3D Animal Reconstruction with Deformable Template Models

Benjamin Biggs

Supervisor: Dr. Andrew Fitzgibbon
Prof. Roberto Cipolla

Department of Engineering
University of Cambridge

This dissertation is submitted for the degree of
Doctor of Philosophy

Jesus College

January 2021

"Our perfect companions never have fewer than four feet."
Colette (1873 – 1954)

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Benjamin Biggs
January 2021

Acknowledgements

This work would not have been possible without the dedicated support from my PhD supervisors, Andrew Fitzgibbon and Roberto Cipolla. I would also like to thank the dedicated staff at GlaxoSmithKline, in particular the Pharma Supply Chain Tech Digital Innovation group, and particularly my line manager Patrick Hyett and project sponsor Julie Huxley-Jones.

The authors would like to thank Ignas Budvytis and James Charles for technical discussions, Peter Grandi and Raf Czlonka for their impassioned IT support, the Biggs' family for the excellent title pun and Philippa Liggins for proof reading.

The authors would like to thank Richard Turner for useful technical discussions relating to normalizing flows, and Philippa Liggins, Thomas Roddick and Nicholas Biggs for proof reading. This work was entirely funded by Facebook AI Research.

Abstract

TODO. We present a system to recover the 3D shape and motion of a wide variety of quadrupeds from video. The system comprises a machine learning front-end which predicts candidate 2D joint positions, a discrete optimization which finds kinematically plausible joint correspondences, and an energy minimization stage which fits a detailed 3D model to the image. In order to overcome the limited availability of motion capture training data from animals, and the difficulty of generating realistic synthetic training images, the system is designed to work on silhouette data. The joint candidate predictor is trained on synthetically generated silhouette images, and at test time, deep learning methods or standard video segmentation tools are used to extract silhouettes from real data. The system is tested on animal videos from several species, and shows accurate reconstructions of 3D shape and pose.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

1.1 Motivation

Animal welfare is an important concern for business and society, with an estimated 70 billion animals currently living under human care. Monitoring and assessment of animal health can be assisted by obtaining accurate measurements of an individual's shape, volume and movement. These measurements should be taken without interfering with the animal's normal activity, and are needed around the clock, under a variety of lighting and weather conditions, perhaps at long range (e.g. in farm fields or wildlife parks). Therefore a very wide range of cameras and imaging modalities must be handled. For small animals in captivity, a depth camera might be possible, but techniques which can operate solely from intensity data will have a much wider range of applicability

1.2 Approach

We address this problem using techniques from the recent human body and hand tracking literature, combining machine learning and 3D model fitting. A discriminative front-end uses a deep hourglass network to identify candidate 2D joint positions. These joint positions are then linked into coherent skeletons by solving an optimal joint assignment problem, and the resulting skeletons create an initial estimate for a generative model-fitting back-end to yield detailed shape and pose for each frame of the video. Although superficially similar to human tracking, animal tracking (AT) has some interesting differences that make it worthy of study:

Variability. In one sense, AT is simpler than human tracking as animals generally do not wear clothing. However, variations in surface texture are still considerable between individuals, and the variety of shape across and within species is considerably greater. If

tracking is specialized to a particular species, then shape variation is smaller, but training data is even harder to obtain. Training data. For human tracking, hand labelled sequences of 2D segmentations and joint positions have been collected from a wide variety of sources [3–5]. Of these two classes of labelling, animal segmentation data is available in datasets such as MSCOCO [4], PASCAL VOC [6] and DAVIS [7]. However this data is considerably sparser than human data, and must be "shared" across species, meaning the number of examples for a given animal shape class is considerably fewer than is available for an equivalent variation in human shape. While segmentation data can be supplied by non-specialist human labellers, it is more difficult to obtain joint position data. Some joints are easy to label, such as "tip of snout", but others such as the analogue of "right elbow" require training of the operator to correctly identify across species. Of more concern however, is 3D skeleton data. For humans, motion capture (mocap) can be used to obtain long sequences of skeleton parameters (joint positions and angles) from a wide variety of motions and activities. For animal tracking, this is considerably harder: animals behave differently on treadmills than in their quotidian environments, and although some animals such as horses Creatures great and SMAL 3 and dogs have been coaxed into motion capture studios [8], it remains impractical to consider mocap for a family of tigers at play. These concerns are of course alleviated if we have access to synthetic training data. Here, humans and animals share an advantage in the availability of parameterized 3D models of shape and pose. The recent publication of the Skinned Multi-Animal Linear (SMAL) model [9] can generate a wide range of quadruped species, although without surface texture maps. However, as with humans, it remains difficult to generate RGB images which are sufficiently realistic to train modern machine learning models. In the case of humans, this has been overcome by generating depth maps, but this then requires a depth camera at test time [10]. The alternative, used in this work, is to generate 2D silhouette images so that machine learning will predict joint heatmaps from silhouettes only

1.3 Contributions

In summary, the contributions of this thesis are as follows:

1. We demonstrate a robust framework for 3D animal reconstruction using deformable template models

1.4 Co-Authored Papers

Extracts from this thesis appear in the following co-authored publications and preprints. Chapter 2 contains work from:

1. Biggs Benjamin, Roddick Thomas

1.5 Thesis Structure

The following four thesis chapters discuss important

Chapter 2

Background

2.1 Introduction

In this section, I will introduce the components required for the rest of this thesis.

2.2 Representing 3D Objects

2.2.1 Simple Structures

Talk about meshes, neural radiance fields etc.

2.2.2 Articulated Objects

2.3 Camera Geometry

2.3.1 Pinhole Camera Model

2.3.2 Rendering

2.4 Methods for Learning

2.4.1 Energy Minimization

2.4.2 Deep Learning

Chapter 3

Learning from Synthetic Data

3.1 Introduction

And now I begin my third chapter here ...

1. We want precise shape reconstructions in real-time
2. Refer back to SMAL-ST, requires 3D synthetic training data from video
3. A whole chapter on 'training in-the-loop' methods
4. Talk about expectation maximization, SMBLD, EM-in-the-loop
5. Optional: Talk about how this can be extended to a multi-view GSK setup, similar to MeshRCNN

3.2 Abstract

We present a system to recover the 3D shape and motion of a wide variety of quadrupeds from video. The system comprises a machine learning front-end which predicts candidate 2D joint positions, a discrete optimization which finds kinematically plausible joint correspondences, and an energy minimization stage which fits a detailed 3D model to the image. In order to overcome the limited availability of motion capture training data from animals, and the difficulty of generating realistic synthetic training images, the system is designed to work on silhouette data. The joint candidate predictor is trained on synthetically generated silhouette images, and at test time, deep learning methods or standard video segmentation tools are used to extract silhouettes from real data. The system is tested on animal videos from several species, and shows accurate reconstructions of 3D shape and pose.

3.3 Introduction

Animal welfare is an important concern for business and society, with an estimated 70 billion animals currently living under human care [29]. Monitoring and assessment of animal health can be assisted by obtaining accurate measurements of an individual's shape, volume and movement. These measurements should be taken without interfering with the animal's normal activity, and are needed around the clock, under a variety of lighting and weather conditions, perhaps at long range (e.g. in farm fields or wildlife parks). Therefore a very wide range of cameras and imaging modalities must be handled. For small animals in captivity, a depth camera might be possible, but techniques which can operate solely from intensity data will have a much wider range of applicability.

We address this problem using techniques from the recent human body and hand tracking literature, combining machine learning and 3D model fitting. A discriminative front-end uses a deep hourglass network to identify candidate 2D joint positions. These joint positions are then linked into coherent skeletons by solving an optimal joint assignment problem, and the resulting skeletons create an initial estimate for a generative model-fitting back-end to yield detailed shape and pose for each frame of the video.

Although superficially similar to human tracking, animal tracking (AT) has some interesting differences that make it worthy of study:

Variability.

In one sense, AT is simpler than human tracking as animals generally do not wear clothing. However, variations in surface texture are still considerable between individuals, and the variety of shape across and within species is considerably greater. If tracking is specialized to a particular species, then shape variation is smaller, but training data is even harder to obtain.

Training data.

For human tracking, hand labelled sequences of 2D segmentations and joint positions have been collected from a wide variety of sources [6**?** , 37]. Of these two classes of labelling, animal *segmentation* data is available in datasets such as MSCOCO [**?**], PASCAL VOC [26] and DAVIS [68]. However this data is considerably sparser than human data, and must be "shared" across species, meaning the number of examples for a given animal shape class is considerably fewer than is available for an equivalent variation in human shape. While segmentation data can be supplied by non-specialist human labellers, it is more difficult to obtain *joint position* data. Some joints are easy to label, such as "tip of snout", but others

such as the analogue of "right elbow" require training of the operator to correctly identify across species.

Of more concern however, is 3D skeleton data. For humans, motion capture (mocap) can be used to obtain long sequences of skeleton parameters (joint positions and angles) from a wide variety of motions and activities. For animal tracking, this is considerably harder: animals behave differently on treadmills than in their quotidian environments, and although some animals such as horses and dogs have been coaxed into motion capture studios [91], it remains impractical to consider mocap for a family of tigers at play.

These concerns are of course alleviated if we have access to synthetic training data. Here, humans and animals share an advantage in the availability of parameterized 3D models of shape and pose. The recent publication of the Skinned Multi-Animal Linear (SMAL) model [100] can generate a wide range of quadruped species, although without surface texture maps. However, as with humans, it remains difficult to generate RGB images which are sufficiently realistic to train modern machine learning models. In the case of humans, this has been overcome by generating depth maps, but this then requires a depth camera at test time [76]. The alternative, used in this work, is to generate 2D silhouette images so that machine learning will predict joint heatmaps from silhouettes only.

Taking into account the above constraints, this work applies a novel strategy to animal tracking, which assumes a machine-learning approach to extraction of animal silhouettes from video, and then fits a parameterized 3D model to silhouette sequences. We make the following contributions:

- A machine-learned mapping from silhouette data of a large class of quadrupeds to generic 2D joint positions.
- A novel optimal joint assigment (OJA) algorithm extending the bipartite matching of Cao *et al.* [15] in two ways, one which can be cast as a quadratic program (QP), and an extension optimized using a genetic algorithm (GA).
- A procedure for optimization of a 3D deformable model to fit 2D silhouette data and 2D joint positions, while encouraging temporally coherent outputs.
- We introduce a new benchmark animal dataset of joint annotations (BADJA) which contains sparse keypoint labels and silhouette segmentations for eleven animal video sequences. Previous work in 3D animal reconstruction has relied on bespoke hand-clicked keypoints [100, 99] and little quantitative evaluation of performance could be given. The sequences exhibit a range of animals, are selected to capture a variety of animal movement and include some challenging visual scenarios such as occlusion and motion blur.

The system is outlined in Fig. **??**. The remainder of the paper describes related literature before a detailed description of system components. Joint accuracy results at multiple stages of the pipeline are reported on the new BADJA dataset, which contains ground truths for real animal subjects. We also conduct experiments on synthetic animal videos to produce joint accuracy statistics and full 3D mesh comparisons. A qualitative comparison is given to recent work [100] on the related single-frame 3D shape and pose recovery problem. The paper concludes with an assessment of strengths and limitations of the work.

3.4 Related work

3D animal tracking is relatively new to the computer vision literature, but animal breed identification is a well studied problem [22]. Video tracking benchmarks often use animal sequences [51, 45], although the tracking output is typically limited to 2D affine transformations rather than the detailed 3D mesh that we propose. Although we believe our work is the first to demonstrate dense 3D tracking of animals in video without the need for user-provided keypoints, we do build on related work across computer vision:

Morphable shape models.

Cashman and Fitzgibbon [16] obtained one of the first 3D morphable animal models, but their work was limited to small classes of objects (e.g. dolphins, pigeons), and did not incorporate a skeleton. Their work also showed the use of the 2D silhouette for fitting, which is key to our method. Reinert *et al.* [71] meanwhile construct 3D meshes by fitting generalized cylinders to hand-drawn skeletons. Combined skeletal and morphable models were used by Khamis *et al.* [44] for modelling the human hand, and Loper *et al.* [55] in the SMPL model which has been extensively used for human tracking.

The SMPL model was extended to animals by Zuffi *et al.* [100], where the lack of motion capture data for animal subjects is cleverly overcome by building the model from 41 3D scans of toy figurines from five quadruped families in arbitrary poses. Their paper demonstrates single-frame fits of their model to real-world animal data, showing that despite the model being built from "artists' impressions" it remains an accurate model of real animals. This is borne out further by our work. Their paper did however depend on per-frame human annotated keypoint labels, which would be costly and challenging to obtain for large video sequences. This work was recently extended [99] with a refinement step that optimizes over model vertex positions. This can be considered independent to the initial SMAL model fit and would be trivial to add to our method.

Shape from silhouette.

Silhouette images have been shown to contain sufficient shape information to enable their use in many 3D recovery pipelines. Chen *et al.* [20] demonstrate single-view shape reconstruction from such input for general object classes, by building a shape space model from 3D samples. More related to our work, Favreau *et al.* [28] apply PCA to silhouette images to extract animal gaits from video sequences. The task of predicting silhouette images from 2D input has been effectively used as a proxy for regressing 3D model parameters for humans [**? ?**] and other 3D objects [90].

Joint position prediction.

There is an extensive body of prior work related to joint position prediction for human subjects. Earlier work used graphical approaches such as pictorial structure models [7, 69, 37], which have since been replaced with deep learning-based methods [15, 13]. Few works predict animal joint positions directly owing to the lack of annotated data, although Mathis *et al.* [59] demonstrate the effectiveness of human pose estimation architectures for restricted animal domains. Our method instead trains on silhouette input, allowing the use of synthetic training imagery. The related task of animal part segmentation [89, 88] has seen some progress due to general object part datasets [19, 97].

3.4.1 Preliminaries

We are given a deformable 3D model such as SMAL [100] which parametrizes a 3D mesh as a function of *pose* parameters $\theta \in \mathbb{R}^P$ (e.g. joint angles) and *shape* parameters $\beta \in \mathbb{R}^B$. In detail, a 3D mesh is an array of vertices $v \in \mathbb{R}^{3\times V}$ (the vertices are columns of a $3 \times V$ matrix) and a set of triangles represented as integer triples (i, j, k), which are indices into the vertex array. A deformable model such as SMPL or SMAL may be viewed as supplying a set of triangles, and a function

$$v(\theta, \beta) : \mathbb{R}^P \times \mathbb{R}^B \mapsto \mathbb{R}^{3\times V} \tag{3.1}$$

which generates the 3D model for a given pose and shape. The mesh topology (i.e. the triangle vertex indices) is provided by the deformable model, and is the same for all shapes and poses we consider, so in the sequel we shall consider a mesh to be defined only by the 3D positions of its vertices.

In any given image, the model's 3D *position* (i.e. translation and orientation) is also unknown, and will be represented by a parametrization ϕ which may be for example translation as a 3-vector and rotation as a unit quaternion. Application of such a transformation to a $3 \times V$ matrix will be denoted by $*$, so that

$$\phi * v(\theta, \beta) \tag{3.2}$$

represents a 3D model of given pose and shape transformed to its 3D position.

We will also have occasion to talk about model *joints*. These appear naturally in models with an explicit skeleton, but more generally they can be defined as some function mapping from the model parameters to an array of 3D points analogous to the vertex transformation above. We consider the joints to be defined by post-multiplying by a $V \times J$ matrix $\mathtt{K}$. The j^{th}

column of K defines the 3D position of joint j as a linear combination of the vertices (this is quite general, as v may include vertices not mentioned in the triangulation). A general camera model is described by a function $\pi : \mathbb{R}^3 \mapsto \mathbb{R}^2$. This function incorporates details of the camera intrinsics such as focal length, which are assumed known. Thus

$$\kappa(\phi, \theta, \beta) := \pi(\phi * v(\theta, \beta)\mathtt{K}) \tag{3.3}$$

is the $2 \times J$ matrix whose columns are 2D joint locations corresponding to a 3D model specified by (position, pose, shape) parameters (ϕ, θ, β).

The model is also assumed to be supplied with a rendering function R which takes a vertex array in camera coordinates, and generates a 2D binary image of the model silhouette. That is,

$$R\big(\phi * v(\theta, \beta)\big) \in \mathbb{B}^{W \times H} \tag{3.4}$$

for an image resolution of $W \times H$. We use the differentiable renderer of Loper *et al.* [56] to allow derivatives to be propagated through R.

3.5 Method

The test-time problem to be solved is to take a sequence of input images $\mathscr{I} = [I_t]_{t=1}^{T}$ which are segmented to the silhouette of a single animal (i.e. a video with multiple animals is segmented multiple times), producing a sequence of binary silhouette images $\mathscr{S} = [S_t]_{t=1}^{T}$.

The computational task is to output for each image the shape, pose, and position parameters describing the animal's motion.

As outlined above, the method has three parts. (1.) The discriminative front-end extracts silhouettes from video, and then uses the silhouettes to predict 2D joint positions, with multiple candidates per joint. (2.) Optimal joint assignment (OJA) corrects confused or missing skeletal predictions by finding an optimal assignment of joints from a set of network-predicted proposals. Finally, (3.) a generative deformable 3D model is fitted to the silhouettes and joint candidates as an energy minimization process.

3.5.1 Prediction of 2D joint locations using multimodal heatmaps

The goal of the first stage is to take, for each video frame, a $W \times H$ binary image representing the segmented animal, and to output a $W \times H \times J$ tensor of heatmaps. The network architecture is standard: a stacked hourglass network [62] using synthetically generated training data, but the training procedure is augmented using "multi-modal" heatmaps.

For standard unimodal heatmaps, training data comprises (S, κ) pairs, that is pairs of binary silhouette images, and the corresponding 2D joint locations as a $2 \times J$ matrix. To generate each image, a random shape vector β, pose parameters θ and camera position ϕ are drawn, and used to render a silhouette $R\big(\phi * v(\theta, \beta)\big)$ and 2D joint locations $\kappa(\phi, \theta, \beta)$, which are encoded into a $W \times H \times J$ tensor of heatmaps, blurring with a Gaussian kernel of radius σ.

The random camera positions are generated as follows: the orientation of the camera relative to the animal is uniform in the range $[0, 2\pi]$, the distance from the animal is uniform in the range 1 to 20 meters and the camera height is in the range $[0, \frac{\pi}{2}]$. This smaller range is chosen to restrict unusual camera elevation. Finally, the camera "look" vector is towards a point uniformly in a 1m cube around the animal's center, and the "up" vector is Gaussian around the model Y axis.

This training process generalizes well from synthetic to real images due to the use of the silhouette, but the lack of interior contours in silhouette input data often results in confusion between joint "aliases": left and right or front and back legs. When these predictions are wrong and of high confidence, little probability mass is assigned to the area around the correct leg, meaning no available proposal is present after non-maximal suppression.

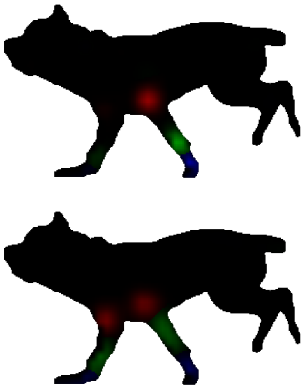


Fig. 3.1 Example predictions from a network trained on unimodal (top) and multi-modal (bottom) ground-truth for front-left leg joints.

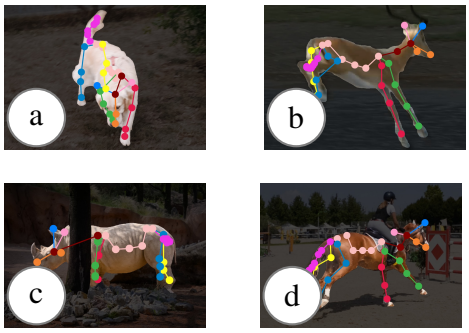


Fig. 3.2 Example outputs from the joint prediction network, with maximum likelihood predictions linked into skeleton.

We overcome this by explicitly training the network to assign some probability mass to the "aliased" joints. For each joint, we define a list of potential aliases as weights $\lambda_{j,j'}$ and linearly blend the unimodal heatmaps G to give the final training heatmap H:

$$H_j(p) = \sum_{j'} \lambda_{j,j'} G(p; \kappa_{j'}, \sigma) \tag{3.5}$$

For non-aliased joints j (all but the legs), we simply set $\lambda_{j,j} = 1$ and $\lambda_{j,j'} = 0$, yielding the unimodal maps, and for legs, we use 0.75 for the joint, and 0.25 for the alias. We found this ratio sufficient to ensure opposite legs have enough probability mass to pass through a modest non-maximal suppression threshold without overly biasing the skeleton with maximal predicted confidence. An example of a heatmap predicted by a network trained on multimodal training samples is illustrated in Fig. 3.1.

3.5.2 Optimal joint assignment (OJA)

Since heatmaps generated by the joint predictor are multi-modal, the non-maximum suppression procedure yields multiple possible locations for each joint. We represent the set of joint proposals $X = \{x_{jp}\}$, where x_{jp} indicates the 2D position of proposal $p \in \{1, ..., N_j\}$ associated with joint $j \in J$. Before applying the optimizer, we must select a subset of proposals $X^* \subseteq X$ which form a complete skeleton, i.e. precisely one proposal is selected for every joint. In this section we consider how to choose the optimal subset by formulating the problem as an extended optimal assignment problem.

In order to select a complete skeleton proposal from the set of joint proposals $\{x_{jp}\}$, we introduce a binary indicator vector $\bar{a}_j = \{a_{jp}\} \in \{0,1\}^{N_j+1}$, where $a_{jp} = 1$ indicates that the p^{th} proposal for joint j is a correct assignment, and the $p = N_j + 1$ position corresponds to a *null proposal*, indicating that joint j has no match in this image. The null proposals are handled as described in each of the energy terms below. Let A be the jagged array $[\bar{a}_j]_{j=1}^{J}$ containing all assignment variables (for the current frame), and let $X^* = X(A)$ denote the subset of points selected by the binary array A. Optimal assignment minimizes the function

$$L(A) = L_{\text{prior}}(A) + L_{\text{conf}}(A) + L_{\text{temp}}(A) + L_{\text{cov-sil}}(A) + L_{\text{cov-bone}}(A) \tag{3.6}$$

which balances agreement of the joint configuration with a learned *prior*, the network-supplied *confidences*, *temporal* coherence, and *coverage* terms which encourage the model to correctly project over the silhouette. Without the coverage terms, this can be optimized as a quadratic program, but we obtain better results by using the coverage terms, and using a genetic algorithm. In addition, the parameters A must satisfy the J constraints $\sum_{p=1}^{N_j+1} a_{jp} = 1$, that exactly one joint proposal (or the null proposal) must be selected for each joint.

L_{prior}:

We begin by defining the prior probability of a particular skeletal configuration as a multivariate Gaussian distribution over selected joint positions.

The mean $\mu \in \mathbb{R}^{2J}$ and covariance $\Sigma \in \mathbb{R}^{2J \times 2J}$ terms are obtained from the training examples generated as above. The objective of OJA is to select a configuration X^* which maximizes the prior, which is equivalent to minimizing the Mahalanobis distance $(x^* - \mu)^T \Sigma^{-1} (x^* - \mu)$, which is given by the summation

$$L_{\text{prior}}(A) = \sum_{j}^{J} \sum_{p}^{N_j} \sum_{k}^{J} \sum_{q}^{N_k} a_{jp} a_{kq} (x_{jp} - \mu_j) \Sigma_{jk}^{-1} (x_{kq} - \mu_k) \tag{3.7}$$

This is a quadratic function of A, so $L_{\text{prior}}(A) = \text{vec}(A)^\top Q \text{vec}(A)$ for a fixed matrix Q, and can be formulated as a quadratic program (QP). Null proposals are simply excluded from the sum, equivalent to marginalizing over their position.

L_{conf}:

The next energy term comes from the output of the joint prediction network, which provides a confidence score y_{jp} associated with each joint proposal x_{jp}. Then $L_{\text{conf}}(A) = \sum_j \sum_p -\lambda \log(y_{jp}) a_{jp}$ is a linear function of A, and λ_{conf} is a tunable parameter to control

the relative contribution of the network confidences compared with that of the skeleton prior. Null proposals pay a fixed cost λ_{null}, effectively acting as a threshold whereby the null proposal will be selected if no other proposal is of sufficient likelihood.

L_{temp}:

A common failure case of the joint prediction network is in situations where a joint position is highly ambiguous, for example between the left and right legs. In such cases, the algorithm will commonly alternate between two equally likely predictions. This leads to large displacements in joint positions between consecutive frames which are difficult for the later model fitting stage to recover from. This can be addressed by introducing a temporal term into the OJA. We impose a prior on the distance moved by each joint between frame t_0 and t_1, which is given by a normal distribution with zero mean and variance $\sigma^2 = e^{\tau|t_1-t_0-1|}$. The parameter τ controls the strength of the interaction between distant frames. This results in an additional quadratic term in our objective function, which has the form $L_{temp} = a^\top T^{(t_0,t_1)} a$ for matrix $T^{(t_0,t_1)}$ given by

$$\left[T^{(t_0,t_1)}\right]_{jp,kq} = \begin{cases} e^{-\alpha|t_1-t_0-1|}||x_{jp}^{(t_0)} - x_{kq}^{(t_1)}||^2 & \text{if } j = k \\ 0 & \text{otherwise} \end{cases} \tag{3.8}$$

QP solution.

Thus far, all terms in $L(A)$ are quadratic or linear. To optimize over an entire sequence of frames, we construct the block diagonal matrix $\hat{Q}$ whose diagonal elements are the prior matrices $Q^{(t)}$ and the block symmetric matrix $\hat{T}$ whose off-diagonal elements are the temporal matrices $T^{(t_0,t_1)}$. The solution vector for the sequence $\hat{A}$ is constructed by stacking the corresponding vectors for each frame. The quadratic program is specified using the open source CVXPY library [23] and solved using the "*Suggest-and-Improve*" framework proposed by Park and Boyd [65]. It is initialized by choosing the proposal with the highest confidence for each joint. Appropriate values for the free parameters $\lambda_{\text{conf,temp}}$ and α were chosen empirically via grid search.

$L_{\text{cov-\{sil,bone\}}}$:

The above quadratic formulation is sufficient to correct many errors in the raw output (which we later demonstrate in the experimental section), but suffers from an 'overcounting' problem, in which leg joint predictions both cover the same silhouette leg region, leaving another leg empty. We therefore extend the definition of $L(A)$ to include two additional terms.

Fig. 3.3 Silhouette coverage loss. The error (shown in red) is the the distance between the median axis transform (right) and the nearest point on an approximate rendering (left).

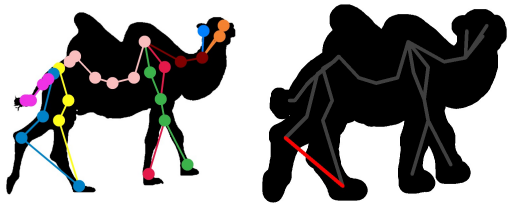


Fig. 3.4 Bone coverage loss. One of the back-right leg joints is incorrectly assigned (left), leading to a large penalty since the lower leg bone crosses outside the dilated silhouette (right).

$L_{\text{cov-sil}}$:

penalizes large silhouette areas with no nearby selected joint. This term requires a precomputed set of silhouette sample points $Z \subseteq \mathbb{R}^2$, which we aim to "cover" as best as possible with the set of selected joints. Intuitively, the silhouette is considered well-covered if all sample points are close to *some* selected joint proposal. The set Z is generated from the medial axis transform (MAT)[11] of the silhouette, $Z^t = \text{MAT}(S^t)$ with a cubed loss strongly penalizing projection outside the silhouette:

$$L_{\text{cov-sil}}(A^t; X^t, Z^t) = \sum_i \min_j \|Z_i^t - \hat{X}_j^t\|^3 \tag{3.9}$$

$L_{\text{cov-bone}}$:

is used to prevent bones crossing the background. The joint hierarchy is stored in a kinematic tree structure $K = \{\{j,k\}$ if joints j,k are connected by a bone$\}$.

$$L_{\text{cov-bone}}(A^t; X^t, S^t, K) = \sum_{\{j,k\} \in K} \left(1 - \min_{\lambda \in [0:0.1:1]} S^t(\hat{X}_j^t + \lambda(\hat{X}_j^t - \hat{X}_k^t))\right) \tag{3.10}$$

GA Solution.

We minimize this more complex objective using a genetic algorithm (GA)[34], which requires defining a fitness function, "genes", an initial population, crossover procedure, and mutation

procedure. The *fitness function* is precisely the energy $L(A)$ given above, and the *genes* are vectors of J integers, rather than one-hot encodings. We begin with a population size of 128 genes, in which the first 32 are set equal to the max confidence solutions given by the network in order to speed up convergence. The remaining 96 are generated by selecting a random proposal for each joint. *Crossover* is conducted as standard by slicing genes in two parts, and pairing first and second parts from different parents to yield the next generation. In each generation, each gene has some probability of undergoing a *mutation*, in which between 1 and 4 joints have new proposals randomly assigned. Weights were set empirically and we run for 1000 generations. Examples of errors corrected by these two energy terms are shown in Fig. 3.3 and Fig. 3.4.

3.5.3 Generative model optimization

The generative model optimization stage refines model parameters to better match the silhouette sequence $\mathscr{S}$, by minimizing an energy which sums 4 terms:

Silhouette energy.

The silhouette energy E_{sil} compares the rendered model to a given silhouette image, given simply by the L2 difference between the OpenDR rendered image and the given silhouette:

$$E_{\text{sil}}(\phi,\theta,\beta;S) = \|S - R\big(\phi * v(\theta,\beta)\big)\| \tag{3.11}$$

Unimodal Prior energy.

The prior term E_{prior} encourages the regressed shape and pose parameters to remain close to a those in the combined artist traininthose in our set of artist 3D dog meshes.

The Mahalanobis distance is used to encourage the model to remain close to: (1) a distribution over shape coefficients given by the mean and covariance of SMAL training samples of the relevant animal family, (2) a distribution of pose parameters built over a walking sequence. The final term ensures the pose parameters remain within set limits.

$$E_{\text{lim}}(\theta) = \max\{\theta - \theta_{\max}, 0\} + \max\{\theta_{\min} - \theta, 0\}. \tag{3.12}$$

Joints energy.

The joints energy E_{joints} compares the rendered model joints to the OJA predictions, and therefore must account for missing and incorrect joints. It is used primarily to stabilize the nonlinear optimization in the initial iterations, and its importance is scaled down as the silhouette term begins to enter its convergence basin.

$$E_{\text{joints}}(\phi, \theta, \beta; X^*) = \|X^* - \phi * \nu(\theta, \beta)\mathtt{K}(:, j)\| \tag{3.13}$$

Temporal energy.

The optimizer for each frame is initialized to the result of that previous. In addition, a simple temporal smoothness term is introduced to penalize large inter-frame variation:

$$E_{\text{temp}}(\phi, \theta, \beta; X^*) = (\phi_t - \phi_{t+1})^2 + (\beta_t - \beta_{t+1})^2 \tag{3.14}$$

The optimization is via a second order dogleg method [57].

3.6 Experiments

Datasets.

In order to quantify our experiments, we introduce a new benchmark animal dataset of joint annotations (BADJA) comprising several video sequences with 2D joint labels and segmentation masks. These sequences were derived from the DAVIS video segmentation dataset [68], as well as additional online stock footage for which segmentations were obtained using Adobe's UltraKey tool [1]. A set of twenty joints on the 3D SMAL mesh were labeled, illustrated in Fig. 3.5. These joints were chosen on the basis of being informative to the skeleton and being simple for a human annotator to localize. To make manual annotation feasible and to ensure a diverse set of data, annotations are provided for every fifth frame.

The video sequences were selected to comprise a range of different quadrupeds undergoing various movement typical of their species. Although the dataset is perhaps insufficient in size to train deep neural networks, the variety in animal shape and pose renders it suitable for evaluating quadruped joint prediction methods.

3.6.1 Joint prediction

For the joint predictor ρ we train a stacked hourglass network [62]. Following state-of-the-art performance on related human 2D pose estimation datasets ([6], [**?**]), we construct a network consisting of 8 stacks, 256 features and 1 block. As input we provide synthetically-generated silhouette images of size 256×256, which are obtained by randomly sampling shape and pose parameters from the SMAL model. The corresponding training targets are ground truth heatmaps produced by smoothing the 2D projected joint locations with a Gaussian kernel. Since we are working with synthetic data, we are able to generate training samples on the fly, resulting in an effectively infinite training set. A small adaptation was required to prevent the network degenerating to an unfavourable solution on silhouette input: foreground masks were applied to both ground truth silhouette and predicted heatmaps to prevent the network degenerating to an all-zero heatmap, which produces a reasonably good loss and prevents the network training successfully. The network was trained using the RMSProp optimizer for 40k iterations with a batch size of 18 and learning rate of 2.5×10^{-4}. The learning rate was decayed by 5% every 10k iterations. Training until convergence took 24 hours on a Nvidia Titan X GPU.

Joint accuracy is evaluated with the Probability of Correct Keypoint (PCK) metric defined by Yang and Ramanan [94]. The PCK is the percentage of predicted keypoints which are within a threshold distance d from the ground truth keypoint location. The threshold distance is given by $d = \alpha\sqrt{|S|}$ where $|S|$ is the area of the silhouette and α is a constant factor which we set to $\alpha = 0.2$ for these experiments.

Fig. 3.2 shows a selection of maximum likelihood joint predictions on real world images. Note that despite being trained only on synthetic data, the network generalizes extremely well to animals in the wild. The performance extends even to species which were not present in the SMAL model, such as the impala and rhino. The network is also robust to challenging poses (3.2b), occlusions (3.2c) and distraction objects such as the human rider in (3.2d). It is however susceptible to situations where the silhouette image is ambiguous, for example if the animal is facing directly towards or away from the camera. Figure 3.10 contains examples of failure modes.

3.6.2 Optimal joint assignment

Following non-maximum suppression of the joint heatmaps obtained in Section 3.6.1, we apply OJA to select an optimal set of joints with which to initialize the final optimization stage. It can be seen that the OJA step is able to address many of the failure cases introduced by the joint prediction network, for example by eliminating physically implausible joint

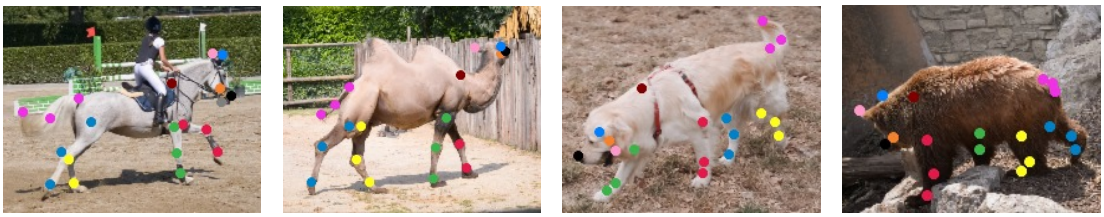


Fig. 3.5 Example joint annotations from the BADJA dataset. A total of 11 video sequences are in the dataset, annotated every 5 frames with 20 joint positions and visibility indicators.

(a) (b) (c)

Fig. 3.6 Example skeletons from raw predictions (a), processed with OJA-QP (b), and OJA-GA (c).

	Raw	QP	GA
bear	83.1	83.7	**88.9**
camel	73.3	74.1	**87.1**
cat	58.5	**60.1**	58.4
cows	89.2	88.4	**94.7**
dog	**66.9**	66.6	**66.9**
horsejump-high	26.5	**27.7**	24.4
horsejump-low	26.9	27.0	**31.9**
tiger	76.5	88.8	**92.3**
rs_dog	64.2	63.4	**81.2**
Average	62.8	64.4	**69.5**

Table 3.1 Accuracy of OJA on BADJA test sequences.

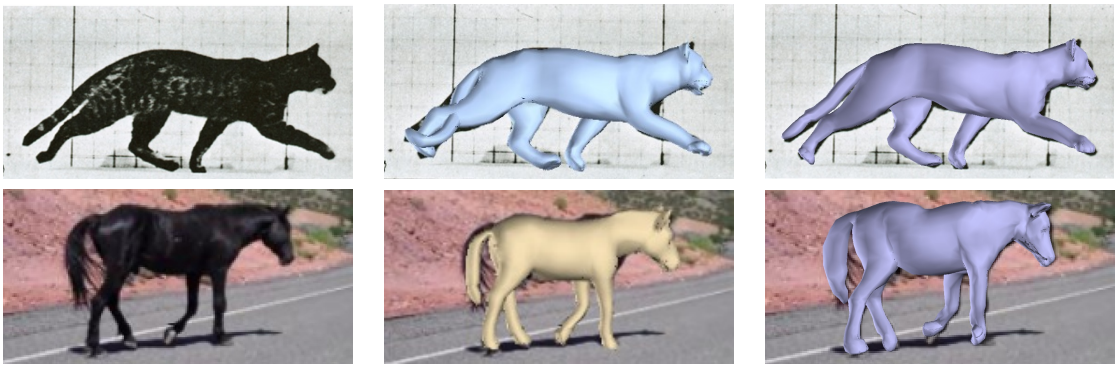
configurations (Fig. 3.6, row 1) or by resolving the ambiguity between the left and right legs (Fig. 3.6, row 2). Table 3.1 summarizes the performance of both the raw network predictions and results of the two OJA methods. Over most of the sequences in the BADJA dataset it can be seen that the use of coverage terms (employed by the OJA-GA model) improves skeleton accuracy. In particular, the bear, camel and rs_dog sequences show substantial improvements. The method does however struggle on the horsejump_high sequence, in which part of the silhouette is occluded by the human rider which adversely affects the silhouette coverage term. Across all sequences the selected OJA-GA method improves joint prediction accuracy by 7% compared to the raw network output.

3.6.3 Model fitting

The predicted joint positions and silhouette are input to the optimization phase, which proceeds in four stages. The first stage solves for the model's global rotation and translation

Seq.	Family	PCK (%)		Mesh	Seq.	Family	PCK (%)		Mesh
		Raw	OJA-GA				Raw	OJA-GA	
01	Felidae	91.8	91.9	38.2	06	Equidae	84.4	84.8	19.2
02	Felidae	94.7	95.0	42.4	07	Bovidae	94.6	95.0	40.6
03	Canidae	87.7	88.0	27.3	08	Bovidae	85.2	85.8	41.5
04	Canidae	87.1	87.4	22.9	09	Hippopotamidae	90.5	90.6	11.8
05	Equidae	88.9	89.8	51.6	10	Hippopotamidae	93.7	93.9	23.8

Table 3.2 Quantitative evaluation on synthetic test sequences. We evaluate the performance of the raw network outputs and quadratic program post-processing using the probability of correct keypoint (PCK) metric (see sec. 3.6.1). We evaluate mesh fitting accuracy by computing the mean distance between the predicted and ground truth vertices.



RGB SMAL [100] **Ours**

Fig. 3.7 Our results are comparable in quality to SMAL [100], but note that we do not require hand-clicked keypoints.

parameters, which positions the camera. We follow SMPLify [**?**] by solving this camera stage for torso points only, which remain largely fixed through shape and pose variation. We then solve for all shape, pose and translation parameters and gradually decrease the emphasis of the priors. The silhouette term is introduced in the penultimate stage, as otherwise we find this can lead to the optimizer finding unsatisfactory local minima.

The final outputs of our optimization pipeline are shown in Fig. 3.9. In each of the cases illustrated the optimizer is able to successfully find a set of pose and shape parameters which, when rendered, closely resembles the input image. The final row of Fig. 3.9 demonstrates the generalizability of the proposed method: the algorithm is able to find a reasonable pose despite no camel figurines being included in the original SMAL model.

Fig. 3.8 Evaluating synthetic data. Green models: ground truth, Orange models: predicted. Frames 5, 10 and 15 of sequence 4 shown. Error on this sequence 22.9.

Comparison to other work.

We compare our approach visually to that given by Zuffi *et al.* [100]. Recall that their results require hand-clicked keypoints whereas ours fits to points predicted automatically by the hourglass network, which was trained on synthetic animal images. Further, their work is optimized for single frame fitting and is tested on animals in simple poses, whereas we instead focus on the more challenging task of tracking animals in video. Fig. 3.7 shows the application of our model to a number of single frame examples from the SMAL result data [100].

Quantitative experiments.

There is no existing ground truth dataset for comparing reconstructed 3D animal meshes, but an estimate of quantitative error is obtained by testing on synthetic sequences for a range of quadruped species. These are generated by randomly deforming the model and varying the camera position to animate animal motion, see Figure 3.8. Table 3.2 shows results on these sequences.

3.6.4 Automatic silhouette prediction

While not the main focus of our work, we are able to perform the full 3D reconstruction process from an input image with no user intervention. We achieve this by using the DeepLabv3+ network [18] as a front-end segmentation engine to automatically generate animal silhouettes. This network was trained on the PASCAL VOC 2012 dataset, which includes a variety of animal quadruped classes. An example result generated using the fully automatic pipeline is shown in Fig. **??**.

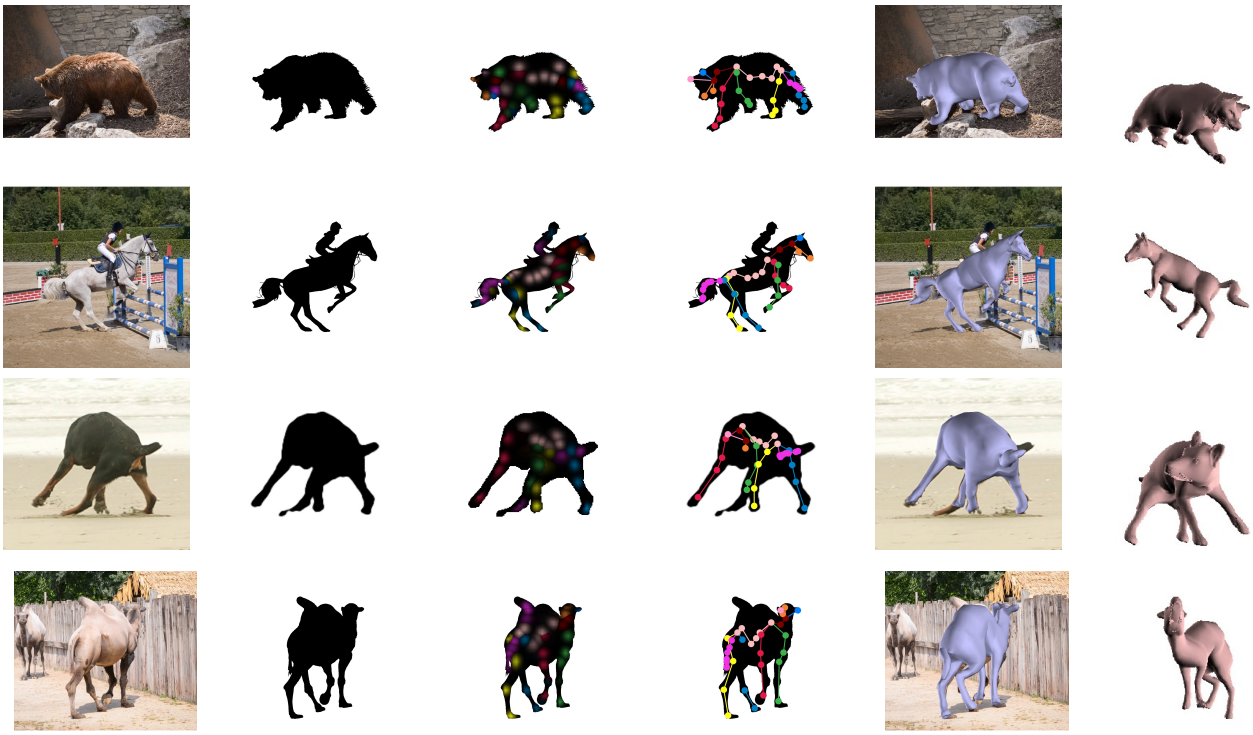


Fig. 3.9 Example results on various animals. From left to right: RGB input, extracted silhouette, network-predicted heatmaps, OJA-processed joints, overlay 3D fit and alternative view

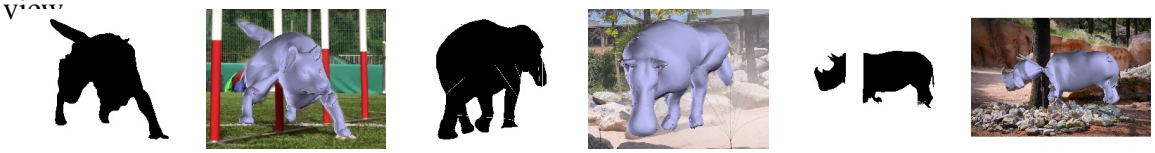


Fig. 3.10 Failure modes of the proposed system. *Left*: Missing interior contours prevent the optimizer from identifying which way the dog is facing. *Middle*: The model has never seen an elephant, so assumes the trunk is the tail. *Right*: Heavy occlusion. The model interprets the tree as background and hence the silhouette term tries to minimize coverage over this region.

3.7 Conclusions

In this work we have introduced a technique for 3D animal reconstruction from video using a quadruped model parameterized in shape and pose. By incorporating automatic segmentation tools, we demonstrated that this can be achieved with no human intervention or prior knowledge of the species of animal being considered. Our method performs well on examples encountered in the real world, generalizes to unseen animal species and is robust to challenging input conditions.

Chapter 4

Precise Shape Reconstructions

4.1 First section of the third chapter

And now I begin my third chapter here ...

4.2 Abstract

We introduce an automatic, end-to-end method for recovering the 3D pose and shape of dogs from monocular internet images. The large variation in shape between dog breeds, significant occlusion and low quality of internet images makes this a challenging problem. We learn a richer prior over shapes than previous work, which helps regularize parameter estimation. We demonstrate results on the Stanford Dog Dataset, an "in-the-wild" dataset of 20,580 dog images for which we have collected 2D joint and silhouette annotations to split for training and evaluation. In order to capture the large shape variety of dogs, we show that the natural variation in the 2D dataset is enough to learn a detailed 3D prior through expectation maximisation (EM). As a by-product of training, we generate a new parameterized model (including limb scaling) SMBLD which we release alongside our new annotation dataset *StanfordExtra* to the research community.

4.3 Introduction

Animals contribute greatly to our society, in numerous ways economic and otherwise (there are more than 63 million pet dogs in the US alone [5]). In consequence, there has been considerable attention in the computer vision research community to the interpretation of imagery of animals. Although these techniques share similarities to techniques for

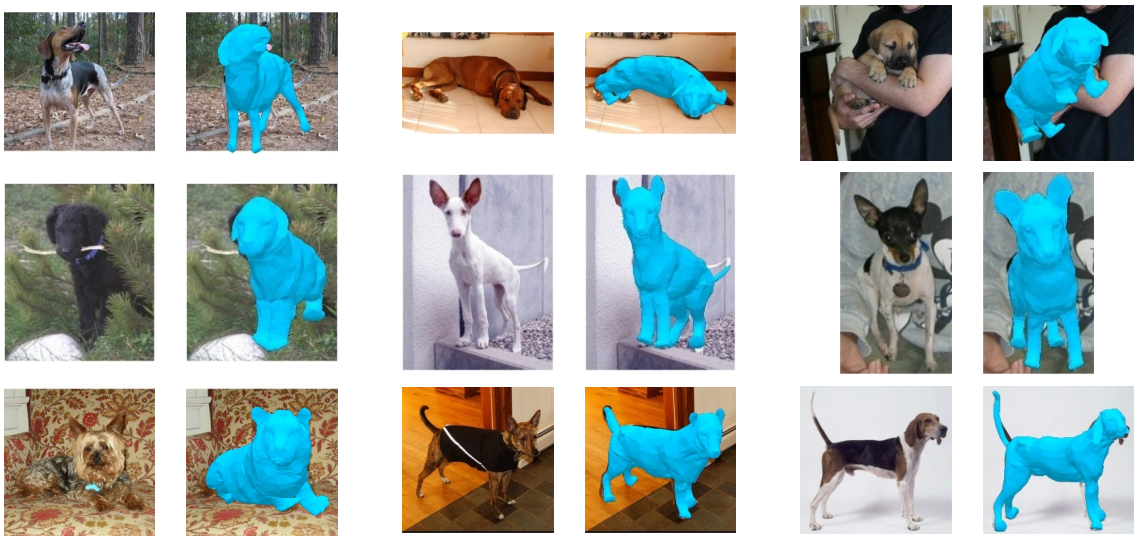


Fig. 4.1 **End-to-end 3D Dog Reconstruction from monocular images.** We propose a novel method that, given a monocular image of a dog can predict a set of parameters for our SMBLD 3D dog model which is consistent with the input. We regularize learning using a multi-modal shape prior, which is tuned during training with an expectation maximization scheme.

understanding images of humans, a key difference is that obtaining labelled training data for animals is more difficult than for humans, because of the wide range of shapes and species of animals, and the difficulty of educating manual labellers in animal physiology.

A particular species of interest is the dog, however it is noticeable that existing work has not yet demonstrated effective 3D reconstruction of dogs over large test sets. We postulate that this is partially because dog breeds are remarkably dissimilar in shape and texture, presenting a challenge to the current state of the art. The methods we propose extend the state of the art in several ways. While each of these qualities exist in some existing works, we believe ours is the first to exhibit this combination, leading to a new state of the art in terms of scale and object diversity.

1. We reconstruct pose and shape on a test set of 1703 low-quality internet images of a complex 3D object class (dogs).

2. We directly regress to object pose and shape from a single image without a model fitting stage.

3. We use easily obtained 2D annotations in training, and none at test time.

4. We incorporate fitting of a new multi-modal prior into the training phase (via EM update steps), rather than fitting it to 3D data as in previous work.

Paper	Animal Class	Training requirements	Template Model	Video required	Test Time Annotation	Model Fitting	Test Size
This paper	Dogs	J2, S2, T3, P3	SMAL	No	None	No	1703
3D-Safari [98]	Zebras, horses	M3 (albeit synthetic), J2, S2, P3	SMAL	3-7 frames / animal	None	Yes	200
Lions, Tigers and Bears (SMALR) [**?**]	MLQ	Not trained	SMAL	3-7 frames / animal	J2, S2	Yes	14
3D Menagerie (SMAL) [**?**]	MLQ	Not trained	SMAL	No	J2, S2	Yes	48
Creatures Great and SMAL [9]	MLQ	Not trained	SMAL	Yes	S2 (for best results shown)	Yes	9
Category Specific Mesh Reconstructions [**?**]	Birds	J2, S2	Bird convex hull	No	None	No	2850
What Shape are Dolphins [16]	Dolphins, Pigeons	Not trained	Dolphin Template	25 frames / category	J2, S2	Yes	25
Animated 3D Creatures [**?**]	MLQ	Not trained	Generalized Cylinders	Yes	J2, S2	Yes	15

Table 4.1 Literature summary: Our paper extends large-scale "in-the-wild" reconstruction to the difficult class of diverse breeds of dogs. MLQ: Medium-to-large quadrupeds. J2: 2D Joints. S2: 2D Silhouettes. T3: 3D Template. P3: 3D Priors. M3: 3D Model.

5. We introduce new degrees of freedom to the SMAL model, allowing explicit scaling of subparts.

4.3.1 Related work

The closest work in terms of scale is the category-specific mesh reconstruction of Kanazawa et al. [**?**], where 2850 images of birds were reconstructed. However doing so for the complex pose and shape variations of dogs required the advances described in this paper.

Table 4.1 summarizes previous work on animal reconstruction. It is interesting to note that while several papers demonstrate reconstruction across species, which *prima facie* is a richer class than just dogs, the test-time requirements (e.g. manually-clicked keypoints/silhouette segmentations, input image quality etc.) are considerably higher for those systems. Thus we claim that the achievement of reconstructing a full range of dog breeds, with variable fur length, varying shape and pose of ears, and with considerable occlusion, is a significant contribution.

Monocular 3D reconstruction of human bodies

The majority of recent work in 3D pose and shape recovery from monocular images tackles the special case of 3D *human* reconstruction. As a result, the research community has collected a multitude of open source human datasets which provide strong supervisory signals for training deep neural networks. These include accurate 3D deformable template models [55] generated from real human scans, 3D motion capture datasets [36**?**] and large 2D datasets [52**?** , 6] which provide keypoint and silhouette annotations.

The abundance of available human data has supported the development of successful monocular 3D reconstruction pipelines [48**?**]. Such approaches rely on accurate 3D data to build detailed priors over the distribution of human shapes and poses, and use large 2D keypoints datasets to promote generalization to "in-the-wild" scenarios. Silhouette data has also been shown to assist in accurate reconstruction of clothes, hair and other appearance detail [74, 4]. While the dominant paradigm in human reconstruction is now end-to-end deep learning methods, SPIN [**?**] show impressive improvement by incorporating an energy minimization process within their training loop to further minimize a 2D reprojection loss subject to fixed pose & shape priors. Inspired by this innovation, we learn an iteratively-improving shape prior by applying expectation maximization during the training process.

Monocular 3D reconstruction of animal categories. While animals are often featured in computer vision literature, there are still relatively few works that focus on accurate 3D animal reconstruction.

A primary reason for this is absence of large scale 3D datasets[1] stemming from the practical challenges associated with 3D motion capture, as well as a lack of 2D data which captures a wide variety of animals. The recent Animal Pose dataset [14] is one such 2D alternative, but contains significantly fewer labelled images than our new StanfordDogs dataset (4,000 compared to 20,580 in). On the other hand, animal silhouette data is plentiful [**?** 25, 45].

Zuffi et al. [**?**] made a significant contribution to 3D animal reconstruction research by releasing SMAL, a deformable 3D quadruped model (analagous to SMPL [55] for human reconstruction) from 41 scans of artist-designed toy figurines. The authors also released shape and pose priors generated from artist data. In this work we develop *SMBLD*, an extension of SMAL that better represents the diverse dog category by adding scale parameters and refining the shape prior using our large image dataset.

While there have been various "model-free" approaches which do not rely on an initial template model to generate the 3D animal reconstruction, these techniques often do not produce a mesh [2, 63] or rely heavily on input 2D keypoints or video at test-time [86, 70].

[1]Released after the submission of this paper, RGBD-Dog dataset [43] is the first open-source 3D motion capture dataset for dogs.

An exception is the end-to-end network of Kanazawa et al. [**?**], although we argue that the bird category exhibits more limited articulation than our dog category.

We instead focus on model-based approaches. The SMAL authors [**?**] demonstrate fitting their deformable 3D model to quadruped species using user-provided keypoint and silhouette dataset. SMALR [**?**] then demonstrated fitting to broader animal categories by incorporating multi-view constraints from video sequences. Biggs et al. [9] overcame the need for hand-clicked keypoints by training a joint predictor on synthetic data. 3D-Safari [98] further improve by training a deep network on synthetic data (built using SMALR [**?**]) to recover detailed zebra shapes in the wild.

A drawback of these approaches is their reliance on a test-time energy-based optimization procedure, which is susceptible to failure with poor quality keypoint/silhouette predictions and increases the computational burden. By contrast our method requires no additional energy-based refinement, and is trained purely from single in-the-wild images. The experimental section of this paper contains a robust comparison between our end-to-end method and relevant optimization-based approaches.

A major impediment to research in 3D animal reconstruction has been the lack of a strong evaluation benchmark, with most of the above methods showing only qualitative evaluations or providing quantitative results on fewer than 50 examples. To remedy this, we introduce *StanfordExtra*, a new large-scale dataset which we hope will drive further progress in the field.

4.4 Parametric animal model

At the heart of our method is a parametric representation of a 3D animal mesh, which is based on the Skinned Multi-Animal Linear (SMAL) model proposed by [**?**]. SMAL is a deformable 3D animal mesh parameterized by shape and pose. The *shape* $\beta \in \mathbb{R}^B$ parameters are PCA coefficients of an undeformed template mesh with limbs in default position. The *pose* $\theta \in \mathbb{R}^P$ parameters meanwhile govern the joint angle rotations (35×3 Rodrigues parameters) which effect the articulated limb movement. The model consists of a linear blend skinning function $F_v : (\theta, \beta) \mapsto V$, which generates a set of vertex positions $V \in \mathbb{R}^{3889 \times 3}$, and a joint function $F_J : (\theta, \beta) \mapsto J$, which generates a set of joint positions $J \in \mathbb{R}^{35 \times 3}$.

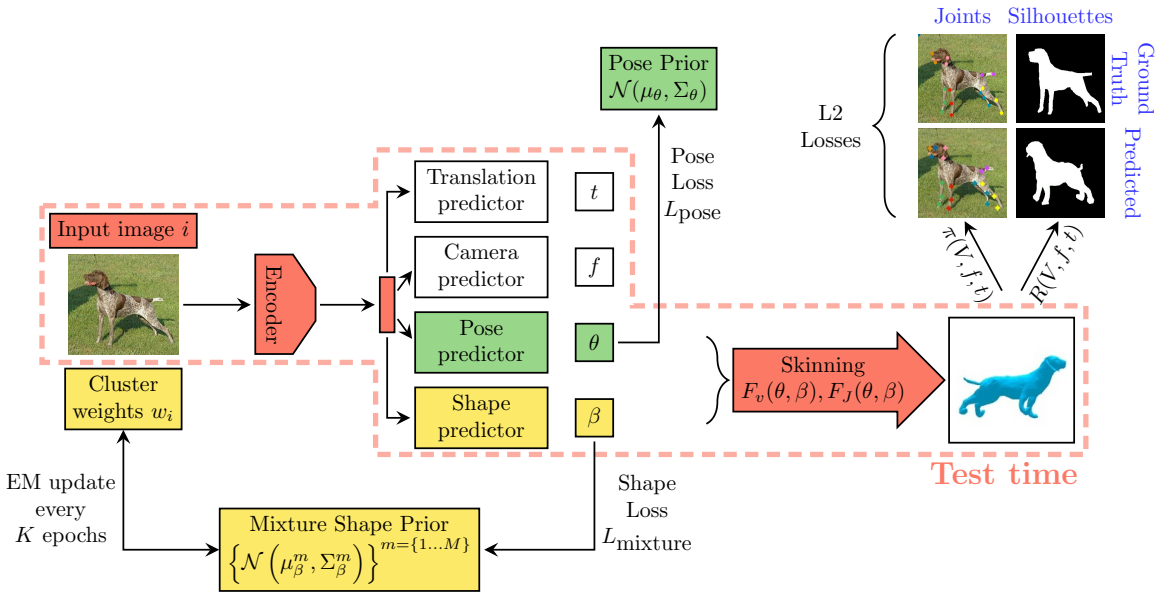


Fig. 4.2 Our method consists of (1) a deep CNN encoder which condenses the input image into a feature vector (2) a set of prediction heads which generate SMBLD parameters for shape β, pose θ, camera focal length f and translation t (3) skinning functions F_v and F_J which construct the mesh from a set of parameters, and (4) loss functions which minimise the error between projected and ground truth joints and silhouettes. Finally, we incorporate a mixture shape prior (5) which regularises the predicted 3D shape and is iteratively updated during training using expectation maximisation. At test time, our system (1) condenses the input image, (2) generates the SMBLD parameters and (3) constructs the mesh.

4.4.1 Introducing scale parameters

While SMAL has been shown to be adequate for representing a variety of quadruped types, we find that the modes of dog variation are poorly captured by the current model. This is unsurprising, since SMAL used only four dogs in its construction.

We therefore introduce a simple but effective way to improve the model's representational power over this particularly diverse animal category. We augment the set of shape parameters β with an additional set κ which independently scale parts of the mesh. For each model joint, we define parameters $\kappa_x, \kappa_y, \kappa_z$ which apply a local scaling of the mesh along the local coordinate x, y, z axes, before pose is applied. Allowing each joint to scale entirely independently can however lead to unrealistic deformations, so we share scale parameters between multiple joints, e.g. leg lengths. The new Skinned Multi-Breed Linear Model for Dogs (SMBLD) is therefore adapted from SMAL by adding 6 scale parameters to the existing set of shape parameters. Figure 4.3 shows how introducing scale parameters increases the flexibility of the SMAL model. We also extend the provided SMAL shape prior (which later

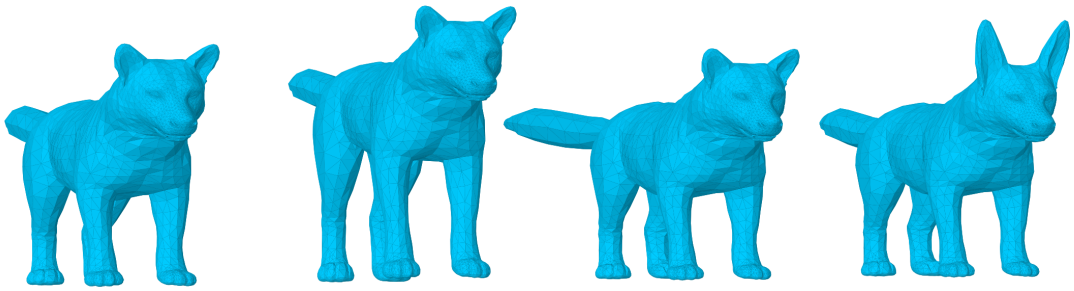


Fig. 4.3 **Effect of varying SMBLD scale parameters**. *From left to right*: Mean SMBLD model, 25% leg elongation, 50% tail elongation, 50% ear elongation.

initializes our EM procedure) to cover the new scale parameters by fitting SMBLD to a set of 13 artist-designed 3D dog meshes. Further details left to the supplementary.

4.5 End-to-end dog reconstruction from monocular images

We now consider the task of reconstructing a 3D dog mesh from a monocular image. We achieve this by training an end-to-end convolutional network that predicts a set of SMBLD model and perspective camera parameters. In particular, we train our network to predict pose θ and shape β SMBLD parameters together with translation t and focal length f for a perspective camera. A complete overview of the proposed system is shown in Figure 4.2.

4.5.1 Model architecture

Our network architecture is inspired by the model of 3D-Safari [98]. Given an input image cropped to (224, 224), we apply a Resnet-50 [33] backbone network to encode a 1024-dimensional feature map. These features are passed through various linear prediction heads to produce the required parameters. The pose, translation and camera prediction modules follow the design of 3D-Safari, but we describe the differences in our shape module.

Pose, translation and camera prediction.

These modules are independent multi-layer perceptrons which map the above features to the various parameter types. As with 3D-Safari we use two linear layers to map to a set of 35×3 3D pose parameters (three parameters for each joint in the SMBLD kinematic tree) given in Rodrigues form. We use independent heads to predict camera frame translation $t_{x,y}$ and depth t_z independently. We also predict the focal length of the perspective camera similarly to 3D-Safari.

Shape and scale prediction.

Unlike 3D-Safari, we design our network to predict the set of shape parameters (including scale) rather than vertex offsets. We observe improvement by handling the standard 20 blend-shape parameters and our new scale parameters in separate linear prediction heads. We retrieve the scale parameters by $\kappa = \exp x$ where x are the network predictions, as we find predicting log scale helps stabilise early training.

4.5.2 Training losses

A common approach for training such an end-to-end system would be to supervise the prediction of (θ, β, t, f) with 3D ground truth annotations [**?** 40, 67]. However, building a suitable 3D annotation dataset would require an experienced graphics artist to design an accurate ground truth mesh for each of 20,520 StanfordExtra dog images, a prohibitive expense.

We instead develop a method that instead relies on *weak 2D supervision* to guide network training. In particular, we rely on only 2D keypoints and silhouette segmentations, are significantly cheaper to obtain.

The rest of this section describes the set of losses used to supervise the network at train time.

Joint reprojection.

The most important loss to promote accurate limb positioning is the joint reprojection loss L_{joints} which compares the projected model joints $\pi(F_J(\theta, \beta), t, f)$ to the ground truth annotations $\hat{X}$. Given the parameters predicted by the network, we apply the SMBLD model to transform the pose and shape parameters into a set of 3D joint positions $J \in \mathbb{R}^{35\times 3}$, and project them to the image plane using translation and camera parameters. The joint loss L_{joints} is given by the ℓ_2 error between the ground truth and projected joints:

$$L_{\text{joints}}(\theta, \beta, t, f; \hat{X}) = \|\hat{X} - \pi(F_J(\theta, \beta), t, f)\|_2 \tag{4.1}$$

Note that many of our training images exhibit significant occlusion, so $\hat{X}$ contains many invisible joints. We handle this by masking L_{joints} to prevent invisible joints contributing to the loss.

Silhouette loss.

The silhouette loss L_{sil} is used to promote shape alignment between the SMBLD dog mesh and the input dog. In order to compute the silhouette loss, we define a rendering function $R : (v,t,f) \mapsto S$ which projects the SMBLD mesh to produce a binary segmentation mask. In order to allow derivatives to be propagated through R, we implement R using the differentiable Neural Mesh Renderer [42]. The loss is computed as the ℓ_2 difference between a projected silhouette and the ground truth mask $\hat{S}$:

$$L_{\text{sil}}(\theta,\beta,t,f;\hat{S}) = \|\hat{S} - R\big(F_V(\theta,\beta),t,f\big)\|_2 \tag{4.2}$$

Priors.

In the absence of 3D ground truth training data, we rely on priors obtained from artist graphics models to encourage realism in the network predictions. We model both pose and shape using a multivariate Gaussian prior, consisting of means μ_θ, μ_β and covariance matrices $\Sigma_\theta, \Sigma_\beta$. The loss is given as the log likelihood of a given shape or pose vector under these distributions, which corresponds to the Mahalanobis distance between the predicted parameters and their corresponding means:

$$L_{\text{pose}}(\theta;\mu_\theta,\Sigma_\theta) = (\theta - \mu_\theta)^T \Sigma_\theta^{-1} (\theta - \mu_\theta) \tag{4.3}$$

$$L_{\text{shape}}(\beta;\mu_\beta,\Sigma_\beta) = (\beta - \mu_\beta)^T \Sigma_\beta^{-1} (\beta - \mu_\beta) \tag{4.4}$$

Unlike previous work, we find there is no need to use a loss to penalize pose parameters if they exceed manually specified joint angle limits. We suspect our network learns this regularization naturally because of our large dataset.

4.5.3 Learning a multi-modal shape prior.

The previous section introduced a unimodal, multivariate Gaussian shape prior, based on mean μ_β and covariance matrix Σ_β. However, we find enforcing this prior throughout training tends to result in predictions which appear similar in 3D shape, even when tested on dog images of different breeds. We propose to improve diversity among predicted 3D dog shapes by extending the above formulation to a Mixture of M Gaussians prior. The mixture shape

loss is then given as:

$$L_{\text{mixture}}(\beta_i;\mu_\beta,\Sigma_\beta,\Pi_\beta) = \sum_{m=1}^{M} \Pi_\beta^m L_{\text{shape}}(\beta_i;\mu_\beta^m,\Sigma_\beta^m) \tag{4.5}$$

Where μ_β^m, Σ_β^m and Π_β^m are the mean, covariance and mixture weight respectively for Gaussian component m. For each component the mean is sampled from our existing unimodal prior and the covariance is set equal to the unimodal prior i.e. $\Sigma_\beta^m := \Sigma_\beta$. All mixture weights are initially set to $\frac{1}{M}$.

Each training image i is assigned a set of latent variables $\{w_i^1,\dots w_i^M\}$ encoding the probability of the dog shape in image i being generated by component m.

4.5.4 Expectation Maximization in the loop

As previously discussed, our initial shape prior is obtained from artist data which we find is unrepresentative of the diverse shapes present in our real dog dataset. We address this by proposing to recover the latent variables w_i^m and parameters (μ_β^m, Σ_β^m and Π_β^m) of our 3D shape prior by learning from monocular images of in-the-wild dogs and their 2D training labels in our training dataset.

We achieve this using Expectation Maximization (EM), which regularly updates the means and variances for each mixture component and per-image mixture weights based on the observed shapes in the training set. While training our 3D reconstruction network, we progressively update our shape mixture model with an alternating 'E' step and 'M' step described below:

The 'E' Step.

The 'E' step computes the expected value of the latent variables w_i^m assuming fixed $(\mu_\beta^m,\Sigma_\beta^m,\Pi_\beta^m)$ for all $i \in \{1,\dots,N\}, m \in \{1,\dots,M\}$.

The update equation for an image i with latest shape prediction β_i and cluster m with parameters $(\mu_\beta^m,\Sigma_\beta^m,\Pi_\beta^m)$ is given as:

$$w_i^m := \frac{\mathcal{N}(\beta_i|\mu_\beta^m,\Sigma_\beta^m)\Pi_\beta^m}{\sum_{m'}^{M}\mathcal{N}(\beta_i|\mu_\beta^{m'},\Sigma_\beta^{m'})\Pi_\beta^{m'}} \tag{4.6}$$

The 'M' Step.

The 'M' step computes new values for $(\mu_\beta^m, \Sigma_\beta^m, \Pi_\beta^m)$, assuming fixed w_i^m for all $i \in \{1,\ldots,N\}, m \in \{1,\ldots,M\}$.

The update equations are given as follows:

$$\mu_\beta^m := \frac{\sum_i w_i^m \beta_i}{\sum_i w_i^m} \quad \Sigma_\beta^m := \frac{\sum_i w_i^m (\beta_i - \Sigma_\beta^m)(\beta_i - \Sigma_\beta^m)^T}{\sum_i w_i^m} \quad \Pi_\beta^m := \frac{1}{N}\sum_i w_i^m \tag{4.7}$$

4.6 Experiments

In this section we compare our method to competitive baselines. We begin by describing our new large-scale dataset of annotated dog images, followed by a quantitative and qualitative evaluation.

4.6.1 StanfordExtra: A new large-scale dog dataset with 2D keypoint and silhouette annotations

Fig. 4.4 **StanfordExtra example images**. *Left*: outlined segmentations and labelled keypoints for 24 representative images. *Right*: heatmap of deviation of worker submitted results from mean for each submission.

In order to evaluate our method, we introduce *StanfordExtra*: a new large-scale dataset with annotated 2D keypoints and binary segmentation masks for dogs. We opted to take source images from the existing Stanford Dog Dataset [46], which consists of 20,580 dog images taken "in the wild" and covers 120 dog breeds. The dataset contains vast shape and pose variation between dogs, as well as nuisance factors such as self/environmental occlusion, interaction with humans/other animals and partial views. Figure 4.4 (left) shows samples from the new dataset.

We used Amazon Mechanical Turk to collect a binary silhouette mask and 20 keypoints per image: 3 per leg (knee, ankle, toe), 2 per ear (base, tip), 2 per tail (base, tip), 2 per

face (nose and jaw). We can approximate the difficulty of the dataset by analysing the variance between 3 annotators at both the joint labelling and silhouette task. Figure 4.4 (right) illustrates typical per-joint variance in joint labelling. Further details of the data curation procedure are left to the supplementary materials.

4.6.2 Evaluation protocol

Our evaluation is based on our new StanfordExtra dataset. In line with other methods which tackle "in-the-wild" 3D reconstruction of articulated subjects [**?** 48], we filter images from the original dataset of 20,580 for which the majority of dog keypoints are invisible. We consider these images unsuitable for our full-body dog reconstruction task. We also remove images for which the consistency in keypoint/silhouette segmentations between the 3 annotators is below a set threshold. This leaves us with 8,476 images which we divide per-breed into an 80%/20% train and test split.

We consider two primary evaluation metrics. IoU is the intersection-over-union of the projected model silhouette compared to the ground truth annotation and indicates the quality of the reconstructed 3D shape. Percentage of Correct Keypoints (PCK) computes the percentage of joints which are within a normalized distance (based on square root of 2D silhouette area) to the ground truth locations, and evaluates the quality of reconstructed 3D pose. We also produce PCK results on various joint groups (legs, tail, ears, face) to compare the reconstruction accuracy for different parts of the dog model.

4.6.3 Training procedure

We train our model in two stages. The first omits the silhouette loss which we find can lead the network to unsatisfactory local minima if applied too early. With the silhouette loss turned off, we find it satisfactory to use the simple unimodal prior (and without EM) for this preliminary stage since there is no loss to specifically encourage a strong shape alignment. After this, we introduce the silhouette loss, the mixture prior and begin applying the expectation maximization updates over $M = 10$ clusters. We train the first stage for 250 epochs, the second stage for 150 and apply the EM step every 50 epochs. All losses are weighted, as described in the supplementary. The entire training procedure takes 96 hours on a single P100 GPU.

4.6.4 Comparison to baselines

We first compare our method to various baseline methods. SMAL [**?**] is an approach which fits the 3D SMAL model using per-image energy minimization. Creatures Great and SMAL (CGAS) [9] is a three-stage method, which employs a joint predictor on silhouette renderings from synthetic 3D dogs, applies a genetic algorithm to clean predictions, and finally applies the SMAL optimizer to produce the 3D mesh.

At test-time both SMAL and CGAS rely on manually-provided segementation masks, and SMAL also relies on hand-clicked keypoints. In order to produce a fair comparison, we produce a set of *predicted* keypoints for StanfordExtra by training the Stacked Hourglass Network [62] with 8 stacks and 1 block, and *predicted* segmentation masks using DeepLab v3+ [**?**]. The Stacked Hourglass Network achieves 71.4% PCK score, DeepLab v3+ achieves 83.4% IoU score and the CGAS joint predictor achieves 41.8% PCK score.

Table 4.2 and Figure 4.5 show the comparison against competitive methods. For full examination, we additionally provide results for SMAL and CGAS in the scenario that ground-truth keypoints and/or segmentations are available at test time.

The results show our end-to-end method outperforms the competitors when they are provided with predicted keypoints/segmentations (white rows). Our method therefore achieves a new state-of-the-art on this 3D reconstruction task. In addition, we show our method achieves improved average IoU/PCK scores than competitive methods, even when they are provided ground truth annotations at test time (grey rows). We also demonstrate wider applicability of two contributions from our work (scale parameters and improved prior) by showing improved performance of the SMAL method when these are incorporated. Finally, our model's test-time speed is significantly faster than the competitors as it does not require an optimizer.

4.6.5 Generalization to unseen dataset

Table **??** shows an experiment to compare how well our model generalizes to a new data domain. We test our model against the SMAL [**?**] method (using predicted keypoints and segmentations as above for fairness) on the recent Animal Pose dataset [14]. The data preparation process is the same as for StanfordExtra and no fine-tuning was used for either method. We achieve good results in this unseen domain and still improve over the SMAL optimizer.

Method	Kps	Seg	IoU	PCK				
				Avg	Legs	Tail	Ears	Face
SMAL [**?**]	Pred	Pred	67.9	67.1	65.7	79.5	54.9	87.4
SMAL	GT	GT	69.2	72.6	69.9	**92.0**	58.6	**96.9**
SMAL	GT	Pred	68.6	72.6	70.2	91.5	58.1	**96.9**
SMAL	Pred	GT	68.5	67.4	66.0	79.9	55.0	88.2
CGAS [9]	CGAS	Pred	62.4	43.7	46.5	64.1	36.5	21.4
CGAS	CGAS	GT	63.1	43.6	46.3	64.2	36.3	21.6
SMAL + scaling	Pred	Pred	69.3	69.6	69.4	79.3	56.5	87.6
SMAL + scaling + new prior	Pred	Pred	70.7	71.6	71.5	80.7	59.3	88.0
Ours	—	—	**73.6**	**75.7**	**75.0**	77.6	**69.9**	90.0

Table 4.2 **Baseline comparisons.** Both PCK and silhouette IOU scores are shown for SOTA methods under varying conditions. A combination of both ground truth (GT) and predicted (Pred) keypoints/segmentations using hourglass network and deeplab respectively. For the CGAS method we also test using their keypoint predictor (CGAS). The addition of scaling and new prior are shown to improve the original SMAL method.

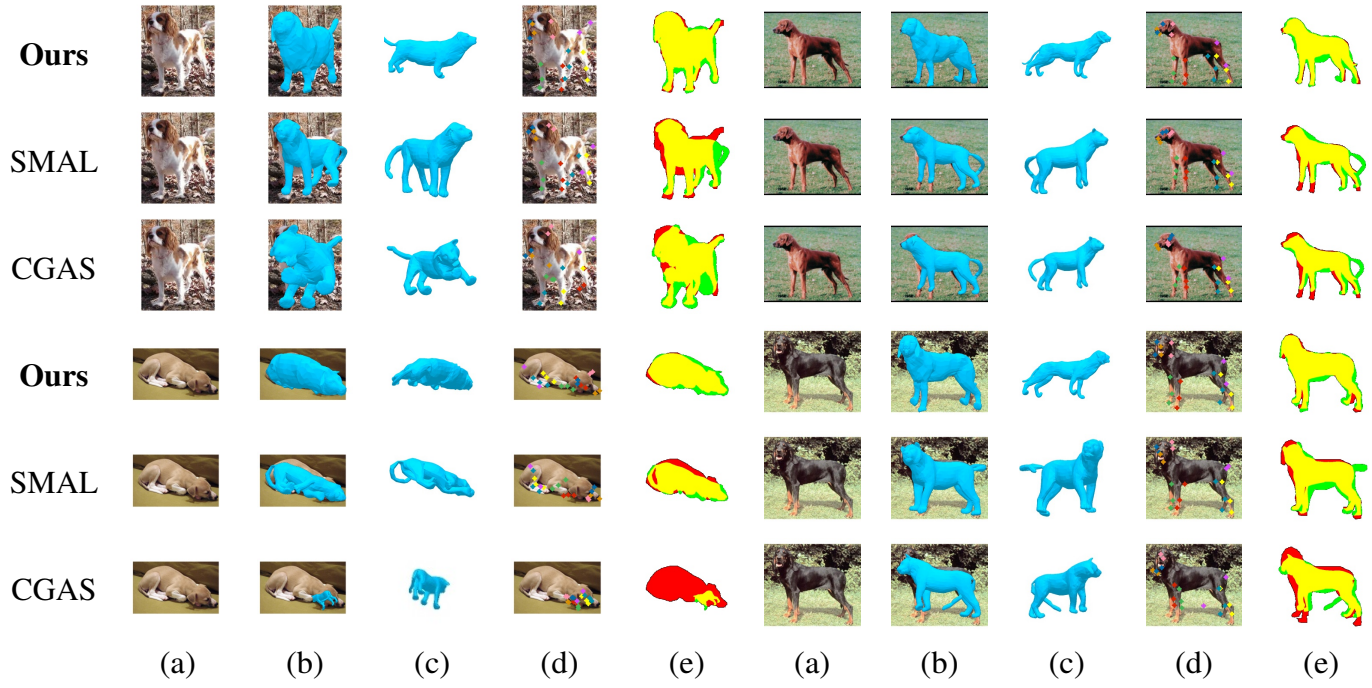


Fig. 4.5 **Qualitiative comparison to SOTA.** Row 1: **Ours**, Row 2: SMAL [**?**], Row 3: CGAS [9]. (a) input image, (b) predicted 3D mesh, (c) canonical view 3D mesh, (d) reprojected model joints and (e) silhouette reprojection error.

4.6.6 Ablation study

We also produce a study in which we ablate individual components of our method and examine the effect on the PCK/IoU performance. We evaluate three variants: (1) **Ours w/o EM** that omits EM updates, (2) **Ours w/o MoG** which replaces our mixture shape prior with a unimodal prior, (3) **Ours w/o Scale** which removes the scale parameters.

The results in Table **??** indicate that each individual component has a positive impact on the overall method performance. In particular, it can be seen that the inclusion of the EM and Mixture of Gaussians prior leads to an improvement in IoU, suggesting that the shape prior refinements steps help the model accurately fit the exact dog shape. Interestingly, we notice that adding the Mixture of Gaussians prior but omitting EM steps slightly hinders performance, perhaps due to an sub-optimal initialization for the M clusters. However, we find adding EM updates to the Mixture of Gaussian model improves all metrics except the ear keypoint accuracy. We observe the error here is caused by the our shape prior learning slightly imprecise shapes for dogs with extremely "floppy" ears. Although there is good silhouette coverage for these regions, the fact our model has only a single articulation point per ear causes a lack of flexibility that results in occasionally misplaced ear tips for these instances. This could be improved in future work by adding additional model joints to the ear. Finally, we find the increased model flexibility afforded by the SMBLD scale parameters have a positive effect on IoU/PCK scores.

4.6.7 Qualitative evaluation

Figure 4.5 shows a range of example system outputs when tested on range of StanfordExtra and Animal Pose [14] dogs with varying pose and shape and in challenging conditions. Note that only StanfordExtra is used for training.

4.7 Conclusions

This paper presents an end-to-end method for automatic, monocular 3D dog reconstruction. We achieve this using only weak 2D supervision, provided by our novel StanfordExtra dataset. Further, we show we can learn a more detailed shape prior by tuning a gaussian mixture during model training and this leads to improved reconstructions. We also show our method improves over competitive baselines, even when they are given access to ground truth data at test time.

Future work should involve tackling some failure cases of our system, for example handling multiple overlapping dogs or dealing with heavy motion blur. Other areas for

research include extending our EM formulation to handle video input to take advantage of multi-view shape constraints, and transferring knowledge accumulated through training on StanfordExtra dogs to other species.

4.8 Acknowlegements

The authors would like to thank the GSK AI team for providing access to their GPU cluster, Michael Sutcliffe, Thomas Roddick, Matthew Allen and Peter Fisher for useful technical discussions, and the GSK TDI group for project sponsorship.

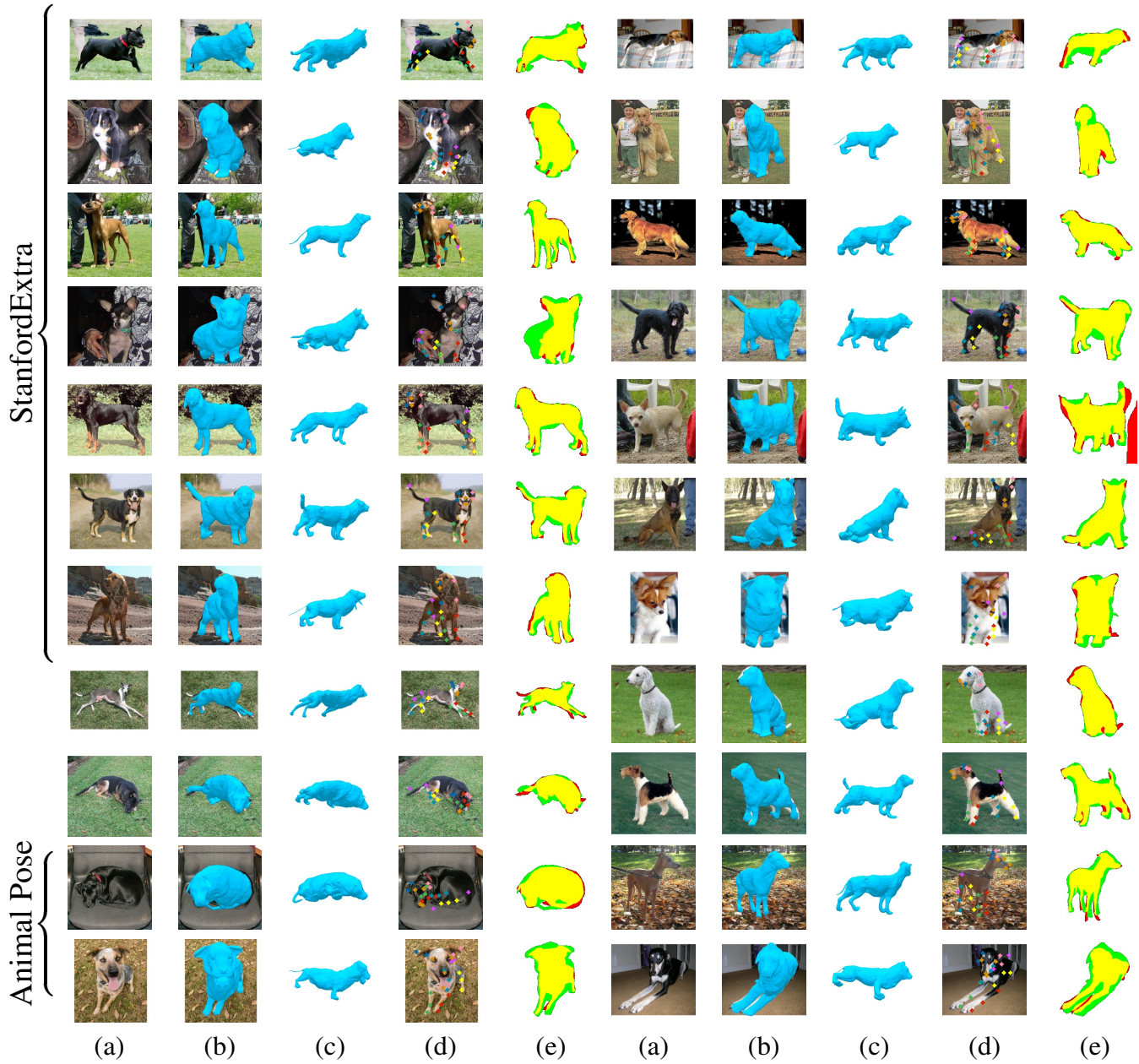


Fig. 4.6 **Qualitative results on StanfordExtra and Animal Pose [14].** For each sample we show: (a) input image, (b) predicted 3D mesh, (c) canonical view 3D mesh, (d) reprojected model joints and (e) silhouette reprojection error.

Chapter 5

Handling Ambiguities

5.1 First section of the third chapter

In this section, blah blah.

5.2 abstract

We consider the problem of obtaining dense 3D reconstructions of humans from single and partially occluded views. In such cases, the visual evidence is usually insufficient to identify a 3D reconstruction uniquely, so we aim at recovering several plausible reconstructions compatible with the input data. We suggest that ambiguities can be modelled more effectively by parametrizing the possible body shapes and poses via a suitable 3D model, such as SMPL for humans. We propose to learn a multi-hypothesis neural network regressor using a best-of-M loss, where each of the M hypotheses is constrained to lie on a manifold of plausible human poses by means of a generative model. We show that our method outperforms alternative approaches in ambiguous pose recovery on standard benchmarks for 3D humans, and in heavily occluded versions of these benchmarks.

5.3 Introduction

We are interested in reconstructing 3D human pose from the observation of single 2D images. As humans, we have no problem in predicting, at least approximately, the 3D structure of most scenes, including the pose and shape of other people, even from a single view. However, 2D images notoriously [27] do not contain sufficient geometric information to allow recovery of the third dimension. Hence, single-view reconstruction is only possible in a probabilistic

Fig. 5.1 **Human mesh recovery in an ambiguous setting.** We propose a novel method that, given an occluded input image of a person, outputs the set of meshes which constitute plausible human bodies that are consistent with the partial view. The ambiguous poses are predicted using a novel *n*-quantized-best-of-*M* method.

sense and the goal is to make the posterior distribution as sharp as possible, by learning a strong prior on the space of possible solutions.

Recent progress in single-view 3D pose reconstruction has been impressive. Methods such as HMR [40], GraphCMR [48] and SPIN [47] formulate this task as learning a deep neural network that maps 2D images to the parameters of a 3D model of the human body, usually SMPL [55]. These methods work well in general, but not always (fig. 5.4). Their main weakness is processing *heavily occluded images* of the object. When a large part of the object is missing, say the lower body of a sitting human, they output reconstructions that are often implausible. Since they can produce only one hypothesis as output, they very likely learn to approximate the mean of the posterior distribution, which may not correspond to any plausible pose. Unfortunately, this failure modality is rather common in applications due to scene clutter and crowds.

In this paper, we propose a solution to this issue. Specifically, we consider the challenge of recovering 3D mesh reconstructions of complex articulated objects such as humans from highly ambiguous image data, often containing significant occlusions of the object. Clearly, it is generally impossible to reconstruct the object uniquely if too much evidence is missing; however, we can still predict a *set* containing all possible reconstructions (see fig. 5.3), making this set as small as possible. While ambiguous pose reconstruction has been previously investigated, as far as we know, this is the first paper that looks specifically at a deep learning approach for ambiguous reconstructions of the *full human mesh*.

Our primary contribution is to introduce a principled multi-hypothesis framework to model the ambiguities in monocular pose recovery. In the literature, such multiple-hypotheses networks are often trained with a so-called *best-of-M* loss — namely, during training, the loss is incurred only by the best of the M hypothesis, back-propagating gradients from that alone [32]. In this work we opt for the *best-of-M* approach since it has been show to outperform alternatives (such as variational auto-encoders or mixture density networks) in

tasks that are similar to our 3D human pose recovery, and which have constrained output spaces [73].

A major drawback of the *best-of-M* approach is that it only guarantees that *one* of the hypotheses lies close to the correct solution; however, it says nothing about the plausibility, or lack thereof, of the *other* $M-1$ hypotheses, which can be arbitrarily 'bad'.[1] Not only does this mean that most of the hypotheses may be uninformative, but in an application we are also unable to tell *which* hypothesis should be used, and we might very well pick a 'bad' one. This has also a detrimental effect during learning because it makes gradients sparse as prediction errors are back-propagated only through one of the M hypotheses for each training image.

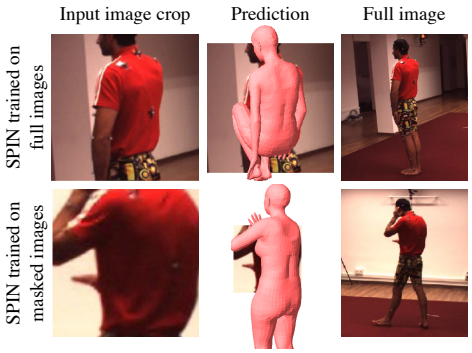


Fig. 5.2 **Top**: Pretrained SPIN model tested on an ambiguous example, **Bottom**: SPIN model after fine-tuning to ambiguous examples. Note the network tends to regress to the mean over plausible poses, shown by predicting the missing legs vertically downward — arguably the average position over the training dataset.

In order to address these issues, our first contribution is a *hypothesis reprojection loss* that forces each member of the multi-hypothesis set to correctly reproject to 2D image keypoint annotations. The main benefit is to constrain the *whole* predicted set of meshes to be consistent with the observed image, not just the best hypothesis, also addressing gradient sparsity.

Next, we observe that another drawback of the best-of-M pipelines is to be tied to a particular value of M, whereas in applications we are often interested in tuning the number of hypothesis considered. Furthermore, minimizing the reprojection loss makes hypotheses geometrically consistent with the observation, but not necessarily likely. Our second contribution is thus to improve the flexibility of best-of-M models by allowing them to output any smaller number $n < M$ of hypotheses while at the same time making these hypotheses *more representative of likely* poses. The new method, which we call n-quantized-best-of-M, does so by quantizing the best-of-M model to output weighed by a *explicit pose prior*, learned by means of normalizing flows.

[1] Theoretically, best-of-M can minimize its loss by quantizing optimally (in the sense of minimum expected distortion) the posterior distribution, which would be desirable for coverage. However, this is *not* the only solution that optimizes the best-of-M training loss, as in the end it is sufficient that *one* hypothesis per training sample is close to the ground truth. In fact, this is exactly what happens; for instance, during training hypotheses in best-of-M are known to easily become degenerate and 'die off', a clear symptom of this problem.

To summarise, our key contributions are as follows. First, we deal with the challenge of 3D mesh reconstruction for articulated objects such as humans in *ambiguous* scenarios. Second, we introduce a *n-quantized-best-of-M* mechanism to allow best-of-*M* models to generate an arbitrary number of $n < M$ predictions. Third, we introduce a mode-wise re-projection loss for multi-hypothesis prediction, to ensure that predicted hypotheses are *all* consistent with the input.

Empirically, we achieve state-of-the-art monocular mesh recovery accuracy on Human36M, its more challenging version augmented with heavy occlusions, and the 3DPW datasets. Our ablation study validates each of our modelling choices, demonstrating their positive effect.

5.4 Introduction

We are interested in reconstructing 3D human pose from the observation of single 2D images. As humans, we have no problem in predicting, at least approximately, the 3D structure of most scenes, including the pose and shape of other people, even from a single view. However, 2D images notoriously [27] do not contain sufficient geometric information to allow recovery of the third dimension. Hence, single-view reconstruction is only possible in a probabilistic sense and the goal is to make the posterior distribution as sharp as possible, by learning a strong prior on the space of possible solutions.

Recent progress in single-view 3D pose reconstruction has been impressive. Methods such as HMR [40], GraphCMR [48] and SPIN [47] formulate this task as learning a deep neural network that maps 2D images to the parameters of a 3D model of the human body, usually SMPL [55]. These methods work well in general, but not always (fig. 5.4). Their main weakness is processing *heavily occluded images* of the object. When a large part of the object is missing, say the lower body of a sitting human, they output reconstructions that are often implausible. Since they can produce only one hypothesis as output, they very likely learn to approximate the mean of the posterior distribution, which may not correspond to any plausible pose. Unfortunately, this failure modality is rather common in applications due to scene clutter and crowds.

In this paper, we propose a solution to this issue. Specifically, we consider the challenge of recovering 3D mesh reconstructions of complex articulated objects such as humans from highly ambiguous image data, often containing significant occlusions of the object. Clearly, it is generally impossible to reconstruct the object uniquely if too much evidence is missing; however, we can still predict a *set* containing all possible reconstructions (see fig. 5.3), making this set as small as possible. While ambiguous pose reconstruction has been

Fig. 5.3 **Human mesh recovery in an ambiguous setting.** We propose a novel method that, given an occluded input image of a person, outputs the set of meshes which constitute plausible human bodies that are consistent with the partial view. The ambiguous poses are predicted using a novel *n*-quantized-best-of-*M* method.

previously investigated, as far as we know, this is the first paper that looks specifically at a deep learning approach for ambiguous reconstructions of the *full human mesh*.

Our primary contribution is to introduce a principled multi-hypothesis framework to model the ambiguities in monocular pose recovery. In the literature, such multiple-hypotheses networks are often trained with a so-called *best-of-M* loss — namely, during training, the loss is incurred only by the best of the M hypothesis, back-propagating gradients from that alone [32]. In this work we opt for the *best-of-M* approach since it has been show to outperform alternatives (such as variational auto-encoders or mixture density networks) in tasks that are similar to our 3D human pose recovery, and which have constrained output spaces [73].

A major drawback of the *best-of-M* approach is that it only guarantees that *one* of the hypotheses lies close to the correct solution; however, it says nothing about the plausibility, or lack thereof, of the *other* $M-1$ hypotheses, which can be arbitrarily 'bad'.[2] Not only does this mean that most of the hypotheses may be uninformative, but in an application we are also unable to tell *which* hypothesis should be used, and we might very well pick a 'bad' one. This has also a detrimental effect during learning because it makes gradients sparse as prediction errors are back-

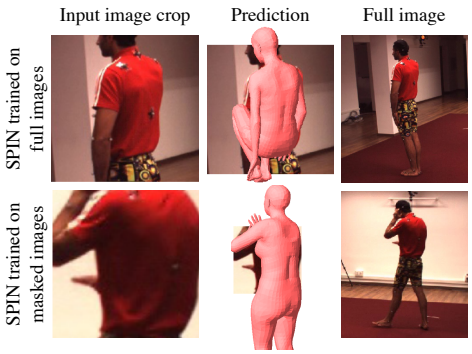


Fig. 5.4 **Top**: Pretrained SPIN model tested on an ambiguous example, **Bottom**: SPIN model after fine-tuning on ambiguous examples. Note the network tends to regress to the mean over plausible poses, shown by predicting the missing legs vertically downward — arguably the average position over the training dataset.

[2] Theoretically, best-of-M can minimize its loss by quantizing optimally (in the sense of minimum expected distortion) the posterior distribution, which would be desirable for modeling ambiguous tasks. The only solution that optimizes the best-of-M training loss, as in the end it is sufficient that *one* hypothesis per training sample is close to the ground truth. In fact, this is exactly what happens; for instance, during training hypotheses in best-of-M are known to easily become degenerate and 'die off', a clear symptom of this problem.

propagated only through one of the M hypotheses for each training image.

In order to address these issues, our first contribution is a *hypothesis reprojection loss* that forces each member of the multi-hypothesis set to correctly reproject to 2D image keypoint annotations. The main benefit is to constrain the *whole* predicted set of meshes to be consistent with the observed image, not just the best hypothesis, also addressing gradient sparsity.

Next, we observe that another drawback of the best-of-M pipelines is to be tied to a particular value of M, whereas in applications we are often interested in tuning the number of hypothesis considered. Furthermore, minimizing the reprojection loss makes hypotheses geometrically consistent with the observation, but not necessarily likely. Our second contribution is thus to improve the flexibility of best-of-M models by allowing them to output any smaller number $n < M$ of hypotheses while at the same time making these hypotheses *more representative of likely* poses. The new method, which we call n-quantized-best-of-M, does so by quantizing the best-of-M model to output weighed by a *explicit pose prior*, learned by means of normalizing flows.

To summarise, our key contributions are as follows. First, we deal with the challenge of 3D mesh reconstruction for articulated objects such as humans in *ambiguous* scenarios. Second, we introduce a *n-quantized-best-of-M* mechanism to allow best-of-M models to generate an arbitrary number of $n < M$ predictions. Third, we introduce a mode-wise reprojection loss for multi-hypothesis prediction, to ensure that predicted hypotheses are *all* consistent with the input.

Empirically, we achieve state-of-the-art monocular mesh recovery accuracy on Human36M, its more challenging version augmented with heavy occlusions, and the 3DPW datasets. Our ablation study validates each of our modelling choices, demonstrating their positive effect.

5.5 Related work

There is ample literature on recovering the pose of 3D models from images. We break this into five categories: methods that reconstruct 3D points directly, methods that reconstruct the parameters of a 3D model of the object via optimization, methods that do the latter via learning-based regression, hybrid methods and methods which deal with uncertainty in 3D human reconstruction.

Reconstructing 3D body points without a model. Several papers have focused on the problem of estimating 3D body points from 2D observations [8, 61, 72, 82, 48]. Of these, Martinez et al. [58] introduced a particularly simple pipeline based on a shallow neural network. In this work, we aim at recovering the full 3D surface of a human body, rather than only lifting sparse keypoints.

Fitting 3D models via direct optimization. Several methods *fit* the parameters of a 3D model such as SMPL [54] or SCAPE [8] to 2D observations using an optimization algorithm to iteratively improve the fitting quality. While early approaches such as [30, 78] required some manual intervention, the SMPLify method of Bogo et al. [12] was perhaps the first to fit SMPL to 2D keypoints fully automatically. SMPL was then extended to use silhouette, multiple views, and multiple people in [49, 35, 96]. Recent optimization methods such as [39, 66, 92] have significantly increased the scale of the models and data that can be handled.

Fitting 3D models via learning-based regression. More recently, methods have focused on regressing the parameters of the 3D models directly, *in a feed-forward manner*, generally by learning a deep neural network [83, 84, 64, 67, 40]. Due to the scarcity of 3D ground truth data for humans in the wild, most of these methods train a deep regressor using a mix of datasets with 3D and 2D annotations in form of 3D MoCap markers, 2D keypoints and silhouettes. Among those, HMR of Kanazawa et al. [40] and GraphCMR of Kolotouros et al. [48] stand out as particularly effective.

Hybrid methods. Other authors have also combined optimization and learning-based regression methods. In most cases, the integration is done by using a deep regressor to initialize the optimization algorithm [78, 49, 72, 67, 85]. However, recently Kolotouros et al. [47] has shown strong results by integrating the optimization loop in learning the deep neural network that performs the regression, thereby exploiting the weak cues available in 2D keypoints.

Modelling ambiguities in 3D human reconstruction. Several previous papers have looked at the problem of modelling ambiguous 3D human pose reconstructions. Early work includes Sminchisescu and Triggs [80], Sidenbladh et al. [77] and Sminchisescu et al. [79].

More recently, Akhter and Black [3] learn a prior over human skeleton joint angles (but not directly a prior on the SMPL parameters) from a MoCap dataset. Li and Lee [50] use the Mixture Density Networks model of [10] to capture ambiguous 3D reconstructions of sparse

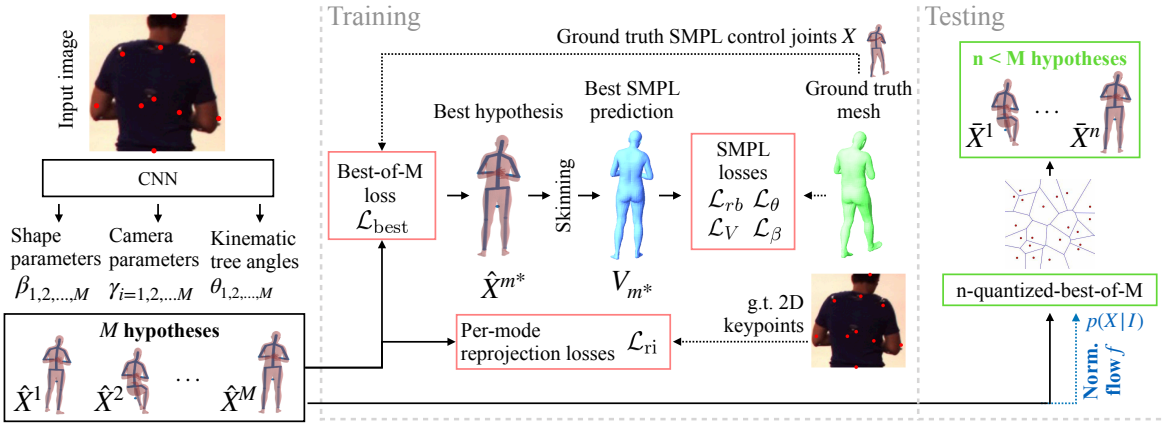


Fig. 5.5 **Overview of our method.** Given a single image of a human, during training, our method produces multiple skeleton hypotheses $\{\hat{X}^i\}_{i=1}^{M}$ that enter a Best-of-M loss which selects the representative $\hat{X}^{m^*}$ which most accurately matches the ground truth control joints X. At test time, we sample an arbitrary number of $n < M$ hypotheses by quantizing the set $\{\hat{X}^i\}$ that is assumed to be sampled from the probability distribution $p(X|I)$ modeled with normalizing flow f.

human body keypoints directly in physical space. Sharma et al. [75] learn a conditional variational auto-encoder to model ambiguous reconstructions as a posterior distribution; they also propose two scoring methods to extract a single 3D reconstruction from the distribution.

Cheng et al. [21] tackle the problem of video 3D reconstruction in the presence of occlusions, and show that temporal cues can be used to disambiguate the solution. While our method is similar in the goal of correctly handling the prediction uncertainty, we differ by applying our method to predicting *full mesh* of the human body. This is arguably a more challenging scenario due to the increased complexity of the desired 3D shape.

Finally, some recent concurrent works also consider building priors over 3D human pose using normalizing flows. Xu et al. [93] release a prior for their new GHUM/GHUML model, and Zanfir et al. [95] build a prior on SMPL joint angles to constrain their weakly-supervised network. Our method differs as we learn our prior on 3D SMPL joints.

5.6 Preliminaries

Before discussing our method, we describe the necessary background, starting from SMPL.

SMPL. SMPL is a model of the human body parameterized by axis-angle rotations $\theta \in \mathbb{R}^{69}$ of 23 body joints, the shape coefficients $\beta \in \mathbb{R}^{10}$ modelling shape variations, and a global

rotation $\gamma \in \mathbb{R}^3$. SMPL defines a *skinning function* $S : (\theta, \beta, \gamma) \mapsto V$ that maps the body parameters to the vertices $V \in \mathbb{R}^{6890 \times 3}$ of a 3D mesh.

Predicting the SMPL parameters from a single image. Given an image **I** containing a person, the goal is to recover the SMPL parameters (θ, β, γ) that provide the best 3D reconstruction of it. Existing algorithms [41] cast this as learning a deep network $G(I) = (\theta, \beta, \gamma, t)$ that predicts the SMPL parameters as well as the translation $t \in \mathbb{R}^3$ of the perspective camera observing the person. We assume a fixed set of camera parameters. During training, the camera is used to constrain the reconstructed 3D mesh and the annotated 2D keypoints to be consistent. Since most datasets only contain annotations for a small set of keypoints ([31] is an exception), and since these keypoints do not correspond directly to any of the SMPL mesh vertices, we need a mechanism to translate between them. This mechanism is a fixed linear regressor $J : V \mapsto X$ that maps the SMPL mesh vertices $V = S(G(I))$ to the 3D locations $X = J(V) = J(S(G(I)))$ of the K joints. Then, the projections $\pi_t(X)$ of the 3D joint positions into image **I** can be compared to the available 2D annotations.

Normalizing flows. The idea of normalizing flows (NF) is to represent a complex distribution $p(X)$ on a random variable X as a much simpler distribution $p(z)$ on a transformed version $z = f(X)$ of X. The transformation f is learned so that $p(z)$ has a fixed shape, usually a Normal $p(z) \sim \mathcal{N}(0, 1)$. Furthermore, f itself must be *invertible* and *smooth*. In this paper, we utilize a particular version of NF dubbed RealNVP [24]. A more detailed explanation of NF and RealNVP has been deferred to the supplementary.

5.7 Method

We start from a neural network architecture that implements the function $G(I) = (\theta, \beta, \gamma, t)$ described above. As shown in SPIN [47], the HMR [41] architecture attains state-of-the-art results for this task, so we use it here. However, the resulting regressor $G(I)$, given an input image I, can only produce a single unique solution. In general, and in particular for cases with a high degree of reconstruction ambiguity, we are interested in predicting *set* of plausible 3D poses rather than a single one. We thus extend our model to explicitly produce a set of M different hypotheses $G_m(I) = (\theta_m, \beta_m, \gamma_m, t_m)$, $m = 1, \ldots, M$. This is easily achieved by modifying the HMR's final output layer to produce a tensor M times larger, effectively stacking the hypotheses. In what follows, we describe the learning scheme that drives the monocular predictor G to achieve an optimal coverage of the plausible poses consistent with the input image. Our method is summarized in fig. 5.5.

5.7.1 Learning with multiple hypotheses

For learning the model, we assume to have a training set of N images $\{I_i\}_{i=1,\dots,N}$, each cropped around a person. Furthermore, for each training image I_i we assume to know (1) the 2D location Y_i of the body joints (2) their 3D location X_i, and (3) the ground truth SMPL fit $(\theta_i, \beta_i, \gamma_i)$. Depending on the set up, some of these quantities can be inferred from the others (e.g. we can use the function J to convert the SMPL parameters to the 3D joints X_i and then the camera projection to obtain Y_i).

Best-of-M loss. Given a single input image, our network predicts a set of poses, where at least one should be similar to the ground truth annotation X_i. This is captured by the best-of-M loss [32]:

$$\mathscr{L}_{\text{best}}(J, G; m^*) = \frac{1}{N}\sum_{i=1}^{N} \left\| X_i - \hat{X}^{m_i^*}(I_i) \right\|, \;\; m_i^* = \underset{m=1,\dots,M}{\operatorname{argmin}} \left\| X_i - \hat{X}^{m}(I_i) \right\|, \tag{5.1}$$

where $\hat{X}^m(I_i) = J(G_m(V(I_i)))$ are the 3D joints estimated by the m-th SMPL predictor $G_m(I_i)$ applied to image I_i. In this way, only the best hypothesis is steered to match the ground truth, leaving the other hypotheses free to sample the space of ambiguous solutions. During the computation of this loss, we also extract the best index m_i^* for each training example.

Limitations of best-of-M. As noted in section 5.4, best-of-M only guarantees that one of the M hypotheses is a good solution, but says nothing about the other ones. Furthermore, in applications we are often interested in modulating the number of hypotheses generated, but the best-of-M regressor $G(I)$ only produces a fixed number of output hypothesis M, and changing M would require retraining from scratch, which is intractable.

We first address these issues by introducing a method that allows us to train a best-of-M model for a large M once and leverage it later to generate an arbitrary number of $n < M$ hypotheses without the need of retraining, while ensuring that these are good representatives of likely body poses.

***n*-quantized-best-of-M** Formally, given a set of M predictions $\hat{\mathscr{X}}^M(I) = \{\hat{X}^1(I), \dots, \hat{X}^M(I)\}$ we seek to generate a smaller n-sized set $\bar{\mathscr{X}}^n(I) = \{\bar{X}^1(I), \dots, \bar{X}^n(I)\}$ which preserves the information contained in $\hat{\mathscr{X}}^M$. In other words, $\bar{\mathscr{X}}^n$ *optimally quantizes* $\hat{\mathscr{X}}^M$. To this end, we interpret the output of the best-of-M model as a set of choices $\hat{\mathscr{X}}^M(I)$ for the possible pose. These poses are of course not all equally likely, but it is difficult to infer their probability from (5.1). We thus work with the following approximation. We consider the prior $p(X)$ on

possible poses (defined in the next section), and set:

$$p(X|I) = p(X|\hat{\mathscr{X}}^M(I)) = \sum_{i=1}^{M} \delta(X - \hat{X}^i(I)) \frac{p(\hat{X}^i(I))}{\sum_{k=1}^{M} p(\hat{X}^k(I))}. \tag{5.2}$$

This amounts to using the best-of-M output as a conditioning *set* (i.e. an unweighted selection of plausible poses) and then use the prior $p(x)$ to weight the samples in this set. With the weighted samples, we can then run K-means [53] to further quantize the best-of-M output while minimizing the quantization energy E:

$$E(\bar{\mathscr{X}}|\hat{\mathscr{X}}) = \mathbb{E}_{p(X|I)} \left[\min_{\{\bar{X}^1,\ldots,\bar{X}^n\}} \|X - \bar{X}^j\|^2 \right] = \sum_{i=1}^{M} \frac{p(\hat{X}^i(I))}{\sum_{k=1}^{M} p(\hat{X}^k(I))} \min_{\{\bar{X}^1,\ldots,\bar{X}^n\}} \|\hat{X}^i(I) - \bar{X}^j\|^2. \tag{5.3}$$

This can be done efficiently on GPU — for our problem, K-Means consumes less than 20% of the execution time of the entire forward pass of our method.

Learning the pose prior with normalizing flows. In order to obtain $p(X)$, we propose to learn a normalizing flow model in form of the RealNVP network f described in section 5.6 and the supplementary. RealNVP optimizes the log likelihood $\mathscr{L}_{\text{nf}}(f)$ of training ground truth 3D skeletons $\{X_1, \ldots X_N\}$ annotated in their corresponding images $\{I_1, \ldots, I_N\}$:

$$\mathscr{L}_{\text{nf}}(f) = -\frac{1}{N} \sum_{i=1}^{N} \log p(X_i) = -\frac{1}{N} \sum_{i=1}^{N} \left(\log \mathscr{N}(f(X_i)) - \sum_{l=1}^{L} \log \left| \frac{df_l(X_{li})}{dX_{li}} \right| \right). \tag{5.4}$$

2D re-projection loss. Since the best-of-M loss optimizes a single prediction at a time, often some members of the ensemble $\hat{\mathscr{X}}(I)$ drift away from the manifold of plausible human body shapes, ultimately becoming 'dead' predictions that are never selected as the best hypothesis m^*. In order to prevent this, we further utilize a re-projection loss that acts across all hypotheses for a given image. More specifically, we constrain the set of 3D reconstructions to lie on projection rays passing through the 2D input keypoints with the following *hypothesis re-projection loss*:

$$\mathscr{L}_{\text{ri}}(J, G) = \frac{1}{N} \sum_{i=1}^{N} \sum_{m=1}^{M} \left\| Y_i - \pi_{t_i}(\hat{X}^m(I)) \right\|. \tag{5.5}$$

Note that many of our training images exhibit significant occlusion, so Y may contain invisible or missing points. We handle this by masking $\mathscr{L}_{\text{ri}}$ to prevent these points contributing to the loss.

Fig. 5.6 **Example samples from the normalizing flow** $f : X \mapsto z;\ p(z) \sim \mathcal{N}(0,1)$, trained on a dataset of ground truth 3D SMPL control skeletons $\{X_1, ..., X_N\}$.

SMPL loss. The final loss terms, introduced by prior work [41, 67, 47], penalize deviations between the predicted and ground truth SMPL parameters. For our method, these are only applied to the best hypothesis m_i^* found above:

$$\mathscr{L}_\theta(G;m^*) = \frac{1}{N}\sum_{i=1}^{N} \|\theta_i - G_{\theta,m_i^*}(I_i)\|;\ \mathscr{L}_V(G;m^*) = \frac{1}{N}\sum_{i=1}^{N} \|S(\theta_i,\beta_i,\gamma_i) - S(G_{(\theta,\beta,\gamma),m_i^*}(I_i))\| \tag{5.6}$$

$$\mathscr{L}_\beta(G;m^*) = \frac{1}{N}\sum_{i=1}^{N} \|\beta_i - G_{\beta,m_i^*}(I_i)\|;\ \mathscr{L}_{\text{rb}}(G;m^*) = \frac{1}{N}\sum_{i=1}^{N} \|Y_i - \pi_{t_i}(\hat{X}^{m_i^*}(I_i))\| \tag{5.7}$$

Note here we use $\mathscr{L}_{\text{rb}}$ to refer to a 2D re-projection error between the best hypothesis and ground truth 2D points Y_i. This differs from the earlier loss $\mathscr{L}_{\text{ri}}$, which is applied across all modes to enforce consistency to the visible *input* points. Note that we could have used eqs. (5.6) and (5.7) to select the best hypothesis m_i^*, but it would entail an unmanageable memory footprint due to the requirement of SMPL-meshing for every hypothesis before the best-of-M selection.

Overall loss. The model is thus trained to minimize:

$$\begin{aligned}\mathscr{L}(J,G) = \lambda_{\text{ri}}\mathscr{L}_{\text{ri}}(J,G) + \lambda_{\text{best}}\mathscr{L}_{\text{best}}(J,G;m^*) + \lambda_\theta\mathscr{L}_\theta(J,G;m^*) \\ + \lambda_\beta\mathscr{L}_\beta(J,G;m^*) + \lambda_{\text{V}}\mathscr{L}_V(J,G;m^*) + \lambda_{\text{rb}}\mathscr{L}_{\text{rb}}(J,G;m^*)\end{aligned} \tag{5.8}$$

where m^* is given in eq. (5.1) and $\lambda_{\text{ri}}, \lambda_{\text{best}}, \lambda_\theta, \lambda_\beta, \lambda_{\text{V}}, \lambda_{\text{rb}}$ are weighing factors. We use a consistent set of SMPL loss weights across all experiments $\lambda_{\text{best}} = 25.0, \lambda_\theta = 1.0, \lambda_\beta = 0.001, \lambda_{\text{V}} = 1.0$, and set $\lambda_{\text{ri}} = 1.0$. Since the training of the normalizing flow f is independent of the rest of the model, we train f separately by optimizing $\mathscr{L}_{\text{nf}}$ with the weight of $\lambda_{\text{nf}} = 1.0$. Samples from our trained normalizing flow are shown in fig. 5.6

5.8 Experiments

In this section we compare our method to several strong baselines. We start by describing the datasets and the baselines, followed by a quantitative and a qualitative evaluation.

Dataset	Quantization n	1		5		10		25	
	Metric	MPJPE	RE	MPJPE	RE	MPJPE	RE	MPJPE	RE
H36M	HMR [40]	—	56.8	—	—	—	—	—	—
	GraphCMR [48]	71.9	50.1	—	—	—	—	—	—
	SPIN [47]	62.2	41.8	—	—	—	—	—	—
	SMPL-MDN	64.4	44.8	61.8	43.3	61.3	43.0	61.1	42.7
	SMPL-CVAE	70.1	46.7	68.9	46.4	68.6	46.3	68.1	46.2
	Ours	**61.5**	**41.6**	**59.8**	**42.0**	**59.2**	**42.2**	**58.2**	**42.2**
3DPW	HMR [40]	—	81.3	—	—	—	—	—	—
	GraphCMR [48]	—	70.2	—	—	—	—	—	—
	SPIN [47]	96.9	**59.3**	—	—	—	—	—	—
	SMPL-MDN	105.8	64.7	96.9	61.2	95.9	60.7	94.9	60.1
	SMPL-CVAE	96.3	61.4	93.7	60.7	92.9	60.5	92.0	60.3
	Ours	**93.8**	59.9	**82.2**	**57.1**	**79.4**	**56.6**	**75.8**	**55.6**
AH36M	SMPL-MDN	113.9	74.7	98.0	70.8	95.1	69.9	91.5	69.5
	SMPL-CVAE	114.5	76.5	111.5	75.7	110.6	75.4	109.7	75.1
	Ours	**103.6**	**67.8**	**96.4**	**67.1**	**93.5**	**66.0**	**90.0**	**64.2**
A3DPW	SMPL-MDN	159.7	82.8	154.6	83.0	149.6	80.7	122.1	76.6
	SMPL-CVAE	156.6	80.2	154.5	79.9	153.9	79.8	153.1	79.8
	Ours	**149.6**	**78.5**	**125.6**	**74.4**	**116.7**	**73.7**	**107.8**	**72.1**

Table 5.1 **Monocular multi-hypothesis human mesh recovery** comparing our approach to two multi-hypothesis baselines (SMPL-CVAE, SMPL-MDN) and state-of-the-art single mode evaluation models [47, 48, 40] on Human3.6m (H36M), its ambiguous version AH36M, on 3DPW and its ambiguous version A3DPW.

Datasets and evaluation protocol. Our evaluation focuses on the Human3.6m (**H36M**) [36, 17] and **3DPW** datasets [87]. H36M is one of the largest datasets of humans annotated with 3D pose using MoCap sensors. As common practice, we train on subjects S1, S5, S6, S7 and S8, and test on S9 and S11. 3DPW is only used for evaluation and, following [48], we evaluate on its test set.

Our evaluation is consistent with [47, 48] - we report two metrics that compare the lifted dense 3D SMPL shape to the ground truth mesh: Mean Per Joint Position Error (**MPJPE**), Reconstruction Error (**RE**). For H36M, all errors are computed using an evaluation scheme known as

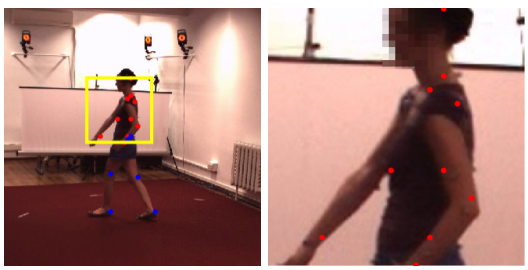


Fig. 5.7 Example image and corresponding annotation from the ambiguous H36M dataset **AH36M**. Best viewed in colour.

Quantization n		5		10		25	
Mode reproj.	Flow weight	MPJPE	RE	MPJPE	RE	MPJPE	RE
		86.4	57.9	84.0	57.5	79.0	56.3
	✓	84.1	**57.0**	81.9	56.7	77.8	55.8
✓		82.7	57.5	79.9	57.0	76.2	55.9
✓	✓	**82.2**	57.1	**79.4**	**56.6**	**75.8**	**55.6**

Table 5.2 **Ablation study on 3DPW** removing either the normalizing flow or the mode re-projection losses and reporting the change in performance.

"Protocol #2". Please refer to supplementary for
a detailed explanation of MPJPE and RE.

Multipose metrics. MPJPE and RE are traditional metrics that assume a single correct ground truth prediction for a given 2D observation. As mentioned above, such an assumption is rarely correct due to the inherent ambiguity of the monocular 3D shape estimation task. We thus also report MPJPE-*n*/RE-*n* an extension of MPJPE RE used in [50], that enables an evaluation of n different shape hypotheses. In more detail, to evaluate an algorithm, we allow it to output n possible predictions and, out of this set, we select the one that minimizes the MPJPE/RE metric. We report results for $n \in \{1, 5, 10, 25\}$.

Ambiguous H36M/3DPW (AH36M/A3DPW). Since H36M is captured in a controlled environment, it rarely depicts challenging real-world scenarios such as body occlusions that are the main source of ambiguity in the single-view 3D shape estimation problem.

Hence, we construct an adapted version of H36M with synthetically-generated occlusions (fig. 5.7) by randomly hiding a subset of the 2D keypoints and re-computing an image crop around the remaining visible joints. Please refer to the supplementary for details of the occlusion generation process.

While 3DPW does contain real scenes, for completeness, we also evaluate on a noisy, and thus more challenging version (A3DPW) generated according to the aforementioned strategy.

Baselines Our method is compared to two multi-pose prediction baselines. For fairness, both baselines extend the same (state-of-the-art) trunk architecture as we use, and all methods have access to the same training data.

SMPL-MDN follows [50] and outputs parameters of a mixture density model over the set of SMPL log-rotation pose parameters. Since a naïve implementation of the MDN model

leads to poor performance ($\approx$ 200mm MPJPE-$n = 5$ on H36M), we introduced several improvements that allow optimization of the total loss eq. (5.8). **SMPL-CVAE**, the second baseline, is a conditional variational autoencoder [81] combined with our trunk network. SMPL-CVAE consists of an encoding network that maps a ground truth SMPL mesh V to a gaussian vector z which is fed together with an encoding of the image to generate a mesh V' such that $V' \approx V$. At test time, we sample n plausible human meshes by drawing $z \sim \mathcal{N}(0,1)$ to evaluate with MPJPE-n/RE-n. More details of both SMPL-CVAE and SMPL-MDN have been deferred to the supplementary material.

For completeness, we also compare to three more baselines that tackle the standard single-mesh prediction problem: HMR [40], GraphCMR [67], and SPIN [47], where the latter currently attain state-of-the-art performance on H36M/3DPW. All methods were trained on H36M [36], MPI-INF-3DHP [60], LSP [38], MPII [6] and COCO [52].

5.8.1 Results

Table 5.1 contains a comprehensive summary of the results on all 3 benchmarks. Our method outperforms the SMPL-CVAE and SMPL-MDN in all metrics on all datasets. For SMPL-CVAE, we found that the encoding network often "cheats" during training by transporting all information about the ground truth, instead of only encoding the modes of ambiguity. The reason for a lower performance of SMPL-MDN is probably the representation of the probability in the space of log-rotations, rather in the space of vertices. Modelling the MDN in the space of model vertices would be more convenient due to being more relevant to the final evaluation metric that aggregates per-vertex errors, however, fitting such high-dimensional (dim=6890×3) Gaussian mixture is prohibitively costly.

Furthermore, it is very encouraging to observe that our method is also able to outperform the single-mode baselines [40, 48, 47] on the single mode MPJPE on both H36M and 3DPW. This comes as a surprise since our method has not been optimized for this mode of operation. The difference is more significant for 3DPW which probably happens because 3DPW is not used for training and, hence, the normalizing flow prior acts as an effective filter of predicted outlier poses. Qualitiative results are shown in fig. 5.8.

Ablation study. We further conduct an ablative study on 3DPW that removes components of our method and measures the incurred change in performance. More specifically, we: 1) ablate the hypothesis reprojection loss; 2) set $p(X|I) = \text{Uniform}$ in eq. (5.3), effectively removing the normalizing flow component and executing unweighted K-Means in n-quantized-best-of-M. Table 5.2 demonstrates that removing both contributions decreases performance, validating our design choices.

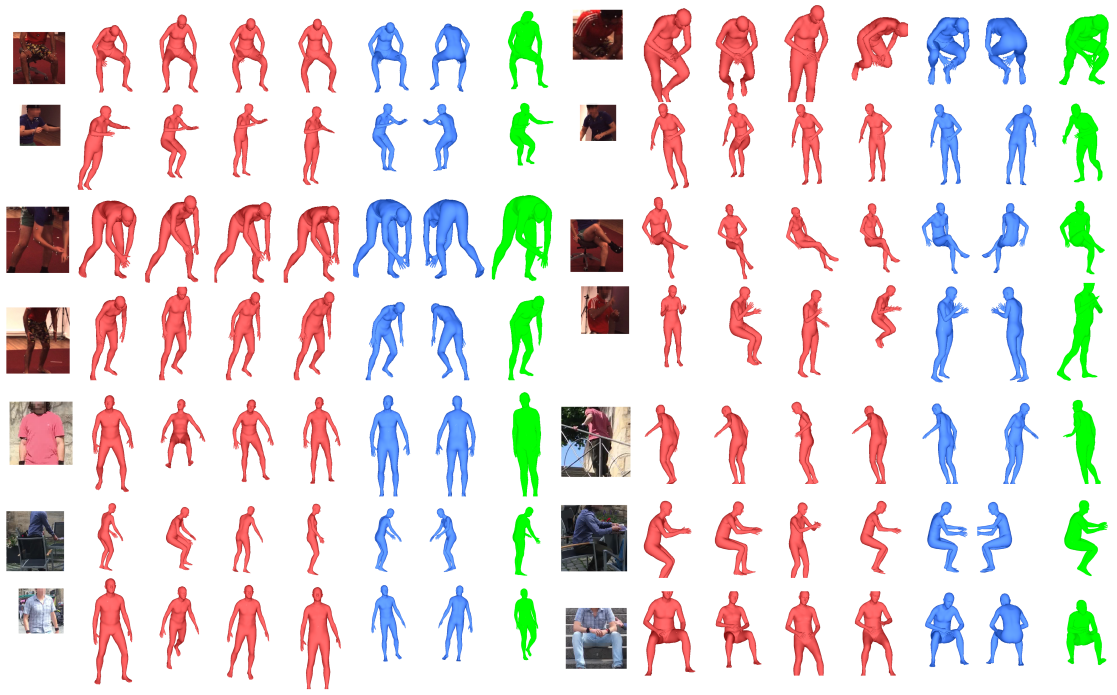


Fig. 5.8 **Qualitative results from $n = 5$ quantization on monocular mesh recovery on AH36m and A3DPW.** From left to right, each group of figures depicts the input ambiguous image, five network hypotheses with the closest to the ground truth in blue, and the ground truth pose in green.

5.9 Conclusions

In this work, we have explored a seldom visited problem of representing the set of plausible 3D meshes corresponding to a single ambiguous input image of a human. To this end, we have proposed a novel method that trains a single multi-hypothesis best-of-M model and, using a novel n-quantized-best-of-M strategy, allows to sample an arbitrary number $n < M$ of hypotheses.

Importantly, this proposed quantization technique leverages a normalizing flow model, that effectively filters out the predicted hypotheses that are unnatural. Empirical evaluation reveals performance superior to several strong probabilistic baselines on Human36M, its challenging ambiguous version, and on 3DPW. Our method encounters occasional failure cases, such as when tested on individuals with unusual shape (e.g. obese people), since we have very few of these examples in the training set. Tackling such cases would make for interesting and worthwhile future work.

Chapter 6

Conclusions

6.1 Discussion and Limitations

In this section I will conclude and discuss limitations

6.1.1 Discussion

Talk about meshes, radiance fields etc.

6.1.2 Applications in Animal Tracking

Discussion as to what GSK have been doing.

6.1.3 Future Work

What needs to happen etc.

References

[1] Adobe Systems Inc. (2018). Creating a green screen key using ultra key. https://helpx.adobe.com/premiere-pro/atv/cs5-cs55-video-tutorials/creating-a-green-screen-key-using-ultra-key.html. Accessed: 2018-03-14.

[2] Agudo, A., Pijoan, M., and Moreno-Noguer, F. (2018). Image collection pop-up: 3d reconstruction and clustering of rigid and non-rigid categories. In *Proc. CVPR*.

[3] Akhter, I. and Black, M. J. (2015). Pose-conditioned joint angle limits for 3D human pose reconstruction. In *Proc. CVPR*.

[4] Alldieck, T., Magnor, M., Bhatnagar, B. L., Theobalt, C., and Pons-Moll, G. (2019). Learning to reconstruct people in clothing from a single rgb camera. In *Proc. CVPR*, pages 1175–1186.

[5] American Pet Products Association (2020). *2019-2020 APPA National Pet Owners Survey*. http://www.americanpetproducts.org.

[6] Andriluka, M., Pishchulin, L., Gehler, P., and Schiele, B. (2014). 2D human pose estimation: New benchmark and state of the art analysis. In *Proc. CVPR*.

[7] Andriluka, M., Roth, S., and Schiele, B. (2010). Monocular 3D pose estimation and tracking by detection. In *Proc. CVPR*, pages 623–630. IEEE.

[8] Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., and Davis, J. (2005). SCAPE: shape completion and animation of people. In *ACM Trans. on Graphics*.

[9] Biggs, B., Roddick, T., Fitzgibbon, A., and Cipolla, R. (2018). Creatures great and SMAL: Recovering the shape and motion of animals from video. In *Proc. ACCV*.

[10] Bishop, C. M. (1994). Mixture density networks. Technical report, Aston University.

[11] Blum, H. (1967). A transformation for extracting new descriptors of shape. *Models for Perception of Speech and Visual Forms, 1967*, pages 362–380.

[12] Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., and Black, M. J. (2016). Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *Proc. ECCV*.

[13] Bulat, A. and Tzimiropoulos, G. (2016). Human pose estimation via convolutional part heatmap regression. In *Proc. ECCV*, pages 717–732. Springer.

[14] Cao, J., Tang, H., Fang, H., Shen, X., Tai, Y., and Lu, C. (2019). Cross-domain adaptation for animal pose estimation. In *Proc. ICCV*, pages 9497–9506.

[15] Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. (2017). Realtime multi-person 2D pose estimation using part affinity fields. In *Proc. CVPR*.

[16] Cashman, T. J. and Fitzgibbon, A. W. (2013). What shape are dolphins? Building 3D morphable models from 2D images. *PAMI*, 35(1):232–244.

[17] Catalin Ionescu, Fuxin Li, C. S. (2011). Latent structured models for human pose estimation. In *Proc. ICCV*.

[18] Chen, L., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation.

[19] Chen, X., Mottaghi, R., Liu, X., Fidler, S., Urtasun, R., and Yuille, A. (2014). Detect what you can: Detecting and representing objects using holistic models and body parts. In *Proc. CVPR*.

[20] Chen, Y., Kim, T.-K., and Cipolla, R. (2010). Inferring 3D shapes and deformations from single views. In *Proc. ECCV*, pages 300–313. Springer.

[21] Cheng, Y., Yang, B., Wang, B., Yan, W., and Tan, R. T. (2019). Occlusion-aware networks for 3d human pose estimation in video. In *Proc. ICCV*.

[22] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). ImageNet: A Large-Scale Hierarchical Image Database. In *Proc. CVPR*.

[23] Diamond, S. and Boyd, S. (2016). CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5.

[24] Dinh, L., Sohl-Dickstein, J., and Bengio, S. (2017). Density estimation using Real NVP. In *Proc. ICLR*.

[25] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338.

[26] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. (2012). The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results.

[27] Faugeras, O. and Luong, Q.-T. (2001). *The Geometry of Multiple Images*. MIT Press.

[28] Favreau, L., Reveret, L., Depraz, C., and Cani, M.-P. (2004). Animal gaits from video. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 277–286.

[29] Food and Agriculture Organization of the United Nations (2016). FAOSTAT statistics database. [Online; data retrieved from FAOSTAT on 21-November-2017].

[30] Guan, P., Weiss, A., Balan, A. O., and Black, M. J. (2009). Estimating human shape and pose from a single image. In *Proc. ICCV*.

[31] Güler, R. A., Neverova, N., and Kokkinos, I. (2018). Densepose: Dense human pose estimation in the wild. In *Proc. CVPR*, pages 7297–7306.

[32] Guzman-Rivera, A., Batra, D., and Kohli, P. (2012). Multiple choice learning: Learning to produce multiple structured outputs. In *Proc. NeurIPS*, pages 1799–1807.

[33] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

[34] Holland, J. H. (1992). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press.

[35] Huang, Y., Bogo, F., Lassner, C., Kanazawa, A., Gehler, P. V., Romero, J., Akhter, I., and Black, M. J. (2017). Towards accurate marker-less human shape and pose estimation over time. In *Proc. 3DV*.

[36] Ionescu, C., Papava, D., Olaru, V., and Sminchisescu, C. (2014). Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *PAMI*, 36(7):1325–1339.

[37] Johnson, S. and Everingham, M. (2010). Clustered pose and nonlinear appearance models for human pose estimation. In *Proc. BMVC*, pages 12.1–12.11.

[38] Johnson, S. and Everingham, M. (2011). Learning effective human pose estimation from inaccurate annotation. In *Proc. CVPR*.

[39] Joo, H., Simon, T., and Sheikh, Y. (2018). Total capture: A 3D deformation model for tracking faces, hands, and bodies. In *Proc. CVPR*.

[40] Kanazawa, A., Black, M. J., Jacobs, D. W., and Malik, J. (2018a). End-to-end recovery of human shape and pose. In *Proc. CVPR*.

[41] Kanazawa, A., Tulsiani, S., Efros, A. A., and Malik, J. (2018b). Learning category-specific mesh reconstruction from image collections. In *Proc. ECCV*.

[42] Kato, H., Ushiku, Y., and Harada, T. (2018). Neural 3d mesh renderer. In *Proc. CVPR*.

[43] Kearney, S., Li, W., Parsons, M., Kim, K. I., and Cosker, D. (2020). Rgbd-dog: Predicting canine pose from rgbd sensors. In *Proc. CVPR*.

[44] Khamis, S., Taylor, J., Shotton, J., Keskin, C., Izadi, S., and Fitzgibbon, A. (2015). Learning an efficient model of hand shape variation from depth images. In *Proc. CVPR*. IEEE.

[45] Khoreva, A., Benenson, R., Ilg, E., Brox, T., and Schiele, B. (2017). Lucid data dreaming for object tracking. *The 2017 DAVIS Challenge on Video Object Segmentation - CVPR Workshops*.

[46] Khosla, A., Jayadevaprakash, N., Yao, B., and Fei-Fei, L. (2011). Novel dataset for fine-grained image categorization. In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO.

[47] Kolotouros, N., Pavlakos, G., Black, M. J., and Daniilidis, K. (2019a). Learning to reconstruct 3D human pose and shape via model-fitting in the loop. In *Proc. ICCV*.

[48] Kolotouros, N., Pavlakos, G., and Daniilidis, K. (2019b). Convolutional mesh regression for single-image human shape reconstruction. In *Proc. CVPR*.

[49] Lassner, C., Romero, J., Kiefel, M., Bogo, F., Black, M. J., and Gehler, P. V. (2017). Unite the people: Closing the loop between 3D and 2D human representations. In *Proc. CVPR*.

[50] Li, C. and Lee, G. H. (2019). Generating multiple hypotheses for 3d human pose estimation with mixture density network. In *Proc. CVPR*.

[51] Li, X., Qi, Y., Wang, Z., Chen, K., Liu, Z., Shi, J., Luo, P., Loy, C. C., and Tang, X. (2017). Video object segmentation with re-identification. *The 2017 DAVIS Challenge on Video Object Segmentation - CVPR Workshops*.

[52] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Proc. ECCV*.

[53] Lloyd, S. (1982). Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137.

[54] Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., and and, M. J. B. (2015a). SMPL: A skinned multi- person linear model. *ACM Trans. on Graphics*.

[55] Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., and Black, M. J. (2015b). SMPL: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):248.

[56] Loper, M. M. and Black, M. J. (2014). OpenDR: An approximate differentiable renderer. In *Proc. ECCV*, pages 154–169. Springer.

[57] Lourakis, M. and Argyros, A. A. (2005). Is Levenberg-Marquardt the most efficient optimization algorithm for implementing bundle adjustment? In *Proc. ICCV*, pages 1526–1531.

[58] Martinez, J., Romero, J., Kiefel, M., Bogo, F., Black, M. J., and Gehler, P. V. (2017). A simple yet effective baseline for 3D human pose estimation. In *Proc. CVPR*.

[59] Mathis, A., Mamidanna, P., Cury, K. M., Abe, T., Murthy, V. N., Mathis, M. W., and Bethge, M. (2018). DeepLabCut: Markerless pose estimation of user-defined body parts with deep learning. Technical report, Nature Publishing Group.

[60] Mehta, D., Rhodin, H., Casas, D., Fua, P., Sotnychenko, O., Xu, W., and Theobalt, C. (2017a). Monocular 3d human pose estimation in the wild using improved cnn supervision. In *Proc. 3DV*. IEEE.

[61] Mehta, D., Sridhar, S., Sotnychenko, O., Rhodin, H., Shafiei, M., Seidel, H.-P., Xu, W., Casas, D., and Theobalt, C. (2017b). VNect: Real-time 3d human pose estimation with a single RGB camera. In *Proc. SIGGRAPH*.

[62] Newell, A., Yang, K., and Deng, J. (2016). Stacked hourglass networks for human pose estimation. In *Proc. ECCV*.

[63] Novotny, D., Ravi, N., Graham, B., Neverova, N., and Vedaldi, A. (2019). C3DPO: Canonical 3d pose networks for non-rigid structure from motion. In *Proc. ICCV*.

[64] Omran, M., Lassner, C., Pons-Moll, G., Gehle, P. V., and Schiele, B. (2018). Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In *Proc. 3DV*.

[65] Park, J. and Boyd, S. (2017). General heuristics for nonconvex quadratically constrained quadratic programming.

[66] Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A. A. A., Tzionas, D., and Black, M. J. (2019). Expressive body capture: 3D hands, face, and body from a single image. In *Proc. CVPR*.

[67] Pavlakos, G., Zhu, L., Zhou, X., and Daniilidis, K. (2018). Learning to estimate 3D human pose and shape from a single color image. In *Proc. CVPR*.

[68] Perazzi, F., Pont-Tuset, J., McWilliams, B., Van Gool, L., Gross, M., and Sorkine-Hornung, A. (2016). A benchmark dataset and evaluation methodology for video object segmentation. In *Proc. CVPR*.

[69] Pishchulin, L., Andriluka, M., Gehler, P., and Schiele, B. (2013). Poselet conditioned pictorial structures. In *Proc. CVPR*, pages 588–595. IEEE.

[70] Probst, T., Pani Paudel, D., Chhatkuli, A., and Van Gool, L. (2018). Incremental non-rigid structure-from-motion with unknown focal length. In *Proc. ECCV*.

[71] Reinert, B., Ritschel, T., and Seidel, H.-P. (2016). Animated 3D creatures from single-view video by skeletal sketching. In *Graphics Interface*, pages 133–141.

[72] Rogez, G., Weinzaepfel, P., and Schmid, C. (2018). LCR-Net++: Multi-person 2D and 3D pose detection in natural images. *PAMI*.

[73] Rupprecht, C., Laina, I., DiPietro, R., Baust, M., Tombari, F., Navab, N., and Hager, G. D. (2017). Learning in an uncertain world: Representing ambiguity through multiple hypotheses. In *Proc. ICCV*.

[74] Saito, S., , Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., and Li, H. (2019). Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. *arXiv preprint arXiv:1905.05172*.

[75] Sharma, S., Varigonda, P. T., Bindal, P., Sharma, A., and Jain, A. (2019). Monocular 3d human pose estimation by generation and ordinal ranking. In *Proc. ICCV*.

[76] Shotton, J., Fitzgibbon, A., Blake, A., Kipman, A., Finocchio, M., Moore, B., and Sharp, T. (2011). Real-time human pose recognition in parts from a single depth image. In *Proc. CVPR*. IEEE.

[77] Sidenbladh, H., Black, M. J., and Fleet, D. J. (2000). Stochastic tracking of 3d human figures using 2d image motion. In *Proc. ECCV*, ECCV '00, page 702–718, Berlin, Heidelberg. Springer-Verlag.

[78] Sigal, L., Balan, A., and Black, M. J. (2008). Combined discriminative and generative articulated pose and non-rigid shape estimation. In *Proc. NeurIPS*.

[79] Sminchisescu, C., Kanaujia, A., Li, Z., and Metaxas, D. (2005). Discriminative density propagation for 3d human motion estimation. volume 1, pages 390– 397 vol. 1.

[80] Sminchisescu, C. and Triggs, B. (2003). Kinematic jump processes for monocular 3d human tracking. In *Proc. CVPR*, CVPR'03, page 69–76, USA. IEEE Computer Society.

[81] Sohn, K., Lee, H., and Yan, X. (2015). Learning structured output representation using deep conditional generative models. In *Proc. NeurIPS*.

[82] Sun, X., Xiao, B., Wei, F., Liang, S., and Wei, Y. (2018). Integral human pose regression. In *Proc. ECCV*.

[83] Tan, V., Budvytis, I., and Cipolla, R. (2017). Indirect deep structured learning for 3D human body shape and pose prediction. In *Proc. BMVC*.

[84] Tung, H.-Y. F., Tung, H.-W., Yumer, E., and Fragkiadaki, K. (2017). Self-supervised learning of motion capture. In *Proc. NeurIPS*.

[85] Varol, G., Ceylan, D., Russel, B., Yang, J., Yumer, E., Laptev, I., and Schmid, C. (2018). BodyNet: Volumetric inference of 3D human body shapes. In *Proc. ECCV*.

[86] Vicente, S. and Agapito, L. (2013). Balloon shapes: Reconstructing and deforming objects with volume from images. In *Proc. 3DV*, pages 223–230.

[87] von Marcard, T., Henschel, R., Black, M., Rosenhahn, B., and Pons-Moll, G. (2018). Recovering accurate 3d human pose in the wild using imus and a moving camera. In *Proc. ECCV*.

[88] Wang, J. and Yuille, A. L. (2015). Semantic part segmentation using compositional model combining shape and appearance. In *Proc. CVPR*, pages 1788–1797.

[89] Wang, P., Shen, X., Lin, Z., Cohen, S., Price, B., and Yuille, A. L. (2015). Joint object and part segmentation using deep learned potentials. In *Proc. ICCV*, pages 1573–1581.

[90] Wiles, O. and Zisserman, A. (2017). Silnet : Single- and multi-view reconstruction by learning from silhouettes. In *Proc. BMVC*.

[91] Wilhelm, N., Vögele, A., Zsoldos, R., Licka, T., Krüger, B., and Bernard, J. (2015). Furyexplorer: visual-interactive exploration of horse motion capture data. In *Visualization and Data Analysis 2015*, volume 9397, page 93970F.

[92] Xiang, D., Joo, H., and Sheikh, Y. (2019). Monocular total capture: Posing face, body, and hands in the wild. In *Proc. CVPR*.

[93] Xu, H., Bazavan, E. G., Zanfir, A., Freeman, W., Sukthankar, R., and Sminchisescu, C. (2020). Ghum & ghuml: Generative 3d human shape and articulated pose models. In *Proc. CVPR*.

[94] Yang, Y. and Ramanan, D. (2013). Articulated human detection with flexible mixtures of parts. *PAMI*, 35(12):2878–2890.

[95] Zanfir, A., Bazavan, E. G., Xu, H., Freeman, W., Sukthankar, R., and Sminchisescu, C. (2020). Weakly supervised 3d human pose and shape reconstruction with normalizing flows. In *Proc. ECCV*.

[96] Zanfir, A., Marinoiu, E., and Sminchisescu, C. (2018). Monocular 3D pose and shape estimation of multiple people in natural scenes — the importance of multiple scene constraints. In *Proc. CVPR*.

[97] Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., and Torralba, A. (2017). Scene parsing through ADE20K dataset. In *Proc. CVPR*.

[98] Zuffi, S., Kanazawa, A., Berger-Wolf, T., and Black, M. J. (2019). Three-d safari: Learning to estimate zebra pose, shape, and texture from images "in the wild". In *Proc. ICCV*.

[99] Zuffi, S., Kanazawa, A., and Black, M. J. (2018). Lions and tigers and bears: Capturing non-rigid, 3D, articulated shape from images. In *Proc. CVPR*.

[100] Zuffi, S., Kanazawa, A., Jacobs, D., and Black, M. J. (2017). 3D menagerie: Modeling the 3D shape and pose of animals. In *Proc. CVPR*, pages 5524–5532. IEEE.