Monocular Human Pose Estimation

Benjamin Laxton
blaxton@cs.ucsd.edu

University of California, San Diego
Advised by Prof. David Kriegman

Abstract

Automatic human motion capture is an important and significant problem in the computer vision community. A successful system may have many applications including inexpensive motion capture and analysis in unconstrained environments, human-computer interfaces, and automatic surveillance systems. This work focuses on an important sub-problem in computer vision based motion capture: monocular human pose estimation. This problem is characterized by methods that do not rely on temporal information or multiple images, but instead try to estimate a person's pose by measuring a single image. This sub-problem is integral to the larger goal of motion capture since human trackers inevitably drift and must be (re)initialized. Additionally, single camera environments abound and a monocular pose estimation method could be deployed in a wide range of settings. The foundations of research in monocular human pose estimation will be presented and recent advancements in the field will be discussed. The various approaches will be compared to one another and presented in the larger context of human motion capture. The analysis culminates with broad insights and suggestions for future work in this area.

1. Introduction

The analysis and recovery of human body motion has received a significant amount of attention in the computer vision research community in recent years. This interest is due, in part, to the interesting and challenging nature of this problem domain, but also results from the potentially far-reaching impacts of a successful system. As computers become an increasingly ubiquitous presence in our world it is important to develop natural ways to interact with them. Automatic human motion analysis is one obvious modality for this type of interaction. Additionally, with the advent of mobile robots, and especially humanoid robots, automatic understanding of human motion is essential for operation in an environment populated by people. Additional application areas include improved surveillance systems and inexpensive motion capture for entertainment and biomechanical research purposes. In particular, successful monocular human pose estimation techniques, the focus of this work, could significantly lower the entry barrier to many of these specialized application areas - allowing anyone with a computer and a video camera to capture motion data and develop and use new applications in this domain.

The general problem of computer vision based automatic human motion capture has been the topic of study for hundreds of researchers over the past decade. Despite the significant work that has gone into this problem, it remains far from solved. There are several sources of significant challenges to this problem such as viewing variability, pose variability, self-occlusion and the inherent high degree of freedom of the problem. In general settings, people wear many different types of clothing with different texture and movement characteristics. This, combined with variations in lighting and viewing direction create many difficulties in defining useful features for human pose estimation in general settings. Additionally, even a modest 3D model of the human body that does not take small movements such as hand, head and facial variations into account has more than 50 degrees of freedom in the pose parameter space.

To address these challenges an array of methods have been proposed that draw from the computer vision and machine learning research fields. Classic model-based approaches derived from early work on object detection have proven successful to some degree in locating people in images and labeling the positions of body parts. A surprising and recent advancement in this area is found in a class of model-free approaches that make use of enormous datasets and cutting edge machine learning techniques to estimate human pose parameters directly from features computed over an image. There is, however, significant room for improvements. Current limitations are due largely to the underlying problems of reliable feature extraction and in-

terpretation, as well as scaling to very high dimensional state spaces. These problems are fundamental to computer vision in general and suggest that human pose estimation may be a good target research area that has clear application driven goals and that requires fundamental vision tasks to be solved for success.

It is informative to note that the problem of human pose estimation has strong parallels to research in the area of hand pose estimation. A recently published review covering the field of research on hand pose estimation makes this observation as well [6]. Both problems address parameter estimation for a high degree of freedom model. Unsurprisingly, methods from each respective area have been borrowed by the other on several occasions [3, 6]. This example illustrates the wider applicability of methods developed for human pose estimation. Thus, successes in the human pose estimation community can have profound effects on this research area as well.

In the remainder of this paper several recent methods for pose estimation will be discussed and specific instances of each method will be covered. The goal is to identify parallels, limitations and orthogonal characteristics that are useful for analyzing the current limitations of human pose estimation techniques and offering suggestions for possible avenues of future work in this area. Section 2 will define the overall problem, put the scope of this work in a broader context and define a taxonomy for describing the current methods for monocular human pose estimation. Section 3 will describe the overarching ideas in model-based human pose estimation techniques and describe details of current implementations. Section 4 will describe the general form of model-free human pose estimation techniques and details of several current algorithms. Finally, a comparison between the various methods, current limitations and suggestions for future work on human pose estimation techniques will be given in Section 5.

2. Statement of Problem and Scope

Techniques for human motion capture seek to accurately estimate the body part positions, called pose, of a moving human body. Human pose can be described by various parameterizations. Suppose that the goal is to approximately describe a human pose in terms of the positions of all large body parts, torso, arms, legs and head, in 3 dimensions (3D). Two obvious parameterizations to achieve this are to describe relative part positions in terms of 3D Euler angles, or alternatively, to describe the joint and end point positions of parts in terms of 3D cartesian coordinates. These two possible parameterizations are shown in Figure 1. Notice that the model of a human body specifies a kinematic chain where the connections of body parts can be described as a parent-child relationship. For example, assume the torso is the root node in the kinematic chain. Then the torso is

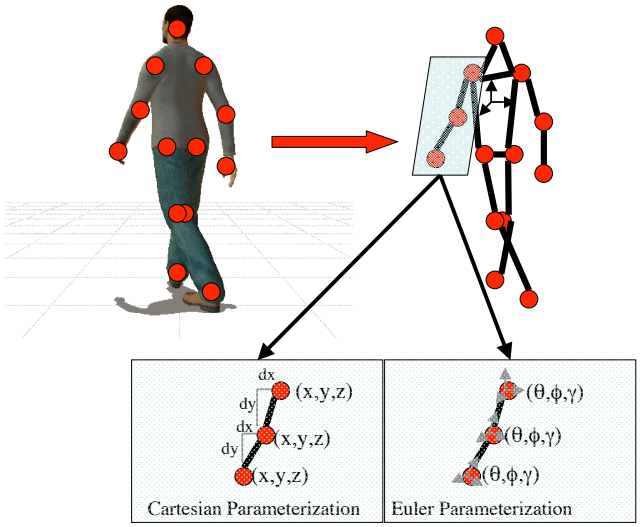

Figure 1. The human body can be modeled as a kinematic chain. Pose parameters for a given body part are defined relative to the parent body part in the kinematic chain representation. Possible parameterizations include relative Cartesian coordinates and relative Euler angles.

the parent of the upper arms, which are in turn the parents of the lower arms. In this way, the position of a particular body part is partially determined by the position of its parent body part and partially by its own pose parameters. Thus, it is often convenient to describe the pose parameters of a body part with respect to the local coordinate frame determined by its parent.

Under these parameter models a human pose is specified by the values of the underlying parameters. Cast in this framework, the goal of human pose estimation is to assign the appropriate parameters to describe a given body pose. Typically the input to a human motion capture system is a video sequence, or a collection of calibrated video sequences, whose frames capture a person exhibiting some motion. The challenge is then to locate the individual body parts in the video frame(s) and estimate the pose parameters from the relative locations. In commercial motion capture (MoCap) systems, the task of locating body parts is made simpler by placing markers on the joints or body parts of a human subject prior to recording the motion. These markers uniquely identify each body part, essentially solving the correspondence problem. In contrast, computer vision techniques for human motion capture, as defined here, are characterized by only using typical video frames as input. Furthermore, no special markings on the human subject are present. Under these conditions the body part location and correspondence problem becomes much more difficult.

Although typical human bodies can be represented by a common underling parameter model, individuals will have different body part sizes, limb lengths, and exhibit different

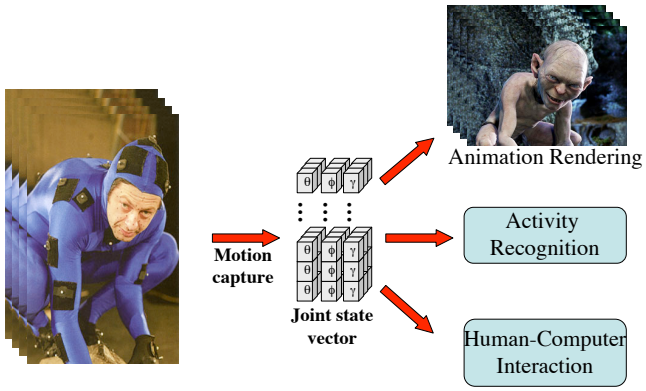

Figure 2. Motion capture is the process of relating image data to pose data. The image data may be constrained, as above, with markers used in commercial motion capture systems, or it may consist of normal video frames. The goal is to estimate the pose parameters of the body over time from these video frames. Applications such as animation rendering, activity recognition and human-computer interaction interfaces may be built on top of the motion capture representation.

ranges of motion depending on flexibility and body shape. These individual parameters may be pre-specified or automatically estimated from the data depending on the sophistication of the MoCap system. The ultimate goal of a computer vision motion capture system is to provide a fully automatic, accurate and robust estimate of a person's pose parameters.

2.1. Taxonomy of Previous Work

Computer vision based motion capture is a large problem domain and previous research spans several subproblems. Although the distinctions between sub-problems in human motion capture are somewhat arbitrary, and certainly blur at the intersection boundaries, it is useful to describe a taxonomy to categorize various approaches. The following taxonomy broadly categorizes approaches to the human motion capture problem into 4 classes:

- Detection and Initialization: The sub-problem of detection and initialization seeks to specify when a person is present in an image or video frame and in some cases give a course positional estimate. This subproblem may also include camera calibration in calibrated systems.

- Pose Estimation: The focus of systems for pose-estimation is to identify how a human body and possibly its constituent parts are positioned. The target may be a very fine-grained pose estimate that gives the relative joint angles between each part of the human body. Pose estimation can be an integral part to a tracking loop providing an (re)initialization to account for drift, or a completely separate process.

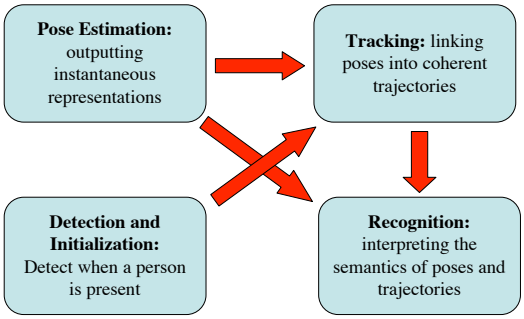

Figure 3. Pose estimation procedures can be used to initialize or reinitialize tracking algorithms. Recognition procedure may use instantaneous pose data or tracking data for interpretation.

- Tracking: This sub-area addresses the problem of matching corresponding body parts over a sequence of video frames. Many cues are used in tracking systems such as kinematic models and common motion models. The output of the tracker may be either 2D or 3D depending on the target application and assumptions about the environment and motion in the scene.

- Recognition: Human motion recognition systems aim to estimate the intent of the humans in the videos through their motions and actions. Depending on the target application, recognition systems may attempt to recognize and label large-scale motions such as walking, running etc., recognize specific gestures for HCI applications, or recognize some set of actions within a targeted context. Recognition, with a few exceptions, is a post-processing step that makes use of tracking and/or pose estimation data as the input and learns a mapping to a set of output action labels.

A graphical depiction of the data-flow between these sub areas is given in Figure 3. Techniques may be further distinguished by the granularity of the output representation, the use of and type of kinematic model, whether the system outputs 2D or 3D estimates, and the speed of processing. An additional way that methods for human motion capture can differ is in the assumptions made about the operating environment, the expected motion types, appearance assumptions etc. A list of common assumptions, adapted from those specified by Moeslund and Granum [17], is shown in Table 1. Generally, in current systems for human motion capture there is an inverse relationship between the performance and the number of assumptions made. Thus, when evaluating motion capture systems it is important to not only look at performance in terms of speed, accuracy,

Scenario Assumptions	Environment Assumptions	Subject Assumptions
Subject is always viewable	Constant lighting	Known subject
Stationary camera	Static background	Markers placed on subject
Only one viewable subject	Uniform background	Known clothing appearance
Only forward facing poses	Known camera parameters	Tight clothing
No occlusions	Known global body position	
Ground plane is flat		

Table 1. Commonly made assumptions in pose estimation methods.

and descriptiveness, but also in terms of the number of limiting assumptions.

2.2. Monocular Pose Estimation

The focus of this paper is on the sub-area of fully articulated pose estimation from monocular images. Specifically, this problem is, given a single image, output an estimate of the body-joint state vector. The image may be a snapshot or single frame from a video sequence. Monocular pose estimation is difficult because it requires a mapping from a possibly noisy and ambiguous image space to a low-dimensional, highly structured pose-space representation. Nevertheless, we know the problem is solvable because people are quite good at performing this task. Additionally, this sub-problem is important for solving human motion capture for several reasons:

- Can be used as an initialization for pose tracking methods and for re-initialization to account for drift.
- Does not make assumptions about the patterns, speed or continuity of motion, making it more amenable to many real applications.
- Provides a compact, yet detailed description of the body pose that is useful for analysis and HCI applications.
- May not require a calibrated environment allowing for deployment in a wide array of settings.

To place the problem of monocular human pose estimation in the broader arena of computer vision, consider a very general description of image formation given in Equation 1.

$$\Phi : W_c \mapsto I_c \tag{1}$$

Φ, a general image formation process that projects the world, W_c, onto an image, I_c, contains many processes that contribute to how an image is formed such as lighting, perspective model, camera sensitivity to light, 3D scene configuration and so forth. When a person is present in the scene, buried within all the other contributing factors are the 3D body configuration parameters that specify the person's pose. Estimating these parameters independently of all other processes is the goal in human pose estimation.

For this reason, automatic human pose estimation techniques contain some feature extraction component that attempts to filter the input image so that the contribution of all parameters except those of the human 3D configuration are removed. Assume that we call the parameters that specifies a body pose Θ and the feature space of some image filtering process $\mathcal{X}$. The mapping from configuration space to some image feature space is defined in Equation 2a.

$$M(\theta_{m\times 1}) \mapsto x_{n\times 1} \tag{2a}$$

$$M^{-1}(x_{n\times 1}) \mapsto \theta_{m\times 1} \tag{2b}$$

where $\theta_{m\times 1} \in \Theta$ is an instance in pose space and $x_{n\times 1} \in \mathcal{X}$ is an instance in feature space. Similarly, the inverse process that maps from some image feature space to human configuration space is given in Equation 2b. It is this inverse mapping process that human pose estimation techniques attempt to capture.

Notice that this inverse mapping is difficult to capture since it is both one-to-many and, depending on the feature space, many-to-one. Specifically, self-occlusions of various body parts, and reflectively symmetric poses can result in instances identical in feature space, but with different parameters in Θ. Examples of these cases are shown in Figure 4. Similarly, depending on the effectiveness of the feature space used, differences in clothing, overall body shape and lighting may produce different instances in feature space, $\mathcal{X}$, that all have the same underlying generating parameters in Θ. The challenge for human pose estimation techniques is to define a reasonable feature space and then estimate this ill-defined inverse mapping.

Approaches to the pose estimation problem can be broadly categorized as either *model-based* or *model-free*. Model-based approaches specify an underlying kinematic model, often a rough approximation of the skeleton, and use this model in conjunction with image measurements to estimate the pose that best fits the model and the observed image features. An illustration showing the human skeleton and several body models for pose estimation is given in Figure 5. Conversely, model-free approaches assume no underlying kinematic model, but instead aim to learn a data mapping that best explains input-output pairs provided as training data that, hopefully, generalizes well to unseen cases. In

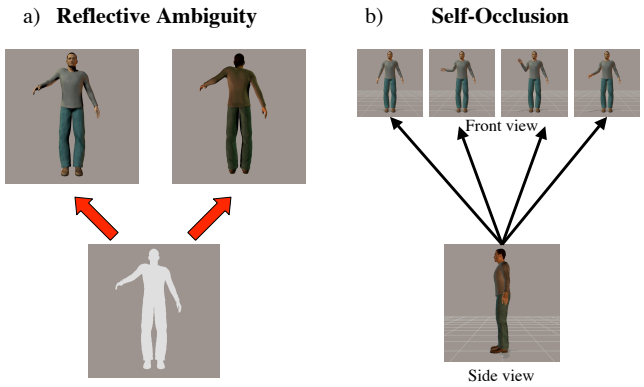

Figure 4. One-to-many mappings can occur as a result of reflective ambiguity in feature space (a) or from self-occlusions (b).

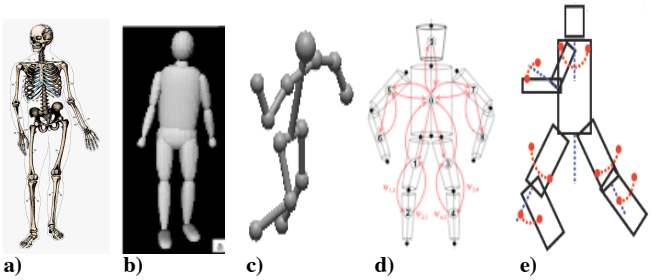

Figure 5. For comparative purposes a human skeleton is displayed with several body models used for human pose estimation. a) The human skeleton *(www.m-w.com)*. b) 3D model [12]. c)Kinematic skeleton model [23]. d)A 'loose-limbed' model [22]. e)A 2D puppet model [18].

model-free approaches the input may be a collection of image features and the desired output a vector describing the relative locations of each body part. Both model-based and model-free approaches have their merits and recent work from both approaches will be explored.

The rest of this paper will cover the techniques, assumptions and performance characteristics of recent research systems for monocular pose estimation. Section 3 will be devoted to model-based approaches and their limitations. Section 4 will cover model-free approaches. Comparisons of the performance, limitations and assumptions common in each approach will be discussed in Section 5 and recommendations for future avenues of research in monocular pose estimation will be presented.

3. Model-Based Pose Estimation Techniques

Model-based human pose estimation techniques are characterized by explicitly specifying a model that relates constraints between body parts. These constraints are not limited to the kinematic constraints of jointed, articulated structures, but also may include appearance constraints, scale constraints and angular constraints among others. Generally, in this framework pose estimation proceeds in an iterative match-update paradigm whereby the body parts in the model are 'posed' to match those of a subject in an input image.

Model-based human pose estimation can proceed in a top-down or a bottom-up fashion. In a top-down technique, the body model as a whole is evaluated against features computed on an image, whereas in a bottom-up technique candidate body parts are first detected and then built into a plausible configuration. Most human pose estimation frameworks proceed in a bottom-up fashion [9, 11, 12, 18, 19, 22] although the distinction can be blurry and some techniques exhibit properties of both [7].

Examples of both 2D and 3D body models are shown in Figure 5. The 2D body model is often described as a cardboard-cutout doll, and the 3D model can be viewed as analogous to an artists wooden manikin or a childrens toy push-puppet depending on whether the dependencies between joints are elastic. More precisely, the 3D model treats body parts as generalized cylinders, while the parts in a 2D model are the projection of 3D cylinders on the image plane yielding generalized rectangles. While all techniques use a model similar in spirit to the ones shown, they may differ in the way the parameterization, the enforced constraints, the number of degrees of freedom etc. The key challenge in model-based human pose estimation lies in efficiently searching for the parameters that minimize the distance between the model and the image observations since the parameter space may be very high dimensional and image measurements are often noisy. The following sections will discuss the details of several model-based approaches for human pose estimation.

3.1. Puppet Models

'Puppet models' are 3D or 2D kinematic models that represent a human body as a collection of individual parts connected at 'joint' locations. The parts in a puppet model generally correspond to human body parts and the joints correspond to joints between the body parts. Each part of the puppet model contains a specification of how the corresponding body part is expected to look in an image. Similarly, the joints in the puppet model specify allowable and expected pose configuration parameters. The part-based decomposition of puppet models takes advantage of independence between various pose parameters to more succinctly and efficiently model human pose space. The driving idea behind techniques for human pose estimation that use explicitly defined puppet models is to find the configuration of the model whose body part descriptions best match the image and whose joint parameters fall within reasonable limits defined by human kinematic constraints. Finding the best solution under these criteria is a difficult problem. Several methods for estimating the parameters are described next.

3.1.1 Global Optimization in a Discretized Space

Part-based models for general object recognition in images date back, at least, to work by Fischler and Elschlager that defined the concept of pictorial structures [8]. Pictorial structures simultaneously capture individual part appearance as well as body joint configural constraints in a graphical model. This general model framework was extended by Felzenszwalb and Huttenlocher to efficiently handle the special case of 2D human pose detection and estimation [7]. A human body model can be represented as a graphical model where the body-parts are the nodes, and kinematic constraints are represented as edges between nodes.

If the only constraints included in the model are those of jointed body-parts, the graphical model that represents a human body conforms to a tree structure. From this observation an efficient dynamic programming approach to finding the *globally* optimal estimate given part-image matching functions and model parameters is possible. One caveat to providing a globally optimal solution with respect to model agreement in reasonable computational time is that the parameter space must be discretized. A discretization for the location of a part may be over all $\{x, y, \omega\}$ triples, representing discretization of 2D translational and rotational space, at some spatial resolution.

Under the assumptions of a tree-structured graphical model and a discretized parameter space, pose estimation can be cast as an energy minimization function over the model configuration, $L = (l_1, \ldots, l_n)$, where l_i specifies the location for part i in the model. The energy minimization takes the form given in Equation 3.

$$L^* = argmin_L \left(\sum_{\langle v_i, v_j \rangle \in E} d_{ij}(l_i, l_j) + \sum_{v_i \in V} m_i(I, l_i) \right) \tag{3}$$

Here $d_{i,j}(l_i, l_j)$ specifies a distance function between the positions of neighboring parts i and j, which captures kinematic constraint parameters. Furthermore, $m_i(I, l_i)$ gives a measure of how well the image, I, matches the appearance model of part i in location specified by l_i.

Now assume that the graphical model that specifies the human body kinematics is given as a graph $G = (V, E)$ where $v_r \in V$ is a specified root node of the model. Furthermore, the children of a node v_i that are directly connected by an edge in the model are specified by C_i. This energy minimization can be recursively defined over the model as in Equation 4.

$$B_j(l_i) = \min_{l_j} \left(d_{ij}(l_i, l_j) + m_j(I, l_j) \right) \tag{4a}$$

$$B_j(l_i) = \min_{l_j} \left(d_{ij}(l_i, l_j) + m_j(I, l_j) + \sum_{v_c \in C_j} B_C(l_j) \right) \tag{4b}$$

$$l_r^* = \text{argmin}_{l_r} \left(m_r(I, l_r) + \sum_{v_c \in C_r} B_C(l_r) \right) \tag{4c}$$

Here, Equation 4a specifies the minimization for the leaf nodes, Equation 4b for non-leaf, non-root nodes, and Equation 4c for the root node.

All possible locations over the discretized parameter space for the root are considered by this recursive relationship. Furthermore, all possible configurations of each part given its parent part location are considered, however, many configurations are pruned early in the process yielding an algorithm quadratic in the spatial discretization. Under certain conditions the running time can be further reduced to linear in the spatial discretization. Since this formulation minimizes the cost of the model match over the entire discretized space of an image, it has a flavor of a top-down technique, however, each individual part is treated quasi-independently in the match function, $m_i(I, l_i)$, giving it characteristics of a bottom-up approach as well.

A specific instance of this method may define the puppet model to include 10 segments, a head, torso, upper and lower arms and upper and lower legs, similar to the model shown in Figure 5 e). The parts, nodes of the graphical model, may be modeled as rectangles with an simple color appearance model. The constraints between parts, the edges in the graphical model, may enforce probabilistically learned distributions on pair-wise range of motion distributions in both rotational and translation dimensions. The kinematic constraints may allow for slight elastic deformations of the parts in the model to account for imperfect model specification and image matching functionality. This method is defined in such a way that many improvements and extensions in terms of part appearance modeling and kinematic constraints are easily incorporated.

3.1.2 Bottom-up Formulation

An alternative to optimizing the model fit over a global, discretized space is to first locate likely image positions of body parts and then combine the parts into likely configurations according to constraints of the human body. This general type of approach is known as bottom-up and has been employed by many researchers for model-based pose estimation [9, 11, 12, 18, 19, 22]. Proponents of bottom-up approaches for pose estimation assert that the search space

can be significantly pruned by rejecting unreasonable features and configurations early on. In practice, this requires appropriate, but simple feature detectors that may exhibit significant noise, as well as grouping primitives that provide clear rules for valid combinations of parts that can robustly encode variations in pose space.

The general approach to bottom-up human pose estimation asserts that a human body configuration is recursively made up of several sub-configurations. Given a set of features, the goal is to recursively combine features into parts, parts into configurations and configurations into meta-configurations until a top-level configuration is produced. Early work caste this as a classification problem, where a classifier for each level in a configuration hierarchy was developed to accept valid and reject invalid configurations. If each level of classifier is fairly good at rejecting invalid configurations, the space of possible poses is significantly reduced as processing moves up the hierarchy. An early work that applied a bottom-up process to human pose estimation is that of Ioffe and Forsyth [11] whose foundations were created in the work on body plans by Forsyth and Fleck [9].

One way that the method of hierarchically building assemblies can be described is as a pyramid of classifiers. A pyramidal classifier for a simplified, five-segment, human model is visualized in Figure 6. For each node, i, in the pyramidal classifier, there is a classifier function, f_i, that assigns values to sub-configurations according to model constraints. A classifier, f_i, accepts a sub-configuration of parts, $L = \langle l_1, l_2, \ldots, l_j \rangle$, if it assigns a value greater than 0. At the lowest level, classifiers may operate on image features, whereas the top level classifier outputs valid whole-body configurations. In this framework, the inverse mapping function for pose estimation from image space is specified by the combination of all hierarchical classifiers.

$$M^{-1}(x_{n\times 1}) \mapsto \theta_{m\times 1} : \mathcal{F} \tag{5a}$$

$$\mathcal{F} = \{f_1, f_2, \ldots, f_k\} \tag{5b}$$

This framework is quite general and leaves the definition of the classifiers and features to be tailored to a specific application. For the application of human pose estimation a common definition for the features is based on groups of parallel line segments, skin tone filters, ridge detection or even face detection. The kinematic constraints of the human body are captured by the classifiers, $\mathcal{F}$, and usually enforce a likelihood over relative scale, orientation and translation between pairs of parts that is learned from a small number of labeled training examples. The actual classifiers may be implemented as simple probability distributions over relative configuration, or more sophisticated probabilistic operators like particle filters and inference over Markov chains.

Building on the general framework of puppet-model human pose detection, there are a number of possible extensions. Whereas the definition of the constraints in both the global and bottom-up models is very general, the problem of human pose detection, especially when the input is somehow limited in domain or pose variation appearance, there can be limiting, but powerful constraints built into this overall framework. For example, the individual part detectors may be much better tuned to the target instance of human pose by making use of face detectors, skin detection and ridge detection [12]. Additionally, in a temporal process where an individual is viewed for many frames, part detections can span the temporal dimension as well, filtering for consistent appearance over time [18]. At the cost of violating the tree structured constraint model, additional constraints may be added between parts that enforce consistency in appearance between left and right limbs, as well as positional dependencies of kinematically independent body parts that arise from a more complex analysis of pose [19]. Finally, as the complexity of the dependencies in the puppet models has increased there has been a tendency to make use of more recent advances in statistical modeling and inference such as Markov chain Monte Carlo [12] and contin-

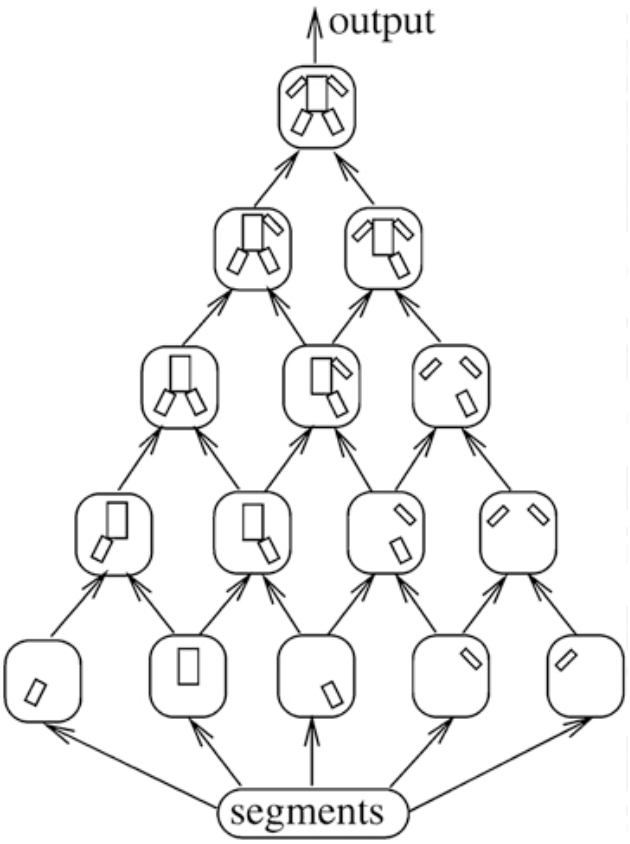

Figure 6. A hierarchical pyramidal classifier for grouping body segments into valid configurations. This illustration was taken from [11].

uous valued particle filters like PAMPAS [22].

3.2. Kinematic Model Constraint Propagation

A categorically different approach to human pose estimation addresses the problem of 3D configural reconstruction of articulated objects from uncalibrated, monocular images using only kinematic constraints and the locations of joints in 2D image space. The method for performing the 3D reconstruction was outlined by Taylor [23]. The reconstruction is valid for any rigidly connected articulated structure under the following assumptions:

1. The image formation can be closely approximated by a scaled orthographic projection model
2. The image coordinates $\langle u, v \rangle$ of the joints between connected segments are given
3. The relative lengths of the segments are known

Under these assumptions a 3D reconstruction up to a global scale ambiguity may be obtained for an articulated object, a specific type of which is the human body.

The first assumption, that the image projection model is scaled orthographic, is valid in many instances where the local depth of the object of interest is small compared to the distance between the object and camera. The scaled orthographic camera model is given in Equation 6.

$$\begin{pmatrix} u \\ v \end{pmatrix} = s \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \tag{6}$$

This system is under-constrained and, as a result, there will be more than one solution for a given set of labeled joints in image coordinates. Nevertheless, the family of solutions can be related under a global scale parameter, s, by carefully parameterizing the model. As a simple example, imagine a single line segment in 3D space whose endpoints in image space are given by $\langle u_A, v_A \rangle, \langle u_B, v_B \rangle$ and whose coordinates in 3D space are given by $\langle X_A, Y_A, Z_A \rangle, \langle X_B, Y_B, Z_B \rangle$. The line segment has a known length of l. Under the scaled orthographic camera model an equation relating the relative depth of the endpoints in 3D space to the image coordinates, length and scale is derived in Equation 7.

$$l^2 = (X_A - X_B)^2 + (Y_A - Y_B)^2 + (Z_A - Z_B)^2 \tag{7a}$$
$$\langle u_A - u_B \rangle = s(X_A - X_B) \tag{7b}$$
$$\langle v_A - v_B \rangle = s(Y_A - Y_B) \tag{7c}$$
$$dZ = (Z_A - Z_B) \tag{7d}$$
$$dZ = \sqrt{l^2 - ((u_A - u_B)^2 + (v_A - v_B)^2) / s^2} \tag{7e}$$

Furthermore, since it does not make sense to have an imaginary displacement value along the z-axis the quantity under the square-root in the derived Equation 7e must be non-negative. This puts an addition constraint, shown in Equation 8, on the scale factor s.

$$s \geq \frac{l}{\sqrt{(u_A - u_B)^2 + (v_A - v_B)^2}} \tag{8}$$

Although in the simplified example of a single segment it was assumed that the overall length was known it is only necessary to know the relative lengths of the segments in an articulated model in general. The overall scale-factor, s, absorbs the need for an absolute length for each segment.

Using the previously derived equations, constraints on s and the three assumptions stated earlier, a straightforward algorithm for recovering 3D coordinates of an articulated model up to scale becomes apparent. Recovering 3D coordinates is achieved through the following steps:

1. Evaluate Equation 8 for each segment, p_i in the model using the labeled image coordinates and relative segment length, l_i, to obtain a lower bound on the scale factor s.
2. Fix some joint in the model as a reference joint at relative depth 0.
3. Starting from the reference joint, compute the value of dZ for all connected segments using Equation 7e and the lower bound scale-factor s.
4. Propagating the previously computed relative depth values, dZ_i, continue in this manner until the relative depth of all points is known.

For each segment there are two possible placements for a computed value of dZ. For an 11 segment model this results in 2048 valid solutions. Given an additional requirement that for each segment the joint that is closest to the camera is labeled as such there is a unique solution. A more automatic way of dealing with this may be to prune the configurations that do not obey joint-angle constraints of the human body.

Notice that the scale factor was fixed at the lower-bound value in the algorithmic steps. This heuristic corresponds to the commonly occurring case where at least one of the segments is approximately perpendicular to the camera vector. In practice, the scale parameter may be swept across a range of values and the resulting model match according to a separate metric could be computed. An illustration of how a reconstruction varies with the scale parameter is shown in Figure 7. Finally, note that for the case of human pose, knowledge of the relative limb lengths is quite reasonable as these may be computed from a corpus of biometric data.

In order for this method to succeed a strong assumption is made; that the image coordinates $\langle u_i, v_i \rangle$ of each joint center are given. Automatically locating these joint centers

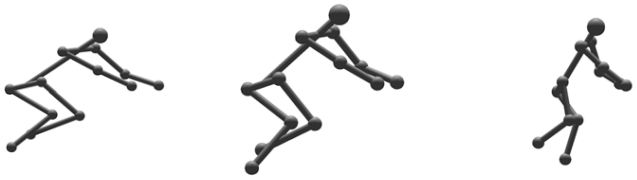

Figure 7. As the scale parameter increases the foreshortening effects are magnified. This image was taken from [23].

in an image remains a difficult problem in the vision community. Nevertheless, some automatic 2D pose estimation systems, notably that of Mori and Malik [14], have used this kinematic method to upgrade a 2D solution to one in 3D. Finally, a related work by Liebowitz and Carlson [13] solves a very similar problem, but does not contain the scale ambiguity. However, two uncalibrated camera views are used, making it out the scope of this work.

4. Model-Free Pose Estimation Techniques

In contrast to human pose estimation techniques that make use of an explicitly defined, often simplified, kinematic model of the human body, model-free techniques make no such assumption about the form of the generating parametric process Θ. Instead, model-free techniques attempt to learn the mapping from image feature space to pose space, $M^{-1}(x) \mapsto \theta$, directly from labeled training pairs of image feature points and their corresponding pose parameters. A general schematic of how model-free pose estimation techniques operate is shown in Figure 8. Although the intermediate feature space, method of learning the inverse mapping, and dimensionality of the estimated parameter space may differ, model-free techniques set the problem up as learning a mapping from an input vector space to an output vector space.

While model-free pose estimation techniques do not require a hand-specified kinematic model, they do require training data with labeled pose parameters. Each training pair consists of an image of a person and the corresponding pose parameters. This labeled training data can be obtained in a number of ways. One possibility is to hand label the body part positions in the training images, which yields 2D image coordinates. Another method that yields 3D pose parameter labels is to simultaneously record motion-capture data using a commercial motion capture system and video using a calibrated video camera. A third option is to use a graphics rendering package that has the capability of animating human models, such as POSER, to generate synthetic frames. In this way, human motion capture data can be used to animate the human model and synthetically generated frames corresponding to each pose parameter setting can be stored.

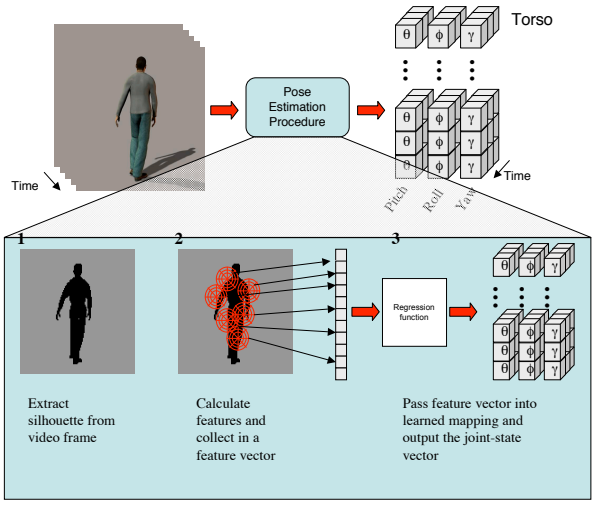

Figure 8. An illustration of the problem formulation and data-flow through a general model-free pose estimation routine. Features are extracted from an image, fed into a learned mapping from feature space to pose space and the resulting pose parameters are output.

Currently, due to computational limits, it appears to be infeasible to work with a dense sampling of the entire possible human pose space in the model-free pose estimation framework. For this reason, researchers in this area make the argument that the range of likely or often-occurring human poses is much smaller than that of all possible poses. Solutions are thus tuned for a specific subset of poses or range of motion. This argument is valid in many cases, especially in targeted applications where a specific type of motion is expected. In this way, a prior on the expected poses, specified by the set of poses covered by the training set, is implicit in the pose estimation procedure. With the ultimate goal of an accurate 3D pose estimation from a single image, in previous work on model-free pose estimation a variety of image features and learning procedures for mapping from feature space to pose space have been proposed. The remainder of this section will describe the proposed types of image features and learning procedures.

4.1. Image Features for Model-Free Pose Estimation

Since there is no kinematic model to constrain the estimation procedure in model-free approaches they need a feature vector representation that robustly captures variations in body shape and pose relevant to the pose estimation task. For the most part, model-free techniques for human pose estimation assume that the pixels corresponding to a person are detected and then define some global measure over the person-pixels. The global nature of the features used in most systems allows for efficient and consistent feature vector extraction once a person is localized. Furthermore, they

provide an input vector with a uniform representation and dimensionality across all input instance, which facilitates straightforward application of many techniques for learning a regression from input (feature) space to output (pose) space. Features based on body contours and silhouettes, as well as edges and edge gradients have been proposed for model-free pose estimation tasks. The following section will describe some possible features in detail and characterize their usefulness with respect to the pose estimation task.

4.1.1 Contour Features

One popular class of image features used by several researchers [2, 5, 14, 20, 24] in model-free human pose estimation are contour features that rely on silhouette extraction. When placed in the context of the ramifications of Equation 1 it is easy to see why contour based features are popular. Assuming that silhouettes can be reliably extracted from images or video frames the nuisance parameters that result from variations in environment, clothing, lighting and other factors disappear. What results then is essentially the problem of estimating body pose from silhouettes, where the silhouette shape is largely determined by the body-pose parameters, Θ, precisely the parameters that are being estimated.

In order to be useful in a model-free pose estimation framework, a silhouette or contour based feature should provide a descriptive summary of the silhouette, be insensitive to small variations and noise and be relatively compact. To this end several types of features computed over binary silhouette images have been proposed. One class of features that have been used by researchers are image moments [5, 20, 24]. These include the scale and translation invariant *Alt* moments, as well as *Hu* moments that have the added property of rotational invariance. The general form for deriving a given *Alt* moment, η_{pq}, is given in Equation 9.

$$\eta_{pq} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{u_i I_i - \bar{u}}{\sigma_u}\right)^p \left(\frac{v_i I_i - \bar{v}}{\sigma_v}\right)^q \qquad (9)$$

Where n is the number of pixels in an image, u_i, v_i are the row and column of pixel i, I_i is the intensity of pixel i and $\bar{u}, \sigma_u$ are the mean and variance. The *Hu* moments result from particular combinations of the *Alt* moments derived to be rotationally invariant.

The numerical values of these moments computed over a silhouette image can then be collected in a feature vector that is used as input to the inverse mapping process. These moments provide global descriptions of shape such as area, inertial moments, principle axes and so forth. *Alt* moments have been used more often than *Hu* moments because, for the domain of human pose estimation, rotational invariance can actually have a negative effect on accuracy since images of people are predominately in the upright orientation. Finally, since image moments are global descriptors of shape, inaccuracies in silhouette extraction such as shadow attachment and other sources of noise can skew the entire descriptor.

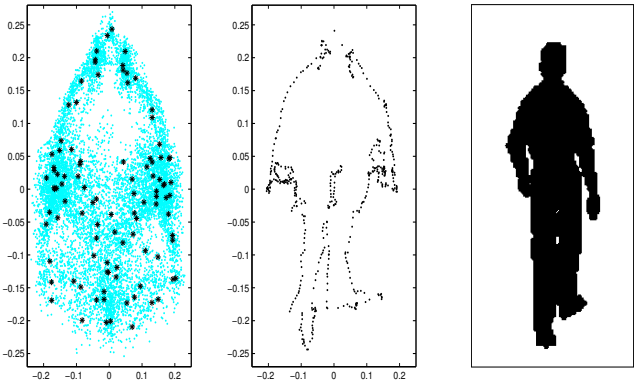

Figure 9. A graphical depiction of the shape variation captured by shape context descriptors computed over a set of human poses. The image on the left is a projection of the feature vectors on the first two principle components. The middle image is the average shape-context feature and the right image is a typical silhouette. This figure was taken from [2]

To counter the effects of local inaccuracies in silhouette extraction, while still providing a reasonable description of overall shape, shape contexts [4] have been proposed for human pose estimation by several researchers [2, 14]. Given a set of feature points detected on the contours of a shape, a shape context for a given point is a histogram of the relative positions of all other points with respect to it. The histogram generally contains equally spaced angular bins and log-sampled distance bins. A graphical depiction of a shape context feature for human pose estimation is shown in Figure 9.

Shape context features for pose estimation have been used directly as in the work by Mori and Malik [14] in a classic retrieval scenario explained in Section 4.2.1. In contrast, Agarwal and Triggs [2] advocate a second level of histogramming on top of shape context extraction. In this second level of histogramming some number of clusters that describe the shape-contexts computed over the training set are learned. The shape context for a silhouette image is then projected onto these centers using a soft-voting technique. This allows the histograms to be compared using the Euclidean distance metric and reduces the dimensionality of the input feature vectors. In either case, shape-contexts provide a reasonable description of silhouette shape while maintaining some invariance to local deformations.

4.1.2 Edge and Gradient Features

There are, however, several drawbacks to contour-based features. Firstly, they discard internal edge information that can be vital in determining the pose of self-occluding body parts, as well as dealing with reflective symmetries in poses. In addition, the assumption of accurate silhouette extraction puts restrictions on the operating environment. Video stream methods that use contour features must assume a fixed camera and a slowly changing background that can be modeled with current background-modeling techniques. In the case of still images with no temporal stream, the use of contour features assumes that the human detection problem has been solved.

For these reasons some researchers have proposed edge and gradient based features for human pose estimation [1, 21]. Shakhnarovich et al. [21] make the observation that internal edges can be a significant cue for pose and propose histograms of edge directions as an input feature. Edges are computed using the Sobel operator and histograms are computed over a sliding window of various sizes. This work still assumes background segmentation, and due to the imperfect nature of edge detection a large number of training examples are necessary to account for variations in the texture of clothing.

Histograms of image gradients, a very similar feature type, are proposed by Agarwal and Triggs [1]. However, an additional step of learning a non-negative matrix factorization (NMF) basis for regularly sampled image patches on training images with no background texture is added. Since an NMF basis provides a sparse, purely additive representation it effectively filters out background clutter from test image that contain background elements. This allows for simultaneous detection and pose estimation to some degree.

4.2. Learning a Mapping from Feature Space to Pose Space

Given a feature space derived from an input image the primary task for model-free pose estimation techniques is to provide a reasonable, accurate, smooth and generalizable mapping from the feature vectors, $x_{n\times 1} \in \mathcal{X}$, to the pose parameter vectors, $\theta_{m\times 1} \in \Theta$. Assuming that there is a set of training pairs available with known pose parameters, this problem can be cast in a number of machine learning paradigms. Current approaches to this problem can be classified as either data-driven or regression learning techniques. Given the special nature of this problem, however, a number of additional heuristics can be employed in conjunction with these methods to improve performance. In the following sections details of several data-driven and regression based techniques for pose estimation are described.

4.2.1 Data-Driven Techniques

Data driven techniques for parameter estimation problems rely on a large set of training examples where each example is a pair, $\langle x, \theta \rangle$, consisting of the computed input image feature vector, x, and the known output pose parameter vector, θ. In the simplest form, data-driven parameter estimation for a novel test example whose image feature vector is x_{test} proceeds by linearly searching the training set to find the training pair whose input image feature representation is most similar to x_{test}. The stored output pose parameters of the best match are then returned as the estimate. Of course, this nearest-neighbor approach assumes that the training set provides a very dense covering of the parameter space. Even for a moderately large pose parameter space this requires a prohibitive number of training examples. For this reason, in practice most data-driven techniques for parameter estimation make a slightly weaker assumption: the training set covers the parameter space densely enough that a reasonably smooth interpolation between examples exists. Nevertheless, this set can still be very large. The key challenges that data-driven parameter estimation techniques face are to define a useful similarity function on the input feature vector space for estimating the output parameter vector space, to accurately interpolate between the matches found in the training set for a given x_{test}, and to efficiently search for the similar examples in the training set.

An example of the nearest-neighbor technique for parameter estimation is presented by Howe [10]. This work uses global features computed on silhouettes as the input representation and 3D pose data gathered from a motion capture system as the parameter space. The parameter estimation procedure proceeds as a simple linear comparison of the input representation of a test example, x_{test}, with all the entries in the training database. The training database is populated so that all entries are sufficiently different in pose space, where different is defined as a sufficiently large difference in the positions of the end points (hands and feet) in parameter space. This approach does not use any regression or weighted neighborhood techniques to refine the estimate, but simply returns the candidate that matches best in feature space.

Another approach to data-driven parameter estimation for human pose estimation is to first find the best candidate in the training dataset and then warp the training candidate to more closely match the test example to account for small differences in the two poses. In addition to the coverage requirement of the training set, for this method to work the input feature space must be amenable to warping in a meaningful way with respect to the pose parameter space. One feature type that displays this quality is the shape context [4] described earlier in Section 4.1.1. Shape contexts have been used in this match-warp paradigm for human pose estimation by Mori and Malik [14, 15].

In order to implement the match-warp procedure for human pose estimation the training examples must be labeled such that (1)the pose parameters are defined with respect to the feature points and (2)the boundary points used for the shape context are assigned to a specific body part. Given this type of labeled information the pose estimation process proceeds as follows:

1. Find the training candidate whose shape-context representation best matches with that of x_{test} by comparing with all training examples
2. Warp the best match to x_{test} in feature space on a part-by-part basis using a 2D kinematic chain representation
3. Recover the 3D pose parameters using the method outlined by Taylor [23]

The second step of warping the best match candidate to x_{test} using a kinematic chain representation is essentially a local search over parameter space that uses the human body's kinematic constraints as a heuristic. Searching in this way can help alleviate the problem of a large training set since fewer examples are required for an accurate estimate. This method can proceed using a global feature representation [14] or by decomposing the search for matching candidates on a part-by-part basis [15] taking advantage of the independent motion of body parts to reduce the size of the training set required to cover pose space.

A third option for tackling this problem is to leverage the advances in locality sensitive hashing techniques for information retrieval operations. A recent work using this representation is that of Shakhnarovich et al. [21] that develops the idea of efficient parameter-sensitive hashing for pose estimation. Since there are ambiguities resulting from reflection and occlusion in feature space with respect to the pose parameters, a locality-sensitive hashing functions that enforces similarity between both the feature and parameter spaces in the inverse mapping, $M^{-1}(x_{n\times 1}) \mapsto \theta_{m\times 1}$, will result in a more accurate pose estimate.

This is achieved by defining a probabilistic set of 1-bit hash functions, $h \in H$, computed over the training set so that for each pair of training examples, $\langle x_1, \theta_1 \rangle$ and $\langle x_2, \theta_2 \rangle$, a hash function $h_{x1} = h_{x2}$ iff $d_\theta(\theta_1, \theta_2) < R$, where d_θ is a distance function defined over parameter space and R is a threshold for similarity. After defining a set of hash functions with this property over the training set, each $h \in H$ is computed for a test example x_{test}. The union of all training examples with at least one similar bit in the hash functions, H, is given as the support-set: a set of estimates that may be close to the query example. Parameter estimation could then proceed in multiple ways such as taking the training example with the most similar hash bits as the estimate (MAP), using the support-set to define the k-nearest neighbors for x_{test} and running gradient descent (Bayes-optimal), or by fitting a linear model to the local support-set and evaluating x_{test} with respect to this model.

4.2.2 Regression Learning Techniques

Whereas data-driven techniques rely on a direct connection with the underlying training data to lookup examples similar to a test example, regression-learning techniques first learn an approximate, smooth regression mapping from input image feature space to parameter space and then use the learned mapping to generalize to new test cases. One of the key challenges for data-driven techniques is to adequately cover a pose space and allow for near-realtime parameter estimation since in many cases a significant portion of the training set is searched on each query. In contrast, regression-learning techniques perform most of the computation offline while learning the mapping. Subsequent queries for new test examples are handled very quickly. The challenge facing regression learning techniques for pose estimation is to provide an accurate regression mapping that captures the intricacies found in the training set while generalizing to a wider selection of test cases. The underlying manifold representing the mapping from input to pose space may be highly non-linear making the regression learning task very challenging. In practice, regression based pose estimation techniques restrict the class of poses to a subset of possible poses or organize the training data into smaller, more homogenous poses to make the regression problem tractable.

An early regression learning technique for human pose estimation is the specialized-mappings architecture [20]. This approach uses the heuristic of first partitioning the training set into similar clusters according to a measurement in pose space. Then, a simpler approximation for the mapping exists for each cluster than the entire space taken as a whole. The general steps for pose estimation using the specialized mappings architecture are as follows:

Assume $x_{n\times 1}$ are feature vectors and θ are joint positions that specify pose.

1. The set θ is partitioned into k subsets using EM-based unsupervised clustering techniques
2. The inverse mapping function $M^{-1}(x) \mapsto \theta$ that maps from input x to output θ is approximated by a non-linear (multi-layer) perceptron for each of the k clusters.
3. Input features from a novel test image x_{test} are presented to each of the perceptron-based mapping functions, $M^{-1}_{I:I=1..k}$ resulting in k pose estimates.
4. A rendering function from pose space to image feature (silhouette) space is used to generate images resulting

from each of the k pose estimates. The best matching projection is chosen as the pose estimate.

It is important to note that dividing the pose space into several clusters necessities an extra, and possibly expensive, step of rendering probable pose estimates and performing distance calculations in image feature space to evaluate the multiple estimate candidates. Furthermore, there are several disjoint steps in this process resulting in many tunable parameters.

The specialized mapping architecture advocates dividing the pose space into a finite, discrete number of subsets to allow for an approximate mapping to be found. As the number of subsets becomes infinite, the continuum of hypotheses approximate a function in pose space. This is the concept used by Tian, Li and Sclaroff [24] in their work that uses Gaussian process latent variable model (GPLVM) to approximate this functional mapping in pose space for human pose estimation. To make the GPLVM learning process tractable, the notion of an active-set is used. An active-set is simply a representative subset of the training data. The GPLVM is set up as an optimization process over radial-basis kernel parameters, and active-set member selection. Once the learning process has converged, the pose parameters for a new example, x_{test}, are estimated by first finding the example in the active-set most like x_{test} in terms of the similarity in feature space. Then, optimization over Equation 10 is performed using the chosen example from the active-set to initialize.

$$\theta^* = \text{argmin}_{x,\theta}(L_{\tilde{\theta}}(x,\theta) + w_i C_{ALT}) \tag{10a}$$

$$C_{ALT} = \|\Phi(\theta) - s_{test}\|^2 \tag{10b}$$

Here, $L_{\tilde{\theta}}(x,\theta)$ contains the learned parameters from the GPLVM process, C_{ALT} is a distance function in silhouette space measuring the closeness of the current parameter estimate to the input silhouette image and w_i specifies an importance weighting on the model fit, $L_{\tilde{\theta}}(x,\theta)$, versus the silhouette match, C_{ALT}. This approach presents a more unified model for regression based human pose estimation.

While the GPLVM approach makes few assumptions about the underlying model, it still requires a post-estimation rendering and matching step for estimation refinement. A recently proposed approach for regression based human pose estimation frames the problem as a very general mapping problem that can be solved by various machine learning techniques [2, 1]. The formulation relies on the assumption that a mapping from feature space, $\mathcal{X}$, to pose space, Θ, can be approximated functionally as a linear combination of basis vectors as in Equation 11.

$$\theta_{m\times 1} = \sum_{k=1}^{p} a_k \phi_k(x_{n\times 1}) + \epsilon \tag{11}$$

Here, a_k are weight vectors for the basis functions, ϕ_k, evaluated over the feature vectors, $\phi_k(x_{n\times 1})$. Note that the weight vectors can be collected in a matrix $A_{m\times p} = (a_1\ a_2\ \ldots a_p)$ and similarly the basis functions can be collected in a function $f(z) = \langle \phi_1(x)\ \phi_2(x)\ \ldots \phi_p(x) \rangle$ yielding the simplified linear combination notation in Equation 12.

$$\theta_m = A \cdot f + b \tag{12}$$

Using this general setup, the process of training a model can be posed as optimizing over Equation 13.

$$A := \text{argmin}_A\{\|AF - X\|^2 + R(A)\} \tag{13}$$

Here the feature vectors, x_i, have been gathered into feature matrix X, the basis functions have been gathered into matrix F, and $R(A)$ is a regularization term on weights A to penalize over-fitting.

Ridge-regression, also known as damped least squares, and relevance vector machines (RVMs) [25] have been used to optimize over this formulation. In the case of RVMs, the optimization process enforces sparsity in the solution that is useful for the pose estimation task. The choice of the basis has been shown to have little effect on the quality of the learned mapping. Implementations using a linear basis, where basis vectors translate to individual features, and a kernel basis, where basis vectors translate to training examples, have shown comparable results. The strength of this approach is in the simplicity of the problem formulation and subsequent estimation for new examples. Estimating the pose for a new example, x_{test}, is a simple matter of evaluating the chosen basis functions, $\phi_i(x_{test})$, over the input feature vector and weighting them according to the learned weights, a_i.

5. Discussion and Current Limitations

Several methods for estimating human pose from a single image using both model-based and model-free approaches have been presented. Although the underlying goal is similar, the various methods employed by each technique span a large range of design decisions, assumptions and potential operating environments. For comparative purposes, information pertaining to the scope, features and limitations of many recent methods for both model-based and model-free techniques is provided in Table 2.

One notably lacking category in Table 2 is a comparison of estimation accuracy. The omission of this statistic hints at one of the major difficulties facing the human pose-estimation research community: a common evaluation dataset and the appropriate metrics have not been defined and widely adopted. With little exception, researchers create their own datasets, and while some are made available, they are generally not used for comparison purposes in the

Category	Paper	Dimensionality	DOF	Limiting Assumptions
Model-based	Felzenszwalb [7]	2D	full-body(24)	Known appearance, no occlusions
	Forsyth [9]	2D	unspecified	known appearance, no occlusions
	Ioffe [11]	2D	full-body	Gen. rectangles, all parts visible
	Lee [12]	3D	full-body(31)	Frontal poses, canonical parts
	Ramanan [18]	2D	full-body(24)	Gen. rectangles, constant appearance
	Ren [19]	2D	full-body	part symmetry and appearance constraint
	Sigal [22]	3D	full-body	part detection solved
	Taylor [23]	3D	full-body(25)	known 2D joint centers
Model-free	Agarwal [1]	3D	upper-body	known body type, frontal pose
	Agarwal [2]	3D	full-body(54)	background sub., small pose space
	Brand [5]	3D	full-body	back-sub, small pose space
	Howe [10]	3D	full-body	back-sub, dense training covering
	Mori [14]	2D$\mapsto$3D	full-body	back-sub
	Mori [15]	2D$\mapsto$3D	full-body	back-sub
	Rosales [20]	2D	full-body	back-sub
	Shakhnarovich [21]	3D	upper-body	frontal poses, known body type
	Tian [24]	2D	full-bosy	back-sub

Table 2. Comparison of several pose estimation techniques.

community. This problem is compacted by the parallel issue of the use of various, incompatible parameterizations of the pose estimate model. This problem is beginning to be addressed by the community in the form of the Workshop on Evaluation of Articulated Human Motion and Pose Estimation the second of which is held in June 2007. In conjunction with this effort a standard dataset, HumanEva, with video and ground truth pose data has been released. This standard dataset is an important step since obtaining ground-truth pose data from commercial MoCap systems still remains a time-consuming and costly venture that may not be available to many researchers.

Another trend that is apparent from Table 2 is that model-free techniques more often supply 3D pose estimates and work with more degrees of freedom than many of the model-based techniques. Model-free techniques also generally run much faster on novel test input images than the model-based techniques. This discrepancy in efficiency on test cases follows directly from the inverse nature of the two classes of approaches. The model-free techniques generally devote significant processing power during the offline learning phase. Once learned, this model-free mapping function can be computed relatively quickly for test examples and usually yields a very small number of potential solutions. In contrast, the model-based techniques generally use some form of graphical model that is either hand specified, or described by simple distributions that can be learned from very few training examples. These models are very powerful and can represent a large, high-dimensional pose space, however, finding a solution for a test example often involves an expensive optimization procedure, or search, over this large space. Looking at this from the opposite perspective, model-based approaches, in theory, can succinctly handle a large range of poses, while current techniques for model-free pose estimation require exponentially large training datasets to adequately cover the same space.

The overall challenge for model-based human pose estimation techniques is in dealing efficiently and effectively with a high-dimension search space with multiple minima. This manifests itself in two obvious ways; one is the need for more discriminative and accurate body-part detectors and the other relates to probabilistic inferences methods needed to search the large parameter space and scale these methods to 3D pose estimation.

The prevailing wisdom in the model-based pose estimation community has been that low-level body-part detectors can be very noisy since the constraints defined by the body-model will significantly prune the false positives. This is true to some extent, but current part detectors are perhaps too noisy and ill-behaved over the image space, resulting in many probable, but incorrect solutions to the pose estimation problem. Additionally, if the search space is not sufficiently pruned in the body-part detection step, the optimization over the probabilistic kinematic model is very expensive. Several recent works have attempted to address this problem by introducing more specific and discriminative part detectors such as specialized face detectors, shoulder contour detectors etc [12], however, for many body parts this is a very difficult task since out of context they can be very hard to distinguish as in Figure 10. Orthogonally, recent techniques for continuous-valued probabilistic inference such as PAMPAS have enabled the large state space to be handled more effectively [22]. Nevertheless, there is still significant room for improvement in terms of both body-

Figure 10. Taken out of context body parts can be very hard to distinguish, even for people. This image was taken from [16]

part detection and inference mechanisms.

In contrast, the main challenge facing model-free human pose estimation techniques is in representing a large portion of the allowable human pose space. While many model-free techniques provide estimates in 3D, they often limit the range of allowable input poses by either focusing only on the upper body [1, 21] or limiting the poses to those of frames of a motion sequence, such as walking, that exhibits only a small amount of variation [2, 20]. One reason for this limitation results from the global nature of the image features used to learn the mapping. With a global representation, independent motions of body parts lead to an exponential blow-up in the number of examples needed for adequate coverage of pose space.

A related challenge for model-free methods is in handling multiple body types. The shape of the body can be viewed as an additional parameter space affecting pose and it is not handled by most techniques. Finally, with the exception of a recent work by Agarwal and Triggs [1], the image features used are computed over contours, or images without background clutter. In order to be applicable to general scenarios this clean background assumption needs to be relaxed.

One possible improvement that may address the limitation on poses handled by model-free systems is to introduce factored pose spaces as a processing step. This could possibly be done in a number of ways. One way to factor the pose space may be to use a body-localized decomposition of the features so that only features from the right arm are used to learn the mapping to right arm parameters and so forth. This bears some resemblance to the bottom-up methodology found in the model-based community. It may throw away information about dependencies found in motion sequences, but could potentially handle a much larger pose-space than globally calculated features.

A second way to address this problem may be to partition the training poses into similar clusters prior to the learning phase. This approach has been explored to some degree by, [20, 24], but there is room for improvement in terms of automatically learning these clusters for a wide number of poses and in efficiently choosing the appropriate cluster for a novel test example. Finally, improvements in the robustness of image features with respect to texture variation and background clutter may be gained by using new feature types that account for these variations. The histogram of gradients (HOG) features have proven useful for the human detection task and there is limited evidence that they will be useful for pose estimation in cluttered environments as well [1].

While there is much room for improvement, current model-free human pose estimation techniques seem to offer more promise for a monocular pose estimation, especially in reasonably restricted settings. This may come as a surprise as the prevailing knowledge among many researchers in this area has been that kinematic models are necessary for human pose estimation. Improvements in machine learning techniques for very high dimensional problems have had a large impact on changing this view. Additionally, the increasing storage and memory capacities of computers enable model-free approaches to handle large training sets. Finally, most model-free methods rely on labeled training data from commercial MoCap systems, which is still sparse but has become increasingly accessible over the past decade. Nevertheless, model-based methods are certainly useful for computer vision based motion capture in general. A unified system that uses a model-free approach for (re)initialization and a model-based approach for frame-to-frame tracking may prove superior.

References

[1] A. Agarwal and B. Triggs. A local basis representation for estimating human pose from cluttered images. In *Asian Conference on Computer Vision*, 2006. 11, 13, 14, 15

[2] A. Agarwal and B. Triggs. Recovering 3d human pose from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1):44–58, 2006. 10, 13, 14, 15

[3] V. Athitsos and S. Sclaroff. Database indexing methods for 3d hand pose estimation. In *Proc. of the Gesture Workshop*, 2003. 2

[4] S. Belongie, J. Malik, and J. Puzicha. Shape context: A new descriptor for shape matching and object recognition. In *Advances in Neural Information Processing Systems 14*, 2001. 10, 11

[5] M. Brand. Shadow puppetry. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1237–1244, 1999. 10, 14

[6] A. Erol, G. Bebis, M. Nicolescu, R. Boyle, and X. Twombly. Vision-based hand pose estimation: A review. In *Computer Vision and Image Understanding*, 2007. 2

[7] P. Felzenszwalb and D. Huttenlocher. Efficient matching of pictorial structures. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume 2, pages 2066–2073, 2000. 5, 6, 14

[8] M. A. Fischler and R. A. Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computers*, 22(1):67–92, 1973. 6

[9] D. Forsyth and M. Fleck. Body plans. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 678–683, 1997. 5, 6, 7, 14

[10] N. Howe, M. Leventon, and W. Freeman. Bayesian reconstruction of 3D human motion from single camera video. In S. Solla, T. K. Leen, and K.-R. Müller, editors, *Advances in Neural Information Processing Systems 13*, pages 820–826. MIT Press, 2000. 11, 14

[11] S. Ioffe and D. Forsyth. Probabilistic methods for finding people. *International Journal of Computer Vision*, 43(1):45–68, 2001. 5, 6, 7, 14

[12] M. W. Lee and I. Cohen. A model-based approach for estimating human 3d poses in static images. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 905–916, 2006. 5, 6, 7, 14

[13] D. Liebowitz and S. Carlson. Uncalibrated motion capture exploiting articulated structure constraints. In *International Journal of Computer Vision*, pages 171–187, 2003. 9

[14] G. Mori and J. Malik. Estimating human body configurations using shape context matching. In A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, editors, *Proceedings of the Seventh European Conference on Computer Vision*, LNCS 2352, pages 666–680. Springer Verlag, 2002. 9, 10, 11, 12, 14

[15] G. Mori and J. Malik. Recovering 3d human body configurations using shape contexts. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006. 11, 12, 14

[16] G. Mori, X. Ren, A. Efros, and J. Malik. Recovering human body configurations: Combining segmentation and recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 326–333, 2004. 15

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

[18] D. Ramanan, D. Forsyth, and A. Zisserman. Tracking people by learning their appearance. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007. 5, 6, 7, 14

[19] X. Ren, A. Berg, and J. Malik. Recovering human body configrations using pairwise constraints between parts. In *Proceedings of the Eleventh IEEE International Conference on Computer Vision*, 2005. 5, 6, 7, 14

[20] R. Rosales and S. Sclaroff. Learning body pose via specialized maps. In *Advances in Neural Information Processing Systems 14*, pages 1263–1270, 2001. 10, 12, 14, 15

[21] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter-sensitive hashing. In *Proceedings of the Ninth IEEE International Conference on Computer Vision*, volume 2, pages 750–757, 2003. 11, 12, 14, 15

[22] L. Sigal, M. Isard, B. Sigelman, and M. Black. Attractive people: Assembling loose-limbed models using non-parametric belief propagation. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*. MIT Press, 2004. 5, 6, 8, 14

[23] C. J. Taylor. Reconstruction of articulated objects from point correspondence using a single uncalibrated image. 80(3):349–363, 2000. 5, 8, 9, 12, 14

[24] T.-P. Tian, R. Li, and S. Sclaroff. Articulated pose estimation in a learned smooth space of feasible solutions. In *Prc. CVPR Learning Workshop*, 2005. 10, 13, 14, 15

[25] M. Tipping. Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211–244, 2004. 13