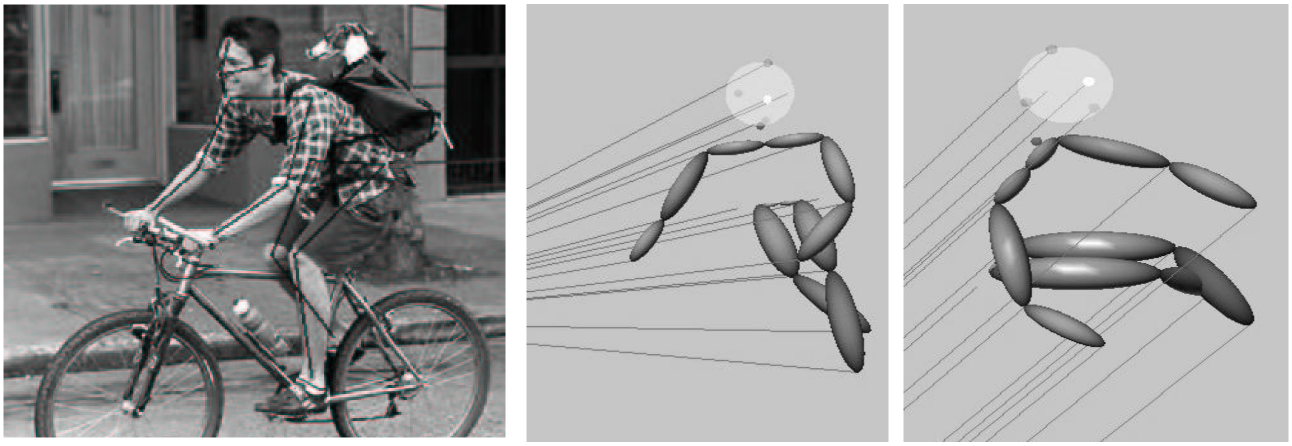


Figure 5.9: Incorrectly computed epipole and pose.

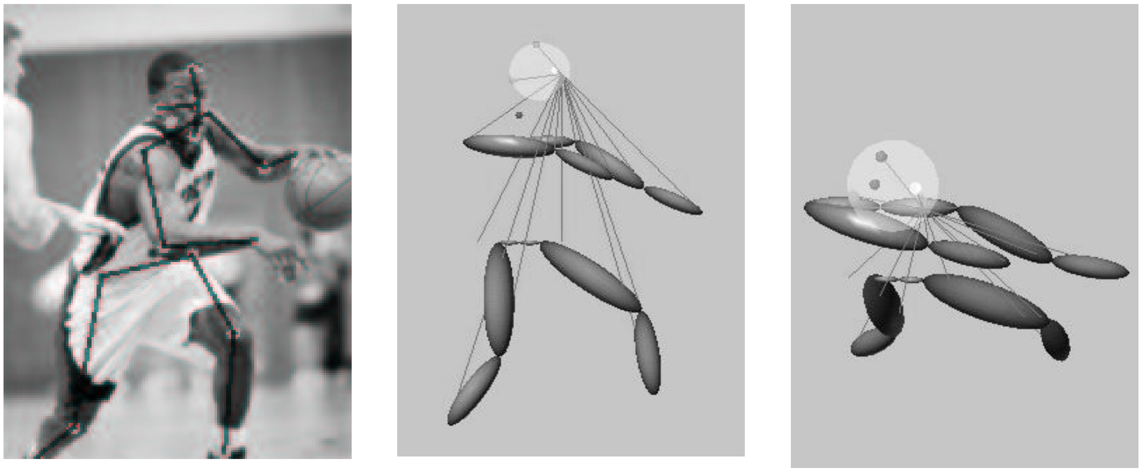


Figure 5.10: Incorrectly computed epipole and pose.

## 5.5 Summary

A method to calculate the 3D positions of various body landmarks in a body-centric coordinate system, given an uncalibrated perspective image and point correspondences in the image of various body landmarks - an important sub-problem of human body tracking and human motion capture, was presented. The small-torso-twist assumption utilized in the solution to the problem provides enough ground truth points on the torso and allows the use of ideas from 3D model based invariance theory resulting in a simple polynomial system of equations in the head orientation angles. Once these are solved, the epipolar geometry and calculation of all of the limb positions becomes possible. While theoretically correct given the assumptions, the method encountered specific problems when applied to real images, which were addressed by way of strategies to reduce error in input as well as the model. An empirical characterization of the influence of errors in the model and image point correspondences on the final reconstruction was presented. Effectiveness of the method on real images with strong perspective effects was also presented.

# Chapter 6

# Conclusions and Directions for Future Research

In summary, we presented contributions to the areas of human action recognition and human body pose estimation, with the underlying goal of making the solutions invariant to viewpoint. Below, we analyze the solutions we presented and discuss the directions in which they can be extended in future.

## 6.1 Action Recognition

In this section, we first compare the 2D and 3D action recognition approaches we presented in chapters 3 and 4. Following that, we discuss improvements and extensions that can be made to the approaches.

### 6.1.1 Comparison of the 2D and 3D approaches

Theoretically, based on the underlying assumptions, we know that the 2D approach is the least general while the full-3D approach the most general among the approaches. The restricted-3D approach lies in between the two approaches in terms of applicability. The fact that performance evaluations of the three approaches were carried out on the same set of data enables us to compare their accuracies. Interestingly, the 2D approach produced more accurate results than the 3D approaches while the restricted-3D approach fared the worst among the three. While the generality of the 3D approaches is better than the 2D approach, it was more difficult to find a good set of joint combinations that produced high inter-class and low intra-class variations for the 3D approaches. As we observed in chapter 4, we did not perform an extensive search for suitable joint combinations because the primary goal was to achieve robustness to viewpoint variations. We had to settle on a few joint combinations that produced reasonable results after a few trials. In contrast, choosing joint combinations for the 2D case was significantly easier for all the four actions evaluated. Further, the actions were particularly amenable to using the 2D approach because they exhibited significant planarity. These two factors partly explain the fact that the 2D approach performed better. In addition, the failure modes of the approaches, especially in relation to the viewpoints chosen, were very different (recall our discussion on failure modes in chapters 3 and 4) which further explains the difference in performances.

### 6.1.2 Future Research Directions in Human Action Recognition

The approaches we presented for human action recognition can be refined and made more mature by the following five ideas for further research.

#### Distance computations

We used a simple standard-deviation normalized distance measure for invariants (equations 3.2 and 4.17). These distances were used to classify the different actions. A better distance measure would take the probability density functions of the invariants into consideration and use Baye's rule to invert them to calculate the probability that the body is in a given pose given the observed invariant values. Such a probabilistic distance measure would help in increasing the classification rates.

### Threshold selection

Our choice of thresholds has not been rigorous. The optimum threshold for the 2D approach and the restricted 3D approach depends upon the viewpoint because the planarity assumption will not be perfectly valid. This makes it impossible to arrive at a globally optimal threshold. However, for a restricted class of viewpoints (say those at a fixed distance around the subject), one may be able to arrive at a single optimal threshold either analytically or empirically using a learning theory approach such as [54]. The empirical approach could also be used for the full-3D case where the threshold does not depend upon the viewpoint.

### Model Acquisition

Model building was done manually. A more systematic way would be to use an automatic search based method to pick the joint combinations with low intra-class and high inter-class variability. This approach would be similar to work by Campbell and Bobick [11] which would also enable us to weed out joints that are irrelevant to an action. However, we would be operating in 2D and 3D projective spaces rather than a Euclidean space that they operated in.

### Action Modeling Formalism

The action modeling formalism we used has been rudimentary. Rather than use dynamic time warping as we did for the 2D case, we could use more sophisticated models such as HMMs or Petri nets. None of the actions we modeled required the modeling of synchronization or concurrency, but coupled HMMs and Petri nets could be considered for other actions where we would need to model these aspects.

**Automatic detection on real image sequences**

The ultimate use of the action recognition approaches we presented in this thesis would be in a module in an end-to-end system where a real image sequence would be processed fully automatically for the detection of human actions captured in it. Though there are several low-level processing approaches that can be integrated with our approaches, Rosales and Sclaroff's approach [60] of mapping a human silhouette directly to a 2D body pose (i.e. location of joints in the image) appears to be the most promising because of its lack of restriction on the set of poses or viewpoints (provided their neural network is trained on a sufficient number of poses and viewpoints). It would be interesting to see how well the integrated system would perform on actions seen from different viewpoints.

## 6.2 Future Research Directions for Pose Estimation

The pose estimation approach we presented in chapter 5 can be extended and improved upon in several ways described below.

### 6.2.1 Image Cues

We only relied on user supplied input of image joint locations for the system. Color, texture and edge cues can provide supplementary information that can be used during the solution process. Especially during the head orientation computation stage, the solutions can be iteratively refined by using the synthesize-and-test strategy.

### 6.2.2 Use of Anthropometry Data

Anthropometry data can be used to create a better human body model for use in the system. Statistics can prune the set of solutions returned by the system as well because we can eliminate infeasible joint angles and body positions (e.g. we can avoid solutions that correspond to large unrealistic head rotation angles and body piercing limb positions).

### 6.2.3 Tracking

The ultimate use of the approach would be in a module that performs optical motion capture in the full perspective case. Bregler and Malik's [9] kinematic chain and optic flow based approach for human body tracking (which we discussed briefly in chapter 1) can be easily extended to the full-perspective case with our full-perspective initialization of the body model. Such an integrated system would be the logical next step in the practical use of the approach.

# Appendix A

# Proofs of Results Used in the Thesis

## A.1 Planar Homography

*Two views of a set of points on a plane are related by a homography,* $\mathbf{P}$, *a* $3 \times 3$ *matrix with eight effective degrees of freedom.*

Recall that the world-to-image transformation matrix $\mathbf{T}$ is of size $3 \times 4$ and it determines how a world point $\mathbf{X}$ is mapped to the image point $\mathbf{x}$. Given $\mathbf{x}$, $\mathbf{X}$ can only be known upto one unknown degree of freedom and can be written algebraically in terms of $\mathbf{x}$, the elements of $\mathbf{T}$ and an unknown scalar. The fact that the point lies on a plane eliminates this unknown degree of freedom. It is possible to proceed in this manner and prove the result by brute force algebra. However, Szeliski in [70] proves the result a bit more elegantly as follows:

$$\mathbf{x} = \mathbf{T}\mathbf{X} \tag{A.1}$$

Inverting the relationship, we can write

$$\mathbf{X} = \mathbf{T}^{*}\mathbf{x} + s\mathbf{t} \tag{A.2}$$

where $\mathbf{T}^{*}$ is the left inverse of $\mathbf{T}$ such that $\mathbf{T}\mathbf{T}^{*} = \mathbf{I_{3\times 3}}$ and $\mathbf{t}$ is the null space of $\mathbf{T}$ such that $\mathbf{T}\mathbf{t} = 0$. $s$ is a scalar that represents the unknown degree of freedom.

We know that the point $\mathbf{X}$ lies on a plane. Let us denote the normal of the plane as $\mathbf{n}$. Hence:

$$\mathbf{n^T X} = 0 \tag{A.3}$$

Substituting for $\mathbf{X}$ we have:

$$\begin{aligned}
&\mathbf{n^T}(\mathbf{T^*x} + s\mathbf{t}) = 0 \\
&\Rightarrow s = -\frac{\mathbf{n^T T^* x}}{\mathbf{n^T t}} \\
&\Rightarrow \mathbf{X} = \left[\mathbf{T}^* - \frac{\mathbf{t n^T T^*}}{\mathbf{n^T t}}\right]\mathbf{x} \\
&\Rightarrow \mathbf{X} = \tilde{\mathbf{T}}\mathbf{x}
\end{aligned} \tag{A.4}$$

where $\tilde{\mathbf{T}}$ is a $4 \times 3$ matrix.

From a different camera with transformation matrix $\mathbf{T}'$ the image of the point will be

$$\begin{aligned}
\mathbf{x}' &= \mathbf{T}'\tilde{\mathbf{T}}\mathbf{x} \\
&= \mathbf{P}\mathbf{x}
\end{aligned} \tag{A.5}$$

where $\mathbf{P}$ is a $3 \times 3$ matrix, the homography. Recall that $\mathbf{x}$ and $\mathbf{x}'$ are both in homogenous coordinates and hence $\mathbf{P}$ has only eight free parameters. If $\mathbf{x}$ and $\mathbf{x}'$ are real image coordinates where their third coordinate is unity, the above relation can be expressed as

$$\mathbf{x}' = \lambda\mathbf{P}\mathbf{x} \tag{A.6}$$

where $\lambda$ is an unknown scalar. The value $\lambda$ will be different for different $\mathbf{x}$.

## A.2 Projective Invariants of Five Points on a Plane

*Let $\mathbf{X_i}$, $i = 1..5$ be five points on a plane. and let $\mathbf{x_i}$ be coordinates of the points as seen in an image of them. The following quantities computed in the image plane, are* *invariant to the viewpoint:*

$$I_1 = \frac{M_{421}M_{532}}{M_{432}M_{521}} \quad , \quad I_2 = \frac{M_{421}M_{531}}{M_{431}M_{521}} \tag{A.7}$$

*where* $M_{ijk}$ *is the determinant* $|\mathbf{x_i} \;\; \mathbf{x_j} \;\; \mathbf{x_k}|$.

If we prove that the quantities as calculated from a different image of the points $\mathbf{x_i}'$ are the same as those calculated using $\mathbf{x_i}$, we are done. From the previous result, we know that $\mathbf{x_i'} = \lambda_i \mathbf{P} \mathbf{x_i}$ where $P$ is an unknown $3 \times 3$ matrix. Let $I_1'$ and $I_2'$ be the corresponding quantities from the different viewpoint and let $M_{ijk}' = |\mathbf{x_i}' \;\; \mathbf{x_j}' \;\; \mathbf{x_k}'|$

$$\begin{aligned} M_{ijk}' &= |\mathbf{x_i}' \;\; \mathbf{x_j}' \;\; \mathbf{x_k}'| \\ &= |(\lambda_i \mathbf{P} \mathbf{x_i}) \;\; (\lambda_j \mathbf{P} \mathbf{x_j}) \;\; (\lambda_k \mathbf{P} \mathbf{x_k})| \\ &= \lambda_i \lambda_j \lambda_k |\mathbf{P}||\mathbf{x_i} \;\; \mathbf{x_j} \;\; \mathbf{x_k}| \end{aligned}$$

Substituting the above into $I_1'$ and $I_2'$ we can verify that we get back $I_1$ and $I_2$ respectively because the scale factors and $|\mathbf{P}|$ cancel out. In fact, other choices for $i, j, k$ where the scale factors cancel out are also possible.

# BIBLIOGRAPHY

[1] J. Aggarwal and Q. Cai. Human motion analysis: A review. *Computer Vision and Image Understanding*, 73:428–440, 1999.

[2] K. Aizawa and T. Huang. Model-based image coding: Advanced video coding techniques for very low bit-rate applications. *Proceedings of the IEEE*, 83(2):259–271, 1995.

[3] A. Ali and J. K. Aggarwal. Segmentation and recognition of continuous human activity. *Proc. IEEE Workshop on Detection and Recognition of Events in Video*, July 2001.

[4] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *IEEE Transactions of Signal Processing*, 50(2):174–188, February 2002.

[5] K.E Astrom and L. Morin. Random cross ratios. *Proc. 9th Scand. Conf. on Image Analysis*, June 1995.

[6] C. Barron and I. A. Kakadiaris. Estimating anthropometry and pose from a single uncalibrated image. *Computer Vision and Image Understanding*, 81, 2001.

[7] A.F. Bobick and J.W. Davis. The recognition of human movement using temporal templates. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(3):257–267, March 2001.

[8] M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complex action recognition. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 994–999, 1997.

[9] C. Bregler and J. Malik. Tracking people with twists and exponential maps. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 1998.

[10] J. B. Burns, R. S. Weiss, and Riseman E. M. The non-existence of general case invariants. In J. L. Mundy and A. Zisserman, editors, *Geometric Invariance in Machine Vision*. MIT Press, Cambridge, MA, 1992.

[11] L. Campbell and A. Bobick. Recognition of human body motion using phase space constraints. *Proc. International Conference on Computer Vision*, pages 624–630, 1995.

[12] L. W. Campbell, D. A. Becker, A. Azerbayejani, A. Bobick, and A. Pentland. Invariant features for 3-d gesture recognition. *2nd Int. Conf. on Automatic Face- and Gesture-Recognition, Killington, Vermont*, 1996.

[13] C. Cedras and M. Shah. Motion-based recognition: A survey. *Image and Vision Computing*, 13(2), 1995.

[14] O. Chomat and J.L. Crowley. Probabilistic recognition of activity using local appearance. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages II: 104–109, 1999.

[15] R.T. Collins et al. A system for video surveillance. Technical Report CMU-RI-TR-00-12, Carnegie Mellon University, 2000.

[16] T. H. Cormen, C. E. Leiserson, and R. L Rivest. *Introduction to Algorithms*. MIT Press/McGraw-Hill, New York, 1990.

[17] D. A. Cox, J. B. Little, and D. O'Shea. *Using Algebraic Geometry*. Springer Verlag, New York, 1998.

[18] R. David and H. Alla. *Petri Nets and Grafcet*. Prentice Hall, 1991.

[19] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification*. Wiley, 2001.

[20] H.L. Eng et al. An automatic drowning detection surveillance system for challenging outdoor pool environments. *Proc. International Conference on Computer Vision*, pages 532–539, 2003.

[21] R. Fablet and M. Black. Automatic detection and tracking of human motion with a view-based representation. *Proc. European Conference on Computer Vision*, May 2002.

[22] O Faugeras. *Three-Dimensional Computer Vision, A Geometric Viewpoint*. The MIT Press, 1993.

[23] H. Fujiyoshi and A. Lipton. Real-time human motion analysis by image skeletonization. *Fourth IEEE Workshop on Applications of Computer Vision*, pages 15–21, 1998.

[24] D. Gavrila and L Davis. 3-d model-based tracking of humans in action. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 73–80, 1996.

[25] D. M. Gavrila. The Visual Analysis of Human Movement. *Computer Vision and Image Understanding*, 73(1):82–98, 1998.

[26] K. Grauman, G. Shakhnarovich, and T. Darrell. Inferring 3d structure with a statistical image-based shape model. *Proc. International Conference on Computer Vision*, 2003.

[27] Y. Guo, G. Xu, and S. Tsuji. Tracking human body motion based on a stick figure model. *Journal of Visual Communication and Image Representation*, 1994.

[28] I. Haritaoglu, D. Harwood, and L. Davis. Ghost: A human body part labeling system using silhouettes. *Proc. International Conference on Pattern Recognition*, pages 77–82, 1998.

[29] I. Haritaoglu, D. Harwood, and L. Davis. W4: Who, when, where, what: A real time system for detecting and tracking people. *Proc. of the Third Face and Gesture Recognition Conf.*, pages 222–227, April 1998.

[30] R. Hartley. Chirality. *International Journal of Computer Vision*, 26(1):41–61, 1998.

[31] A.F. Horadam. *A Guide to Undergraduate Projective Geometry*. Springer-Verlag, 1970.

[32] M. K. Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory IT(8)*, 1962.

[33] Credo Interactive Inc. *http://www.charactermotion.com*.

[34] M. Isard and M. Blake. Condensation - conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.

[35] G. Johannson. Visual perception of biological motion and a model for its analysis. *Perception and Psychophysics*, 14(2):201–211, 1973.

[36] I. T. Jolliffe. *Principal Component Analysis*. Springer Verlag, New York, 1986.

[37] S. Ju, M. Black, and Y. Yacoob. Cardboard people: A parameterized model of articulated motion. *2nd Int. Conf. on Automatic Face- and Gesture-Recognition, Killington, Vermont*, pages 38–44, 1996.

[38] Y. Kameda and M. Minoh. A human motion estimation method using 3-successive video frames. *International Conference on Virtual Systems and Multimedia*, 1996.

[39] H. J Lee and Z. Chen. Determination of 3d human body posture from a single view. *Computer Vision, Graphics and Image Processing*, 30, 1985.

[40] M.K. Leung and Y.H. Yang. First sight: A human-body outline labeling system. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(4):359–377, April 1995.

[41] A. J. Lipton. Local application of optic flow to analyze rigid versus non-rigid motion. *ICCV Workshop on Frame-Rate Applications*, 1999.

[42] D. Marr and H. K. Nishihara. Representation and recognition of the spatial organization of three dimensional shapes. *Proceedings of the Royal Society, London*, B:200:269–274, 1978.

[43] S. Maybank. Probabilistic analysis of the cross-ratio to model based vision. *International Journal of Computer Vision*, 16:5–33, 1995.

[44] S. J. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler. Tracking groups of people. *Computer Vision and Image Understanding*, 80:42–56, 2000.

[45] A. Menache. *Understanding Motion Capture for Computer Animation and Video Games*. Morgan Kaufmann, 2000.

[46] T. Moeslund and E. Granum. A survey of computer vision based human motion capture. *Computer Vision and Image Understanding*, 81(3), March 2001.

[47] C. Mohan, A. Papageorgiou and T. Poggio. Example-based object detection in images by components. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(4):349–361, 2001.

[48] R. Mohr and B. Triggs. Projective geomety for image analysis. In *ISPRS*, 1996.

[49] J.L. Mundy and A Zisserman. *Geometric Invariance in Machine Vision*. MIT Press, Cambridge, MA, 1992.

[50] J.L. Mundy, A. Zisserman, and D. Forsyth. *Applications of Invariance in Computer Vision*. Springer-Verlag, 1994.

[51] C. Myers, L. Rabiner, and A. Rosenberg. Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. *IEEE Transactions on Acoustic, Speech and Signal Processing*, 28(6):623–635, 1980.

[52] Y. Nam, K. Wohn, and H. Lee-Kwang. Modeling and recognition of hand gestures using colored petri nets. *IEEE Transactions on Systems, Man and Cybernetics-Part A*, 29(5):514–521, September 1999.

[53] Georgia Inst. of Technology. Motion capture repository. *ftp.cc.gatech.edu/pub/gvu/cpl/walkers*.

[54] V. Parameswaran, P. Burlina, and R. Chellappa. Performance analysis and learning approaches for vehicle detection and counting. *Proc. IEEE Conference on Acoustics, Speech and Signal Processing*, 1997.

[55] V. Parameswaran and R. Chellappa. View invariants for human action recognition. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2003.

[56] R. Polana and R.C. Nelson. Low level recognition of human motion. In *Proc. IEEE Workshop on Motion of Rigid and Articulated Objects*, pages 77–82, 1994.

[57] L.R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–285, 1989.

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

[59] K. Rohr. Incremental recognition of pedestrians from image sequences. In *CVPR*, pages 8–13, 1993.

[60] R. Rosales and S. Sclaroff. Inferring body pose without tracking body parts. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, June 2000.

[61] C. A. Rothwell. *Object Recognition Through Invariant Indexing*. Oxford Science Publications, 1995.

[62] C. A. Rothwell, A. Zisserman, D. A. Forsyth, and J.L. Mundy. Canonical frames for planar object recognition. *Proc. European Conference on Computer Vision*, pages 757–772, 1992.

[63] S. M. Seitz and C. R Dyer. View-invariant analysis of cyclic motion. *International Journal of Computer Vision*, 25:1–25, 1997.

[64] C. Sminchisescu and B Triggs. Kinematic jump processes for monocular 3d human tracking. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2003.

[65] S. Srinivasan and R Chellappa. Noise-resilient estimation of optical flow by use of overlapped basis functions. *Journal of the Optical Society of America*, 16(3):493–507, March 1999.

[66] C. Stauffer. Automatic hierarchical classification using time-based co-occurrences. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 1999.

[67] C. Stauffer and W Grimson. Learning patterns of activity using real-time tracking. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(8):747–757, 2000.

[68] P.F. Stiller, C.A. Asmuth, and C.S. Wan. Invariant indexing and single view recognition. *Proc. DARPA Image Understanding Workshop*, pages 1423–1428, 1994.

[69] T. Syeda-Mahmood and A. Vasilescu. Recognizing action events from multiple viewpoints. *Proc. IEEE Workshop on Detection and Recognition of Events in Video*, July 2001.

[70] R. Szeliski. Image mosaicing for tele-reality applications. In *IEEE Workshop on Applications of Computer Vision*, pages 44–53, 1994.

[71] C. Taylor. Reconstructions of articulated objects from point correspondences in a single image. *Computer Vision and Image Understanding*, 80(3), 2000.

[72] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *International Journal of Computer Vision*, 9(2):137–154, November 1992.

[73] Carnegie Mellon University. Motion capture repository. *http://mocap.cs.cmu.edu*.

[74] L. Wang, W. Hu, and T. Tan. Recent developments in human motion analysis. *Pattern Recognition*, 36(3):585–601, March 2003.

[75] D. Weinshall. Model-based invariants for 3d vision. *International Journal of Computer Vision*, 10(1):27–42, 1993.

[76] I. Weiss. Geometric invariants and object recognition. *International Journal of Computer Vision*, 10(3):207–231, 1993.

[77] I. Weiss and M. Ray. Model-based recognition of 3d objects from single images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23, February 2001.

[78] G. Welch and G. Bishop. An introduction to the kalman filter. Technical Report TR 95-041, University of North Carolina at Chapel Hill, 1995.

[79] O. Whicher. *Projective Geometry, Creative Polarities in Space and Time*. Rudolf Steiner Press, 1971.

[80] C. Wren, A. Azerbayejani, T. Darrell, and A. Pentland. Pfinder: Real-time tracking of the human body. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):780–785, 1997.

[81] Y. Yacoob and M. J. Black. Parameterized modeling and recognition of activities. *Proc. International Conference on Computer Vision*, pages 120–127, 1998.

[82] J. Yamato, J. Ohya, and K. Ishii. Recognizing human action in time sequential images using hidden markov model. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 56–62, 1992.

[83] V. M. Zatsiorsky. *Kinetics of Human Motion*. Human Kinetics, Champaign, IL, 2002.