Posebits for Monocular Human Pose Estimation

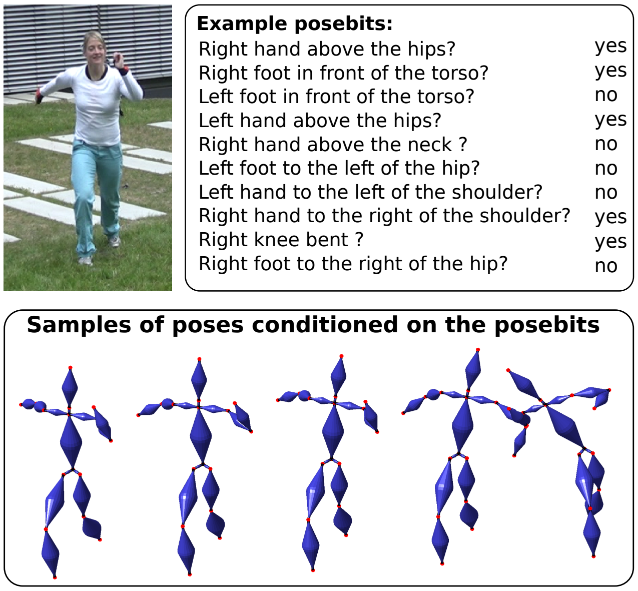
Gerard Pons-Moll
MPI for Intelligent Systems
Tübingen, Germany
gerard.pons.moll@tue.mpg.de

David J. Fleet
University of Toronto
Toronto, Canada
fleet@cs.toronto.edu

Bodo Rosenhahn
Leibniz University of Hannover
Hannover, Germany
rosenhahn@tnt.uni-hannover.de

Abstract

We advocate the inference of qualitative information about 3D human pose, called posebits*, from images. Posebits represent boolean geometric relationships between body parts (*e.g. *left-leg in front of right-leg or hands close to each other). The advantages of posebits as a mid-level representation are 1) for many tasks of interest, such qualitative pose information may be sufficient (*e.g. *semantic image retrieval), 2) it is relatively easy to annotate large image corpora with posebits, as it simply requires answers to yes/no questions; and 3) they help resolve challenging pose ambiguities and therefore facilitate the difficult talk of image-based 3D pose estimation. We introduce posebits, a posebit database, a method for selecting useful posebits for pose estimation and a structural SVM model for posebit inference. Experiments show the use of posebits for semantic image retrieval and for improving 3D pose estimation.*



Example posebits:	
Right hand above the hips?	yes
Right foot in front of the torso?	yes
Left foot in front of the torso?	no
Left hand above the hips?	yes
Right hand above the neck ?	no
Left foot to the left of the hip?	no
Left hand to the left of the shoulder?	no
Right hand to the right of the shoulder?	yes
Right knee bent ?	yes
Right foot to the right of the hip?	no

Figure 1. Top: Posebits are inferred directly from image features using a trained classifier. Since posebits consist of simple yes/no questions, images can be easily annotated by humans. They may be useful for many tasks. For example, on the bottom image, we show samples of poses conditioned on the posebits depicted on the left image. By conditioning the poses on posebits uncertainty about the pose is reduced. Notice how the poses are qualitatively very similar to the observed image. In this example we show the ground truth posebyte. Our model also takes into account uncertainty in the estimation of posebits by marginalizing over them.

1. Introduction

While tremendous effort has focused on the extraction of quantitative 3D human pose from images and video, in this paper we consider the estimation of qualitative pose information, called *posebits*. Posebits are attributes of pose that specify the relative positions or orientations of body parts. They have the advantage that one can easily collect posebit image annotations for training purposes, and they can be reliably inferred from images. Further, they are useful for resolving 3D pose ambiguities and for myriad other tasks where quantitative pose is not required.

The effective use of both generative and discriminative approaches to pose estimation require training data comprising image features and corresponding 3D poses (*e.g.*, see [16, 3, 12, 19, 21]). In practice these data are difficult to obtain, either from high-fidelity commercial marker-based MoCap (Motion Capture) systems [20] or from RGB-D systems such as Microsoft Kinect. They are limited primarily to indoor lab environments, and require significant data curation. Manual annotation of 3D pose from images is not an effective alternative as it is very time-consuming and prone to errors [4, 5].

By contrast, it is relatively simple to obtain training data for posebits from human annotations. Indeed, people often perceive and express pose in terms of the relative positions between body parts (see Fig.), rather than absolute 3D position or joint angle representations that are commonly used in pose estimation tasks. For example, common human verbalizations of pose are: *the left leg in front of right leg, left hand in front of the torso, etc*. It is therefore quite natural to explore the design, inference and use of mid-level, qualita-

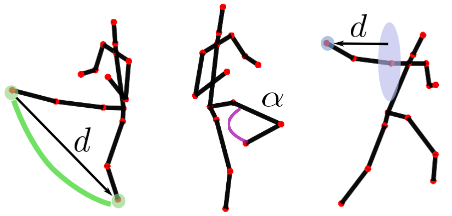


Figure 2. Posebits are qualitative attributes of 3D pose, inferred directly from images. Here, we consider three posebit types, namely, 1) joints distance, 2) articulation angle and 3) relative position. For example, the posebit of type 3 (right) checks whether the right hand is on the right of the shoulder. The plane is centered at the shoulder and the normal direction is the vector going from the chest to the shoulder joint.

tive pose representations.

To this end, we have constructed a preliminary database of images with posebit annotations. Given a task, inspired by work on feature selection and decision trees we also provide a method for selecting useful subsets of posebits. This is accomplished with an algorithm that greedily selects posebits from a pool of candidates by maximizing information gain. Given a set of posebits, called a *posebyte*, we formulate the image classification problem in terms of a structural SVM with loss-based slack rescaling [24]. This exploits both the correlation among pose classes, as well as co-occurrences of posebits. Finally, we show that the inference of posebits facilitates 3D pose inference by reducing uncertainty that stems from pose ambiguities. In particular, it has proven very difficult to fully exploit all available image information when learning mappings from images to full body 3D pose. We find that subtle properties of image appearance can be leveraged more effectively by concentrating on mappings from image features to posebits, thereby helping to reduce pose ambiguity in subsequent pose estimation.

2. Related Work

Most recent work on estimating 3D human pose has focused on the estimation of skeletal joint angles or 3D positions. Faced with measurement noise, missing data, and ambiguity, extensive use of 3D pose data has been common, either to learn generative pose priors, or discriminative mappings from image features to 3D pose (*e.g.* see [3, 12, 13, 18]). Posebits also require labeled training data, but unlike 3D pose, it is easy to obtain posebit annotations. We also make use of MoCap, but without the need for synchronized image and MoCap data.

There also exist geometric approaches to lifting 2D pose to 3D by enumerating 3D poses [10, 22] or by exploiting priors from MoCap dataset [17]. Our work is partially related to [2, 7, 23] in that we also define an intermediate pose representation. However, posebits are more flexible than action classes because they are compositional.

Posebits can also be viewed as attributes of pose. Advantages of attributes have been demonstrated for object categorization [8, 25], and human action recognition [11, 28] with emphasis on transfer learning between classes. Pose attributes have been used for content-based MoCap retrieval [14]. Attributes have also been used to retrieve action-specific priors to stabilize tracking [2]. But none of these approaches infer pose attributes directly from images.

Finally, our work is inspired by work on *poselets* [5], a new notion of parts, which do not necessarily correspond to intuitive body parts (*e.g.* as in [1, 27]). It is argued that the detection of configurations of body is often easier than single parts. However, whereas poselets have shown good performance for people detection, here we focus on estimating the 3D pose information from single images. Unlike poselets, we do not require 3D annotations for training.

3. Posebits

Posebits represent binary, geometric relationships between body parts. They may be useful for myriad tasks, in and of themselves, or as an intermediate representation, *e.g.*, toward 3D pose estimation.

Broadly speaking, we consider three types of posebits which appear relevant to 3D pose inference. But we do not rule out other types that might be relevant to other tasks. The three types, depicted in Fig. 2, are:

1. *Joints distance:* Posebits are activated when two joints in the body are closer or further than a given threshold.
2. *Articulation angle:* Posebits are activated when a given joint angle is bent more than α degrees.
3. *Relative position:* Posebits are activated when a body part A is to the left, right, above, below, in front or behind relative to a second body part B. To determine such posebits, the signed distance between body part A and a plane centered at body part B is computed.

Further, while one might identify hundreds of useful specific posebits, initial exploration of the concept focused only on a relatively small set of 30 candidates, chosen at random from among the three types listed above.

3.1. Posebits database

For selecting, learning and inferring posebits, we exploit a MoCap corpus and a collection of annotated images, which we call the *Posebit Database (PbDb)*. As discussed above, to date we have only annotated images with 30 posebits, but ideally one might want to have annotations with many more than 30 posebits.

At present, PbDb comprises 1) a *MoCap* database comprising 10000 poses taken from Human-Eva [20] and

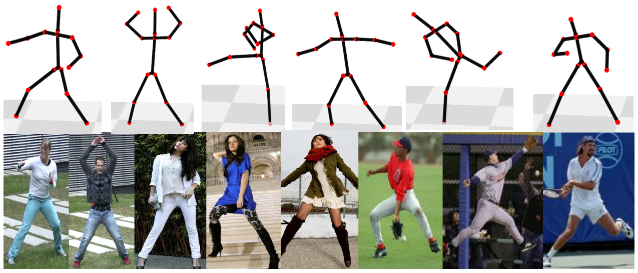


Figure 3. Semantic queries: We queried PbDb with 3 posebits: *left/right foot to the left/right of the hip*, *left foot far from the right foot*. Top row shows several poses retrieved from MoCap DB. Bottom row shows images retrieved from Image DB. Notice how poses are qualitatively similar based on as few as 3 posebits.

HMODB [15], and 2) a set of 4000 images, each annotated with 30 posebits. Images were collected from four publicly available databases. There are 1500 images from Human-Eva [20], 1500 images from HMODB [15], 685 images from Fashion [26] and 305 from Parse [27].

Human-Eva and HMODB come with 3D pose annotations, so it is trivial to compute the corresponding posebits using simple geometric tests, such as point to point, or point to plane, distances, or by thresholding joint angles. Fashion and Parse images do not have 3D pose annotations. However it is straightforward to obtain posebit annotations using Amazon Mechanical Turk, where turkers simply answer yes/no questions about each image. Indeed, based our initial data collection this is an effective way to gather annotations for a much larger image corpus and for many more than 30 posebits. The PbDb image dataset is split into two subsets of 1995 images for training and testing. Fig. 3 shows the result of querying PdDb with a small subset of posebits to obtain semantically similar images.

3.2. Selection

Posebits may be effective in different ways. They may be sufficient for some tasks directly. Or they may be useful as an intermediate representation. Here we focus on their use as a mid-level encoding to facilitate 3D pose inference. It is also clear that different posebits may be useful for different tasks, or redundant. Hence choosing a good set of posebits is essential. To this end we advocate the use of a simple selection mechanism, inspired by decision trees, to choose subsets of posebits from PbDb.

For a given task (*e.g.*, 3D pose estimation), we aim to select a subset of posebits $\mathcal{S}_m$ from a pool of candidates $\mathcal{S}_C$ (*i.e.* PbDb). To this end we use two criteria: Useful posebits are those that can be *reliably* inferred from image features $\mathbf{r}$, and that help reduce uncertainty in the hidden variable of interest, $\mathbf{x}$. Selection makes use of small set of training pairs of image features and 3D poses $\mathcal{L} = \{\mathbf{r}_i, \mathbf{x}_i\}_{i=1}^{L}$, and a larger set of 3D poses $\mathcal{U} = \{\mathbf{x}_j\}_{p=1}^{P}$.

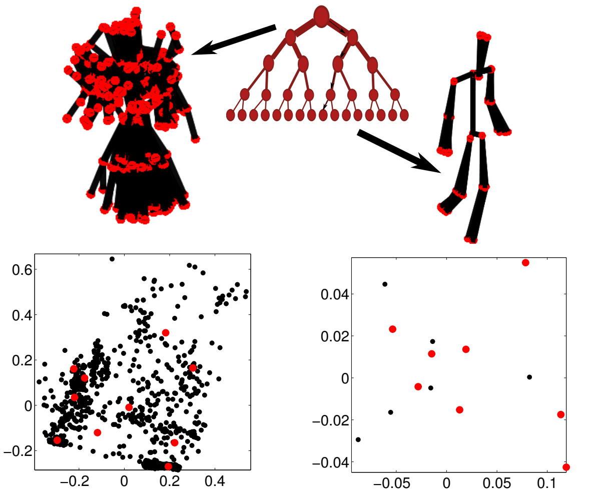


Figure 4. Posebit Binary Tree: Each leaf node contains poses constrained by all posebits in a posebyte. Nodes higher in the tree are constrained by fewer posebits, and hence have greater pose variance. The top left figure depicts poses drawn from the pose distribution conditions on a single posebit. The top right shows the distribution conditioned on 10 posebits. The bottom two plots show the poses at the same nodes projected onto their top two principle directions (*i.e.*, using PCA). Here, red dots depict k-medoids centers, and black dots denote the remaining samples. The variance reduction as one moves down the tree is evident.

To make the problem tractable, we select posebits greedily, one bit at a time, using a forward selection mechanism. Once selected, each posebit partitions the data in two. A set of m posebits yields 2^m posebytes, corresponding to the leaves of a balanced binary tree. At step j, we choose posebit a to maximize *information gain*:

$$a^* = \underset{a \in \mathcal{S}_M}{\arg\max} \quad I_j = I_j^{\mathcal{C}} + \mu \cdot I_j^{\mathcal{R}}, \tag{1}$$

where I_j is the mixed information gain at the *j-th* level of the tree. It comprises a reliability term, $I_j^{\mathcal{R}}$, and a clustering term, $I_j^{\mathcal{C}}$. The parameter μ balances the influence of the two terms ($\mu = 0.5$ in all experiments below).

3.2.1 Clustering

To reduce pose ambiguity, we want posebits that minimize the entropy of $p(\mathbf{x}|\mathbf{a})$, *i.e.*, at each leaf of the binary tree. Thus, when adding the j^{th} posebit, the clustering information gain is computed as $I_j^{\mathcal{C}} = H_{j-1} - H_j$, where H_j is a weighted sum of entropies at each node of the *j-th* level of the tree:

$$H_j = \sum_{c=1}^{2^j} \frac{|\mathcal{S}_c^{\mathbf{x}}|}{|\mathcal{S}^{\mathbf{x}}|} H(\mathcal{S}_c^{\mathbf{x}}) \, . \tag{2}$$

Here, $\mathcal{S}^{\mathbf{x}} = \mathcal{U}$ is the set of MoCap poses, $\mathcal{S}_c^{\mathbf{x}} \subseteq (\mathcal{U})$ is the subset of poses $\mathbf{x}_p$ in posebyte class c, and $H(\mathcal{S})$ is the differential entropy of the pose density for the cluster.

Entropy is difficult estimate with high-dimensional data, so we use the cluster variance as a surrogate for entropy. While a crude assumption, variance provides a measure of cluster compactness and works well in practice. Fig. 4 shows how the conditional pose distribution becomes more concentrated as one travels down from the root to the leaves.

3.2.2 Reliability

A good posebit should also be inferred reliably from image features, and provide as much information about pose as possible. As a simple measure of the extent to which they provide information about pose we consider a information measure in which posebits constitute the only intermediate information available from which one can infer pose.[1]

In more detail, let $\mathbf{x} \in \mathcal{X}^D$ be a target variable, such as 3D pose, let $\mathbf{r} \in \mathcal{R}^d$ denote image features, and let a vector of m posebits, called a posebyte, be denoted by $\mathbf{a} = (a_1, \ldots, a_m) \in \mathcal{A}^m$, where $\mathcal{A} = \{-1, 1\}$. Marginalizing over the posebytes, and supposing that all information about pose is mediated by the posebytes, we consider an approximation to the posterior $p(\mathbf{x}|\mathbf{r})$, *i.e.*,

$$Q(\mathbf{x}|\mathbf{r}, m) \;=\; \sum_{\mathbf{a}\in\mathcal{A}^m} p(\mathbf{x}|\mathbf{a})\, p(\mathbf{a}|\mathbf{r}) \;. \tag{3}$$

Here, $p(\mathbf{x}|\mathbf{a})$ is the conditional pose distribution, and $p(\mathbf{a}|\mathbf{r})$ is posterior posebyte distribution.

When posebits are reliably inferred from the image features, we expect $Q(\mathbf{x}|\mathbf{r})$ to approach the ideal case in which the ground truth posebyte is known, *i.e.*, $Q^{\text{opt}}(\mathbf{x}|\mathbf{r}) = p(\mathbf{x}|\mathbf{a}_{\text{gt}})$. To this end, we express the reliability information gain in terms of the average KL-divergence between $Q(\mathbf{x}|\mathbf{r}, j)$ and $Q^{\text{opt}}(\mathbf{x}|\mathbf{r}, j)$ at level j of the binary tree. The reliability information gain for adding a posebit to the set is defined as $I^{\mathcal{C}} = D^{j-1}_{\text{KL}} - D^{j}_{\text{KL}}$, where D_{KL} the discrete KL-divergence based on training pairs in the labeled set $\mathcal{L} = \{\mathbf{r}_i, \mathbf{a}_i, \mathbf{x}_i\}_{i=1}^{L}$

$$D^{j}_{\text{KL}} \;=\; \sum_i Q^{\text{opt}}(\mathbf{x}_i|\mathbf{r}_i, j) \log\left(\frac{Q^{\text{opt}}(\mathbf{x}_i|\mathbf{r}_i, j)}{Q(\mathbf{x}_i|\mathbf{r}_i, j)}\right). \tag{4}$$

Note that although $Q^{\text{opt}}(\mathbf{x}|\mathbf{r})$ is harder to approximate as we go down the tree, the information gain increases as $Q^{\text{opt}}(\mathbf{x}|\mathbf{r})$ carries much more information about the pose.

In what follows we consider simple ways to approximate the two key elements of Q that are required for this selection criterion, namely, $p(\mathbf{x}|\mathbf{a})$ and $p(\mathbf{a}|\mathbf{r})$.

Modeling $p(\mathbf{a}|\mathbf{r})$: We model $p(\mathbf{a}|\mathbf{r})$ in terms of the score functions for discriminative classifiers for each posebit in PbDb[2]. To each posebit we train an SVM classifier with a linear score function, *e.g.* $F_j(\mathbf{r})$ for posebit j. The score function for a given posebyte, denoted $\hat{G}(\mathbf{a}, \mathbf{r})$, is then used to define the density for $p(\mathbf{a}|\mathbf{r})$.

In particular, for computational convenience, avoiding summations over 2^m posebytes (*e.g.* in (3)), we consider only the top N ranked posebytes [3], $\mathbf{a}_n$, $n = (1 \ldots N)$, thereby approximating $p(\mathbf{a}|\mathbf{r})$ as a multinomial distribution $p(\mathbf{a}|\mathbf{r}) = \sum_n^N \pi_{\mathbf{a}_n} \delta(\mathbf{a} - \mathbf{a}_n)$ where $\pi_{\mathbf{a}_n}$ are computed from the scores using soft-max

$$\pi_{\mathbf{a}_n} = \frac{\exp\left(\hat{G}(\mathbf{a}_n, \mathbf{r})/\tau\right)}{\sum_{s=1}^{N} \exp\left(\hat{G}(\mathbf{a}_s, \mathbf{r})/\tau\right)}, \tag{5}$$

where τ is a temperature parameter set to 0.5. In this setting, Eq. (3) becomes a class conditional mixture model with weights proportional to the posebyte probabilities

$$Q(\mathbf{x}|\mathbf{r}) = \sum_{\mathbf{a}\in\mathcal{A}^m} p(\mathbf{x}|\mathbf{a}) p(\mathbf{a}|\mathbf{r}) \approx \sum_{n=1}^{N} \pi_{\mathbf{a}_n} p(\mathbf{x}|\mathbf{a}_n), \tag{6}$$

And therefore the KL-divergence in Eq.(4) above becomes

$$D^{j}_{\text{KL}} \approx \sum_i p(\mathbf{x}_i|\mathbf{a}_i) \log\left(\frac{p(\mathbf{x}_i|\mathbf{a}_i)}{\sum_{n=1}^{N} \pi_{\mathbf{a}_n} p(\mathbf{x}_i|\mathbf{a}_n)}\right). \tag{7}$$

Modeling $p(\mathbf{x}|\mathbf{a})$: To model $p(\mathbf{x}|\mathbf{a})$ we bin the poses in a MoCap database in 2^m classes, one for each posebyte. Recall that, given a pose $\mathbf{x}$ its corresponding posebyte description is easily obtained with simple geometric computations.

We then represent each class distribution by computing k-medoids obtaining K representatives $\{\mathbf{x}_{k,\mathbf{a}_n}\}_{k=1}^{K}$ for class $\mathbf{a}_n$. To avoid unwanted bias we assume the K poses are equally probable, *i.e.*, $p(\mathbf{x}|\mathbf{a}_n) = \sum_{k=1}^{K} \frac{1}{K} \delta(\mathbf{x} - \mathbf{x}_{k,\mathbf{a}_n})$. In this way, we always sample a fixed set of K codeposes per class, see Fig. 7.

With this approximation to to $p(\mathbf{x}|\mathbf{a})$, and the multinomial approximation to $p(\mathbf{a}|\mathbf{r})$ we obtain a model for Q in terms of a weighted set of samples; *i.e.*, $Q(\mathbf{x}|\mathbf{r}) = \{w_{k,\mathbf{a}_n}, \mathbf{x}_{k,\mathbf{a}_n}\}$, with $k \in \{1 \ldots K\}$ and $n \in \{1 \ldots N\}$, with weights $w_{k,\mathbf{a_n}} = \frac{1}{K} \pi_{\mathbf{a}_n}$.

3.2.3 Selection Experiments

To demonstrate the efficacy of our selection method we compare the influence of the selected posebits on the performance of our pose estimation algorithm (Sec. 4.1). Our

[1] Alternatively, we also found good results using a measure of classification performance but this ignores the effect a miss-classified posebyte has on the final estimation of the hidden variable $\mathbf{x}$.

[2] Ideally we would use score functions for posebyte classifiers by there are 2^m posebytes, and hence we approximate these score functions by the sum of the m single-posebit score functions. This allow us to evaluate the information gain having to train a new structural SVM every time a new posebit is added to the set.

[3] we use $N = 4$ in all experiments.

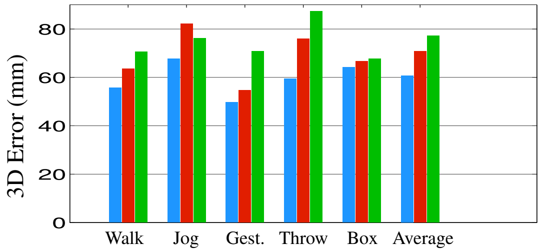


Figure 5. **Posebit selection.** Results on Human-Eva data. (red) 3D error using 10 selected posebits; (green) average error of 20 different random subsets of 10 pose bits; (blue) accuracy obtained using the ground truth for the 10 selected posebits (to indicate the impact of posebyte classification errors).

algorithm uses the general model in Eq. 3 to generate pose proposals. For this experiment we run the algorithm on the sequences in Human-Eva. We first use 20 different random subsets of 10 posebits, picked from the pool of 30 in PbDb, and report the average 3D pose error (green bars in Fig. 5). We also show the top 10 posebits (red bars in Fig. 5). The selected set improves performance significantly on average which demonstrates that the selection provides a good set of posebits for inference. The selected posebits are not necessarily the ones with better classification scores as those might be uninformative or redundant with others. This is shown in the supplemental material where we report the accuracy for each of the 30 posebit candidates.

3.3. Posebits Classifier

The set of m posebits generated and selected by our method (Sec. 3.2) form the posebyte $\mathbf{a} = (a_1, \dots a_m) \in \mathcal{A}^m$. We infer the posebyte directly from raw image features $\mathbf{r} \in \mathcal{R}^d$ using a model based on structural SVM [24]. For learning we only require an image dataset with posebit labels $\mathcal{I} = \{\mathbf{r}_i, \mathbf{a}_i\}_{i=1}^M$.

With a structural SVM the discriminant function for a single posebit $F : \mathcal{R}^d \times \mathcal{A} \mapsto \mathbb{R}$ provides a score for the values the posebit can take. The i-th posebit would then be estimated by maximizing this function

$$\hat{a}_j = \arg\max_{a_j \in \mathcal{A}} \; F(\mathbf{r}, a_j, \mathbf{w}_{a_j}) = \mathbf{w}_{a_j}^T \phi(a_j, \mathbf{r}) \qquad (8)$$

where $\phi(\mathbf{r}, a_j) = a_j \, \mathbf{r}$ is the joint feature map of input $\mathbf{r}$ and output a_j, and $\mathbf{w}_{a_j}^T$ is the vector of weights to be learned.

While such score functions for a single posebit were used above for selecting good posebytes, they do not exploit the shared information among posebyte classes, *i.e.*, classes with similar posebit strings will be semantically more similar in pose space. To this end, we learn a discriminant function $G : \mathcal{R}^d \times \mathcal{A}^m \mapsto \mathbb{R}$ over input output pairs from which we can derive prediction by maximizing over the response variable $\mathbf{a}$ for a given input $\mathbf{r}$. The joint SVM scoring function is expressed as

$$\hat{\mathbf{a}} = \arg\max_{\mathbf{a} \in \mathcal{A}^m} \; G(\mathbf{r}, \mathbf{a}, \beta_{\mathbf{a}}) = \mathbf{a}^T \mathbf{B} \, \mathbf{r} + \mathbf{b}^T \psi(\mathbf{a}) \qquad (9)$$

where $\beta_{\mathbf{a}} = [\mathbf{B}(:) \; \mathbf{b}]$ is the vector of all weights to be learned, $\mathbf{B} \in \mathbb{R}^{d \times m}$ is a matrix the rows of which define separating hyperplanes for the posebits, $\psi(\mathbf{a})$ is a potential that captures posebit co-occurrences. For efficiency and to prevent over-fitting we factorize the prior $\psi(\mathbf{a})$ in pair-wise terms, so Eq. (9) becomes

$$G(\mathbf{r}, \mathbf{a}, \beta_{\mathbf{a}}) = \sum_j^m \mathbf{b}_{a_j}^T \phi(\mathbf{r}, a_j) + \sum_j \sum_k b_{a_j, a_k} \psi(a_j, a_k) \qquad (10)$$

where $\mathbf{b}_{a_j}^T$ is the i-th row of $\mathbf{B}$ providing the score of posebit a_j and $\psi(a_j, a_k)$ is the pairwise potential capturing co-occurrences. The pair-wise potentials consist of normalized histograms learned from MoCap data. Since the prior only depends on the output variable it can be precomputed resulting in substantial computational savings.

Equation (10) can be written as a scalar product $G(\mathbf{r}, \mathbf{a}, \beta_{\mathbf{a}}) = \langle \beta_{\mathbf{a}}, \Phi(\mathbf{r}, \mathbf{a}) \rangle$ between the vector of weights $\beta_{\mathbf{a}}$ and the joint feature map $\Phi(\mathbf{r}, \mathbf{a})$. Having zero training error means that the model scores better the correct posebytes than any other posebyte. Learning the weights $\beta_{\mathbf{a}}$ involves solving the quadratic optimization problem:

$$\begin{aligned} &\min_{\beta_{\mathbf{a}}, \xi} \; \frac{1}{2} \|\beta_{\mathbf{a}}\|^2 + \frac{C}{M} \sum_{i=1}^M \xi_i \\ &s.t \quad \forall i, \quad \forall \mathbf{a} \in \mathcal{A}^m \backslash \mathbf{a}_i, \; \xi_i > 0 \\ &\langle \beta_{\mathbf{a}}, \Phi(\mathbf{r}_i, \mathbf{a}_i) - \Phi(\mathbf{r}_i, \mathbf{a}) \rangle \geq 1 - \frac{\xi_i}{\Delta(\mathbf{a}_i, \mathbf{a})} \end{aligned}$$

The above constraint states that the true output $\mathbf{a}_i$ should score at least a unit better (the margin) than the best runner-up. The objective function penalizes violations of these constraints using scaled slack variables ξ_i. Intuitively, violation of a margin constraint associated with a high loss $\Delta(\mathbf{a}_i, \mathbf{a})$ is penalized severely. We do this by scaling the slack variables with the inverse loss $\Delta(\mathbf{a}_i, \mathbf{a})$. The loss is simply the Hamming distance between posebytes $\mathbf{a}_i$ and $\mathbf{a}$.

3.3.1 Classification Experiments

To learn the model in Eq. (10), we used the training images of the annotated image set, $\mathcal{I} = \{\mathbf{r}_i, \mathbf{a}_i\}_{i=1}^M$. Assuming a bounding box of the person, we construct the feature vector $\mathbf{r}$ by computing spatial pyramid features [9], which are spatially localized HOG (Histogram of Oriented Gradients) over increasing cells of sizes 8, 16, 32 and 64 pixels. Histograming over larger windows adds robustness to miss-alignments in the training data.

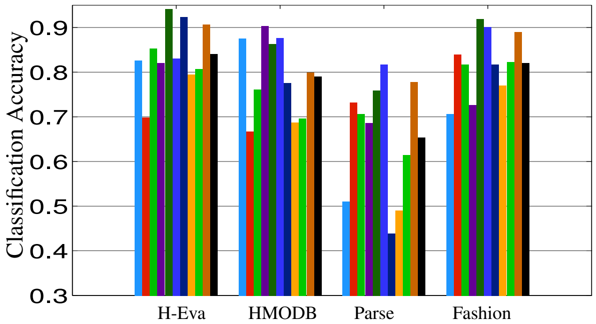


Figure 6. **Classification accuracies** for top 10 posebits selected by our algorithm, when applied to test images from each of the four databases. Colored bars correspond to individual posebits. For instance the left-most two bars (blue and red) correspond to posebits *Right hand above the hips?* and *Right foot in front of the torso?* respectively. The black right-most bar is the average accuracy over the 10 posebits. Performance is very good for HumanEva, HMODB and Fashion. On the Parse datasets some of the posebits are not reliably detected due to the high variability in the poses seen in the images. For example the posebit *Left hand to the left of the torso?* is not reliably estimated for Parse. This might be due to a bias in our dataset, *i.e.*, we do not have enough positive examples for that posebit. Other posebits such as *Right hand above the neck ?* (fifth bar from the left) is accurately classified in all datasets. Note however, that it is selected fifth because other posebits were deemed more informative, despite having lower test accuracies.

Figure 6 depicts the classification accuracies in the test sets of the four datasets, H-Eva, HMODB, Fashion and Parse, *i.e.*, the fraction of test images for which the classifier was correct for a given posebit. Our model can predict posebits from images with remarkably high accuracies (70-90). The dataset where we perform more modestly is Parse. That is probably due to the high variability in pose and appearance and due to the fact that we only use 150 images for training (one order than magnitude less than for the other datasets). Since there is more redundancy in H-Eva and HMODB, better accuracies can be obtained. Notably, we obtain good accuracies across datasets even though a single model was trained using a joint dataset as opposed to training separate models.

4. Experiments

Here we report two more experiments with the use of posebits, one with monocular 3D pose estimation and one with pose-based image retrieval.

4.1. 3D Pose Estimation

We first consider the use of posebits for 3D pose estimation. The goal is to demonstrate a reduction in pose ambiguity that stems from the use of posebits. To that end, we use the $Q(\mathbf{x}|\mathbf{r})$ in Eq. (3) as a proposal distribution during inference. However, unlike it's construction in the posebit selection algorithm, here we use the m-bit structural SVM, rather than single posebit classifiers, to obtain the classifier scores $G(\mathbf{r}, \mathbf{a})$ (see Sec.3.3).

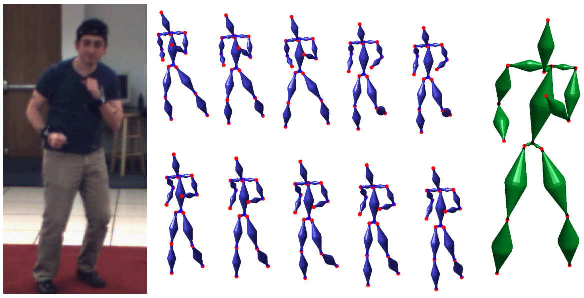


Figure 7. Poses sampled from $Q(\mathbf{x}|\mathbf{r})$ are evaluated top-down. We show the input image on the left, the pose proposals from $Q(\mathbf{x}|\mathbf{r})$ in the center (blue poses), and the inferred 3D pose (in green pose) on the right. In the center, each row corresponds to pose representatives $\mathbf{x}_{k,\mathbf{a}_n}$ for the top 2 ranked posebyte classes $\mathbf{a}_n$. As it can be seen in the poses in the middle uncertainty is reduced thanks to the information about pose mediated by posebytes.

To demonstrate a reduction in uncertainty we use a simple top-down generative model that uses of $Q(\mathbf{x}|\mathbf{r})$ to generate pose hypotheses. Given some image features $\mathbf{z}$ for pose estimation, we express the posterior as

$$p(\mathbf{x}|\mathbf{z}, \mathbf{r}) \propto p(\mathbf{z}|\mathbf{x}, \mathbf{r})p(\mathbf{x}|\mathbf{r}) \simeq p(\mathbf{z}|\mathbf{x})Q(\mathbf{x}|\mathbf{r}) \qquad (11)$$

where $\mathbf{z}$ and $\mathbf{r}$ are assumed to be conditionally independent given the pose $\mathbf{x}$.

Image Likelihood: Many research papers have focused on the design of high-fidelity likelihood models, such as [6], but while the likelihood is a key ingredient in pose estimation, it is not the primary focus of our work. Instead, here we assume that unlabeled 2D joint locations are available, perhaps obtained from a 2D pose estimation algorithm. Hence, the image features $\mathbf{z} = (\mathbf{m}_1 \dots \mathbf{m}_J)$ consist of a collection of 2D points $\mathbf{m}_i \in \mathbb{R}^2$.

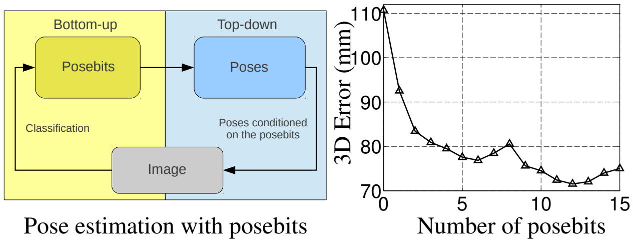
Let $\mathcal{F}(\mathbf{x}; j) : \mathcal{X}^D \mapsto \mathbb{R}^3$ be a function that maps a pose $\mathbf{x}$ to the j-th 3D joint position. We model $p(\mathbf{z}|\mathbf{x})$ as a product of isotropic 2D Gaussians centered at joint locations:

$$p(\mathbf{z}|\mathbf{x}) = \frac{1}{C} \exp\left(-\sum_{i=1}^{P} e^2(\mathbf{m}_i|\mathbf{x}) \right) , \qquad (12)$$

where C is a normalization constant, and $e(\mathbf{m}_i|\mathbf{x})$ is the Euclidean distance between the 2D measurement and the closest 3D joint projected into the image:

$$e(\mathbf{m}_i|\mathbf{x}) = \min_j \|\mathbf{m}_i - \mathrm{Proj}(\mathcal{F}(\mathbf{x}; j))\| . \qquad (13)$$

Here, Proj projects 3D points to the image plane. We a scaled orthographic projection, where the scale is set to match the person's height in the image plane.



Pose estimation with posebits

Figure 8. **Pose estimation error:** Left: bottom-up inferred posebits are used to condition poses and reduce ambiguities in a top-down method. Right: 3D pose error (mm) as a function of the number of posebits. The vertical axis corresponds to the mean pose error computed on the Human-Eva sequences. Notice the significant improvement decreasing the error from 110 mm down to almost 70 mm when using 12 posebits. This clearly demonstrates the usefulness of using posebits during inference.

The pose estimate is given by the mode of the posterior $p(\mathbf{x}|\mathbf{z},\mathbf{r})$, obtained by evaluating the $K \times N$ poses from the proposal distribution $Q(\mathbf{x}|\mathbf{r}) = \{\mathbf{x}_{k,\mathbf{a}_n}, w_{k,\mathbf{a}_n}\}$. Recall, that $Q(\mathbf{x}|\mathbf{r})$ is represented by K poses for each of the N classes of the top ranked posebytes. To build the model for $p(\mathbf{x}|\mathbf{a})$, the poses in the Posebit MoCap set are scaled to a unit pose, *i.e.*, all bones are re-scaled by the size of a template pose. In addition, all poses are centered at the origin and the yaw angle[4] is set to zero[5].

The root orientation is estimated by uniformly sampling rotations (θ_{root}) about the vertical axis at 32 equi-spaced angles. Let $\mathcal{M}(\theta;\mathbf{x})$ be a function that rotates a pose $\mathbf{x}$ by θ degrees. Then, the pose estimate is obtained by maximizing

$$\mathbf{x}^* = \arg\max_{\mathbf{x}_{k,\mathbf{a}_n}} \left(\max_{\theta_{\text{root}}} \left(p\left(\mathbf{z}|\mathcal{M}(\theta_{\text{root}};\mathbf{x}_{k,\mathbf{a}_n})\right) w_{k,\mathbf{a}_n} \right) \right)$$

where $\mathbf{x}_{k,\mathbf{a}_n}$ is the k-th pose of the class corresponding to posebyte $\mathbf{a}_n$, and $w_{n,\mathbf{a}_k} \propto p(\mathbf{x}|\mathbf{r})$ are the importance sampling weights (see Sec. 3.2.2). In Fig. 7, we show an example of how $Q(\mathbf{x}|\mathbf{r})$ is used to reduce uncertainty about pose. A diagram of the approach is shown in Fig. 8 left.

Validation: We test the algorithm on the H-Eva sequences and report the mean pose error. Fig. 8 right shows mean pose error as a function of the number of posebits. As expected, with increasing numbers of posebits, the inference becomes less ambiguous and estimator accuracy thereby increases. The best results are obtained using 12 of the 30 random posebits currently in PbDb. That said, we think that 10 is a good trade-off between accuracy and annotation effort required to collect training data. Notice the big drop in pose error as we increase the number of posebits. We also show qualitative results in Fig. 9(a).

Our current unoptimized Matlab implementation runs at an average of 22 frames per second using 10 posebits, 4 mixtures and 10 code-poses per class.

4.2. Image Retrieval

Posebits may be useful for many applications beyond pose estimation. Here we consider image retrieval based on pose attributes. That is, posebits inferred from an image are used to retrieve other images in the DB with similar poses. We use the top ranked posebytes by the classifier to retrieve images with the similar posebyte strings. Qualitative results are shown in Fig. 9.

5. Conclusions

We introduced posebit, a semantically powerful pose descriptor. Experiments show that our selection method learns a good set of posebits, *i.e.*, retains the ones that can be reliably inferred from images and are informative about the pose. We have also shown that using posebits as a midlayer representation can improve monocular pose estimation. One of the advantages of the proposed method is that annotation is easier and more intuitive for the human observer. This enables easy collection of training data. Experiments reveal that posebits can resolve many of the monocular ambiguities and can be useful as basis for many potential applications. In particular, we do not see posebits as a competitor to existing approaches but rather as a powerful complementary feature. For future work, we plan on annotating more data, and to explore more posebit applications.

Acknowledgments. This work was also partly funded by the ERC grant DYNAMIC MINVP. D.J.F. was funded in part by the Canadian Institute for Advanced Research (CIfAR), NSERC Canada, and GRAND NCE.

References

[1] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *CVPR*, pages 1014–1021, 2009. 2

[2] A. Baak, B. Rosenhahn, M. Müller, and H. Seidel. Stabilizing motion tracking using retrieved motion priors. In *ICCV*, pages 1428–1435, 2009. 2

[3] L. Bo and C. Sminchisescu. Twin Gaussian Processes for Structured Prediction. *IJCV*, 87:28–52, 2010. 1, 2

[4] L. Bourdev, S. Maji, T. Brox, and J. Malik. Detecting people using mutually consistent poselet activations. In *ECCV*, pages 168–181. Springer, 2010. 1

[5] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *ICCV*, pages 1365–1372, 2009. 1, 2

[6] M. de La Gorce, D. J. Fleet, and N. Paragios. Model-based 3d hand pose estimation from monocular video. *TPAMI*, 33(9):1793–1805, 2011. 6

[7] J. Gall, A. Yao, and L. Van Gool. 2D action recognition serves 3D human pose estimation. In *ECCV*, pages 425–438, 2010. 2

[4] The viewpoint w.r.t. the camera is arbitrary

[5] For more implementation details see the *supplemental material*

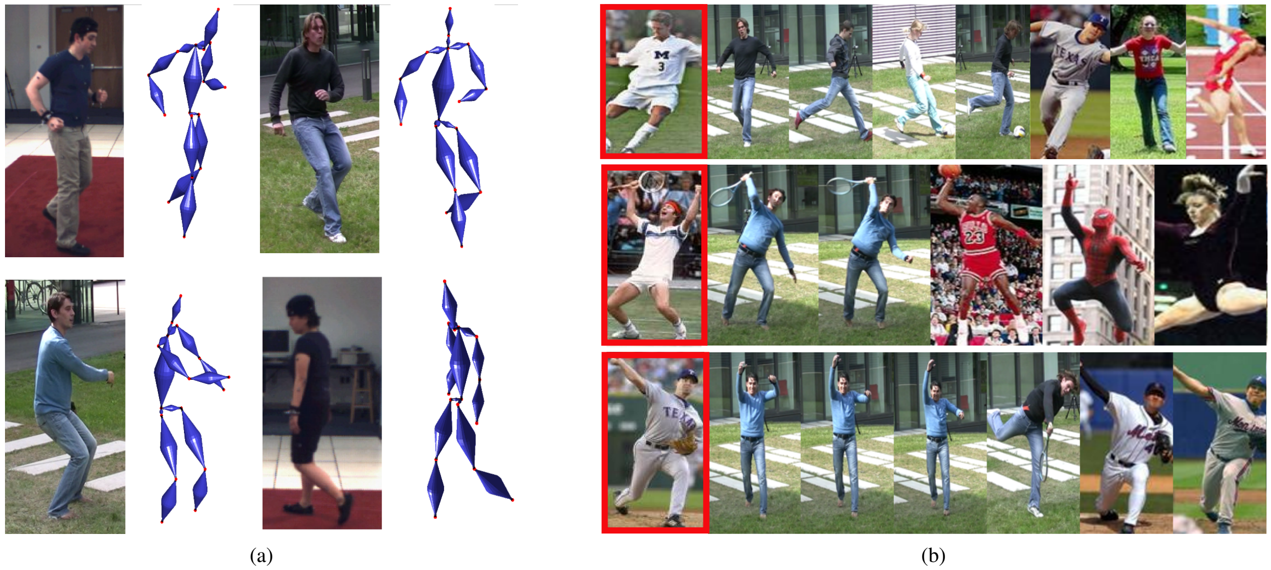


Figure 9. **(a) Pose estimation** results for images in PbDb. **(b) Retrieval**: We can use the inferred posebits from the image to retrieve images in our database with similar posebit annotations. In particular, here we retrieve images with posebyte annotations that match any of the top 2 ranked posebytes given by our model. We show the query images marked in red on the left column and the retrieved images on the right. Notice the semantic similarity in the images.

[8] C. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, pages 951–958. IEEE, 2009. 2

[9] S. Lazebnik, C. S., and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006. 5

[10] M. Lee and I. Cohen. Proposal maps driven mcmc for estimating human body pose in static images. In *CVPR*, volume 2, 2004. 2

[11] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. In *CVPR*, pages 3337–3344. IEEE, 2011. 2

[12] R. Memisevic, L. Sigal, and D. J. Fleet. Shared kernel information embedding for discriminative inference. *TPAMI*, 34(4):778–790, 2012. 1, 2

[13] G. Mori and J. Malik. Recovering 3d human body configurations using shape contexts. *TPAMI*, pages 1052–1062, 2006. 2

[14] M. Müller, T. Röder, and M. Clausen. Efficient content-based retrieval of motioncapture data. *TOG*, 24(3):677–685, 2005. 2

[15] G. Pons-Moll, A. Baak, G. J., L. Leal-Taixé, M. Mueller, H.-P. Seidel, and B. Rosenhahn. Outdoor human motion capture using inverse kinematics and von mises-fisher sampling. In *ICCV*, nov 2011. 3

[16] G. Pons-Moll and B. Rosenhahn. Model-based pose estimation. In *Visual Analysis of Humans: Looking at People*, pages 139–170. Springer, 2011. 1

[17] V. Ramakrishna, T. Kanade, and Y. Sheikh. Reconstructing 3d human pose from 2d image landmarks. In *ECCV*, pages 573–586. Springer, 2012. 2

[18] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, pages 1297–1304, 2011. 2

[19] H. Sidenbladh, M. Black, and D. Fleet. Stochastic tracking of 3d human figures using 2d image motion. In *ECCV*, volume 1843 of *LNCS*, pages 702–718. Springer Berlin / Heidelberg, 2000. 1

[20] L. Sigal, A. Balan, and M. Black. Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *IJCV*, 87(1):4–27, 2010. 1, 2, 3

[21] C. Sminchisescu, A. Kanaujia, and D. N. Metaxas. Bm3e : Discriminative density propagation for visual tracking. *TPAMI*, 29(11):2030–2044, 2007. 1

[22] C. Sminchisescu and B. Triggs. Kinematic jump processes for monocular 3d human tracking. In *CVPR*, volume 1, 2003. 2

[23] G. W. Taylor, L. Sigal, D. J. Fleet, and G. E. Hinton. Dynamical binary latent variable models for 3d human pose tracking. In *CVPR*, pages 631–638, 2010. 2

[24] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *JMLR*, 6(2):1453, 2006. 2, 5

[25] Y. Wang and G. Mori. A discriminative latent model of object classes and attributes. *ECCV*, pages 155–168, 2010. 2

[26] K. Yamaguchi, H. Kiapour, and L. E. O. andTamara L. Berg. Parsing clothing in fashion photographs. In *CVPR*, 2012. 3

[27] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, pages 1385–1392, 2011. 2, 3

[28] A. Yao, J. Gall, G. Fanelli, and L. Van Gool. Does human action recognition benefit from pose estimation? In *BMVC*, 2011. 2