Reconstructing 3D Human Poses from Keyword Based Image Database Query

Mo'taz Al-Hami
Department of Computer Information System
Hashemite University, Zarqa, Jordan 13133
motaz@hu.edu.jo

Rolf Lakaemper
Department of Computer & Information Sciences
Temple University, Philadelphia, PA 19122
lakamper@temple.edu

Abstract

The focus of this paper lies on the creation of 3D human skeleton from a set of 2D images. Unlike available approaches, which utilize a single 2D image for 3D reconstruction, the prosecuted approach utilities a set of multiple images, which are obtained from a simple query to the google image database. We only assume, that a query keyword can be linked to a set of images, which contain a representative subset related to the query. We expect the data to also contain false (i.e. non human-pose related) images. Our approach uses a human-pose based 3D shape context model for matching human-poses in 3D space, and filter them using a hierarchical binary clustering approach. The performance of this approach is evaluated using different query keywords.

1. Introduction

Human-pose estimation in 2D still images and reconstruction in 3D space is an important field in computer vision. Many applications like 3D modeling of a pose, and actions recognition show a rapidly increasing performance. Such progress is directly affected by the reliability of the recognized and reconstructed poses.

From a different point of view, pose reconstruction in 3D space can support humanoid-robots for adopting self-learning approaches. Precisely, humanoid robots could benefit from using query based 3D awareness about pose, which improves human mimicking tasks (e.g. standing, or sitting). The work in [1] uses a Genetic Algorithm approach to adjust a humanoid robot pose to an unknown sittable object (see Fig. 2). Extending such approaches could be accomplished by allowing the humanoid robot to adopt self-leaning using simple verbals describing a specific pose.

In this work, we focus on generating a target pose representations in 3D space, given a simple verbal pose related description ("standing pose", "warrior pose") as a query keyword to an image database (we use only google image search, and we do not depend on any predefined 3D databases e.g. the MoCap database). Therefore, we aim at adding semantic content to a given query keyword and its relation to a robot's skeleton model, which can be interpreted as the robot's physical self-understanding. Unlike the work in [3], which assumes a predefined prototype for linking keywords to images, this work does not assume any pre-knowledge.

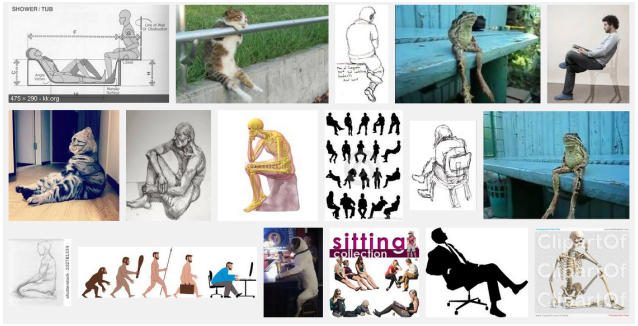
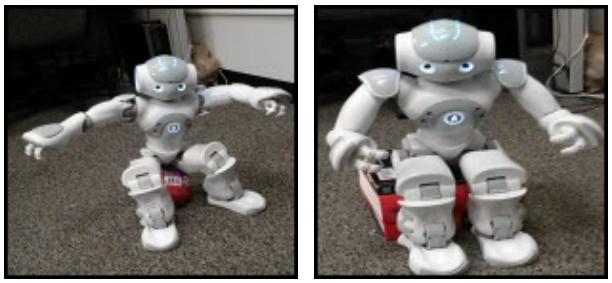


Figure 1: Small example subset of the result from a google image search on query "human sitting". The number of the retrieved images is a count of hundreds, not all showing a reasonable result.

Fig. 1 shows an example of the query "human sitting", which, among correct results, contains the seemingly unavoidable cats, dogs and frogs in the Internet. This example suggests the main challenges of such an approach to self learning postures from querying images database: (1) Images showing humans have to be identified. (2) A skeleton model has to be fit. (3) A 3D human-pose model has to be generated from the 2D skeleton. In general, a keyword query will retrieve a subset of images which are strongly related to the query, a subset of images which are weakly related to the query, and a subset of outliers (false positive images) which are completely unrelated. In this paper, we focus on extracting the strongly related images subset.

Our approach utilizes available human-pose estimators in 2D still images. Precisely, we apply the approach in [4] to estimate poses for all retrieved images. At the same time, we also use the approach in [13] to reconstruct the 3D pose



(a) Applying sitting pose on a ball (b) Applying sitting pose on a box

Figure 2: Applying sitting pose on different sittable objects as an application of the work in this paper. The humanoid robot has learned the pose [1], then applies this knowledge to adjust to the physical environment.

model. The contribution of this work is to bridge the gap between estimating the pose in 2D still images, and reconstructing a 3D pose from a 2D image. Furthermore, we extend these approaches to be fully automated based on a query keyword.

In short, the **Motivation** is: Enhancing the capability of humanoid robots to adopt a self-learning approach related to 3D poses. Such enhancement could be linked directly with the required forward and inverse kinematics which can then be applied by the robot in order to fulfill pose-related tasks.

Our contribution are: (1) We generate a 3D pose from a keyword only, with the help of a publicly available image database. (2) We define a new rotation invariant pose based 3D shape context, to define distances between poses in 3D space. (3) We apply a hierarchical binary clustering approach on a database of 3D poses to get a representative 3D model in order to filter applicable image-data. Although this approach could be applied to many other articulated objects, we focus in this paper only on human poses.

The paper is organized as follows. The related work is described in Section 2. The pose projection and reconstruction gap is discussed in Section 3. Section 4 describes poses matching in 3D space. The hierarchical Binary Clustering (HBC) approach is discussed in Section 5. Section 6 presents experiments. Finally, conclusion remarks appear in Section 7.

2. Related Work

2D human-pose estimation: Deformable models have been utilized extensively in estimating pose. Many works like [5, 7, 14] apply deformable models in pose estimation. Such approach focuses on pose parts appearance and localization in an image. Using a training set, the deformable model builds a learning system which is invariant to a pose location or appearance. For unseen images, the deformable model localizes poses without prior knowledge about these images and their backgrounds as well.

A recent work [15], focuses on a multimodal approach with different modes capturing the pose parts. The approach uses these modes as a rich structure discriminative using linear classifiers. The strength of this approach is its simplicity compared to the deformable model. The work in [6] applies shape matching to detect poses in real-time. Using a predefined database of shapes, the approach computes a distance measure to find the closest shape to the target one. A hidden Markov model, mixed with pictorial structure has been applied in [9] to track a human pose model in videos. In [10] the work benefits from the existing symmetry in the pose (i.e. left/right arms, and left/right legs) and fits a graph model to represent the pose.

Human and robot interaction: A WikiTalk application has been developed in [8]. The goal of this application is to support a social conversation between a human and a humanoid robot. In this application, a NAO humanoid robot is connected directly to Wikipedia in order to apply the knowledge to social conversation. To help a humanoid robot to apply human mimicking tasks like sitting and standing, the work in [1] applies a Genetic Algorithm approach to adjust the humanoid robot's pose in order to stabilize itself on sittable objects (boxes, and balls).

3D human-pose reconstruction: Dimensionality reduction has been applied in [13] to build a pose 3D model using a 2D image with landmarks specifying joint locations. The work employs a predefined database of poses, and uses them to reconstruct a new 3D pose using principal component analysis (PCA). In order to make the reconstructed 3D model closer to the related pose in the 2D image, the work performs a projected matching pursuit algorithm to minimize the error from the difference between landmarks and the projection of the approximate model. In [18] 3D human pose reconstruction and camera parameters estimation have been improved by forcing constraints (i.e. bone symmetry, and rigid body constraints on some body limbs).

In [17], a weak perspective projection approach has been applied to reconstruct 3D human poses, the approach is valid only for images where the depth of field of the pose is small. In [12], the work provides an attempt of reconstructing a 3D pose model for 2D images containing poses by matching with a database of predefined models. The database has a set of labeled examples representing different viewpoints with respect to the camera. The approach finds a sufficient match for a test image in the examples set, and then the 3D model is reconstructed based on the labels of those sufficient examples. The work in [19] focuses on studying activities that include human object interaction like sport activities (i.e. tennis, football). The approach proposes a mutual context approach between pose and objects. This mutual context facilitates the recognition process of an object and the estimation of a pose. Action for pose estimation has been used in [20] for providing real-time 3D

reconstruction with action detection.

3. Human-Pose Projection and Reconstruction

A camera 2D images project the 3D pose information to a 2D surface, where depth information is lost. Obviously, there is a unique 2D pose for the original 3D pose model, yet not vice versa. Assuming a pose model in 3D space is identified by a set $\mathcal{L} = \{l_1, l_2, ..., l_{s+1}\}$ of landmarks, where $|\mathcal{L}| = s + 1$, and s is the number of joints in the pose. These landmarks identify the pose joints locations in 3D space. Each landmark consists of the quad $(x, y, z, 1)$ homogeneous system. A pose $\mathcal{X} = [l_1, l_2, ..., l_{s+1}]^T$ in 3D space is projected into a 2D image such that:

$$w = (I_{|L|\times|L|} \otimes K\mathcal{M})\mathcal{X} \tag{1}$$

where $w \in \mathbb{R}^{3(|\mathcal{L}|)\times 1}$ of the projected landmarks in the 2D image, and each projected landmark consists of the triple $(u, v, 1)^T$. I is the identity matrix, $\otimes$ is the kronecker product. K is a $[3 \times 3]$ matrix representing the intrinsic camera parameters. $\mathcal{M} = [R\ t]$ is the extrinsic camera parameters and consists of the rotation matrix of the camera $R = [3\times3]$ and the translation matrix $t = [3 \times 1]$.

Reconstructing a 3D pose from a 2D still image is not a well defined task (since a point in 3D may lie anywhere along the projection ray). A recent approach [13] applies a projected matching pursuit algorithm to reconstruct a 3D pose using a 2D pose identified by a set of landmarks. A predefined database (i.e. basis poses) $\mathcal{B} = \{b_1, b_2, ..., b_t\}$, $|\mathcal{B}| = t$, of general 3D poses capturing various activities are utilized as training data. The approach tries to approximate a 3D pose as a linear combination of a basis subset $\mathcal{B}^*$, selected from the database $\mathcal{B}$. Such approximation applies linear dimensionality reduction (Principal Component Analysis (PCA)) such that:

$$\bar{\mathcal{X}} = \mu + \sum_{i=1}^{|\mathcal{B}^*|} b_i \bar{\omega}_i \tag{2}$$

where $\bar{\mathcal{X}}$ is the reconstructed pose landmarks, μ is the mean pose in the training database $\mathcal{B}$, and $\bar{\omega}$ are the associated weights for the basis poses. Assuming a fixed intrinsic camera parameter K, the approach approximates the extrinsic camera parameter $\bar{\mathcal{M}}$ using orthogonal Procrustes transformation [16].

The chosen approximated pose is the one which minimizes the projection error such that:

$$\min \parallel w - (I_{|L|\times|L|} \otimes K\bar{\mathcal{M}})\bar{\mathcal{X}} \parallel_2 \tag{3}$$

To elucidate the effect of rotation, imagine the following experiment. A single 3D pose is generated and rotated into 26 different rotation angles around the torso (i.e. a full cycle rotation of 360 degrees is divided into 26 rotations). The camera with fixed parameters is settled in a fixed location in front of the generated pose, and after each rotation the pose is projected into a 2D image (Fig. 3(d) shows part of these projected poses). We apply this approach using a single standing pose and a single sitting pose resulting in 26 standing poses, and 26 sitting poses (both are in 3D space and 2D space). Next, we perform the reconstruction approach [13] in attempt to reconstruct the original 3D poses. As a result, we have: (1) A database of original 3D poses. (2) A database of their projection into 2D space. (3) A database of the reconstructed poses in 3D space based on the previous projected poses.

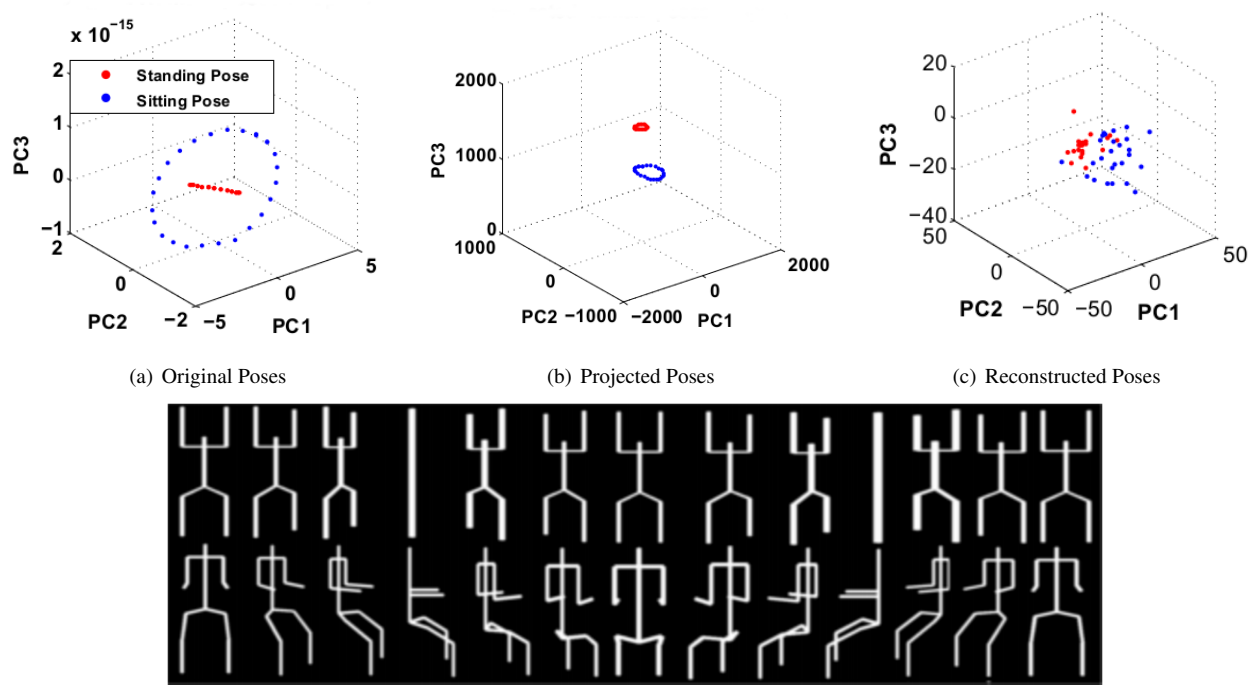
Given the original 3D poses, the projected ones, and the reconstructed poses, the objective is to show the behavior of these high dimensional space inputs (i.e. the pose feature consists of 15 landmarks), and their relation to each other under rotation. Focusing on temporal constraints, these inputs are expected to lie in a low dimensional space manifold (i.e. the original high dimensional space is mapped to a low dimensional manifold using PCA).

Here, we try to visualize these poses as points in a low dimensional manifold (single point for each pose). Fig. 3(a) shows the original poses. Fig. 3(b) shows the projected poses. Fig. 3(c) shows the reconstructed ones. The original and projected poses appear as a closed gate (i.e. cyclic) in the manifold, and poses in each activity (i.e. standing, sitting) are involved within their gate manifold in a consistent way according to their rotation amount. In the reconstructed poses case, the poses lost their consistency in the gate manifold (which is expected), however the closeness property between poses data points appears clearly. Such closeness property allows us to capture closely related poses in 2D images even if they have different appearance (i.e. rotation as shown in Fig. 3(d)). Based on the images retrieved from google images, the closeness of poses in different images is reflected by closeness in their reconstructed poses in the manifold, rather than matching the 2D poses extracted from the images.

4. Human-Poses Matching in 3D Space

Pose matching in 3D space allows us to identify rotation properly, yet due to the ambiguities this is an ill-defined or at least difficult task. Trying to capture the relation between poses in 3D of the retrieved database images would allow us to understand a better pose understanding and better pose synthesis in 3D. Therefore, we focus on building a measurement to capture the distance between arbitrary poses in 3D, taking into account rotation invariant properties.

The distance measure is needed, since we use a clustering approach to deal with false results of the image query. We utilize 3D shape context [11] and extend it to a pose



(a) Original Poses (b) Projected Poses (c) Reconstructed Poses

(d) Poses appearance after projection in 2D images

Figure 3: Process of projecting and reconstructing poses using different rotation angles. (a) Poses appearance in space by utilizing PCA. A single standing pose and single sitting pose were used. These poses were rotated with different rotation angles. (b) Projected poses appearance in space using PCA. (c) Reconstructed poses in space, the reconstruction process is based on the approach in [13]. (d) Subset of the poses in 2D images after projecting the original ones. Similar analysis based on 2D landmarks only is shown in [2]

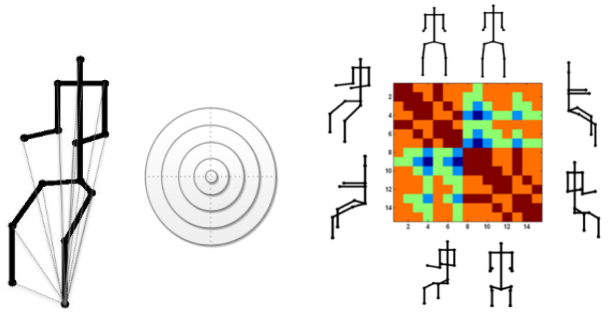
based 3D shape context. The measurement should focus on localizing the landmarks in 3D space and on preserving the rotation invariant property as well.

Let the human-pose model $\mathcal{P}$ consist of a finite collection set of joints $\mathcal{J} = \{j_1, j_2, ..., j_s\}$, where $|\mathcal{J}| = s$. These joints are organized and placed in a kinematic hierarchy model that satisfies a human-pose model (i.e. connecting the joints based on the hierarchy results in a simplified model of a human skeleton, see Fig. 4(a)). For the joints, there is a landmark set $\mathcal{L}$ defining the locations. We define a binning disk $\mathcal{B} = [b_1, b_2, ..., b_k]$ of the bin radii with increasing order. A distance map $\mathcal{D}$ measures the pairwise Euclidean distances between the landmarks $\mathcal{L}$ in the pose model $\mathcal{P}$. A 3D shape context histogram map $\mathcal{H}(|\mathcal{L}|, |\mathcal{L}|)$ keeps track of the frequencies of the distance map $\mathcal{D}$ when it is applied to the bin disk $\mathcal{B}$.

To measure the distance between two different poses $\mathcal{P}_i$ and $\mathcal{P}_j$, we utilize the χ^2 measurement as follows:

$$cost(i,j) = \frac{1}{2}\sum_{m=1}^{|\mathcal{L}|}\sum_{n=1}^{|\mathcal{L}|}\frac{[\mathcal{H}_i(m,n) - \mathcal{H}_j(m,n)]^2}{\mathcal{H}_i(m,n) + \mathcal{H}_j(m,n)} \quad (4)$$

where $cost(i,j)$ is the χ^2 distance between pose $\mathcal{P}_i$ and pose $\mathcal{P}_j$. $\mathcal{H}_i(m,n)$, is the shape context histogram for $\mathcal{P}_i$ at the position (m,n), and $\mathcal{H}_j(m,n)$ is the shape context histogram for $\mathcal{P}_j$ at the same position (m,n).



(a) Human-Pose (b) Binning Disk (c) Shape Context

Figure 4: Creating a 3D shape context feature for a pose. (a) The pose can be characterized as a network of distances between joints landmarks. (b) A Binning disk is used to characterize the network of joints distances into a histogram. (c) The final 3D shape context feature which represents the pose.

To build $\mathcal{H}$, we find the Euclidean distances between a pose $\mathcal{P}$ landmarks (i.e. create the distance map $\mathcal{D}$). Next, we normalize the distance map $\mathcal{D}$ by dividing it by the aver-

age distance $\phi(\mathcal{D})$ in the distance map $\mathcal{D}$. Finally, we count the frequencies in the 3D shape context histogram map $\mathcal{H}$ according to the distance map $\mathcal{D}$ appearance in the binning disk $\mathcal{B}$. This kind of representation of the pose is rotation invariant (the same 3D pose, but with different rotations has a unique 3D shape context histogram map $\mathcal{H}$).

4.1. Distances Between Landmarks

Fig. 4(a) shows an example of estimating the $\mathcal{D}$ map for the lower left leg. For this landmark, the distances to the remaining landmarks have to be estimated, and the same process has to be repeated for the remaining landmarks.

Fig. 4(b) shows the 3D shape context binning template $\mathcal{B}$. The final 3D shape context histogram map $\mathcal{H}$ is shown in Fig. 4(c). For each landmark in a pose there is a row in the histogram $\mathcal{H}$, captures its relation with the remaining landmarks. For each pose there is a feature vector represented by a complete 3D shape context histogram $\mathcal{H}$ which is rotation invariant. Hence, to match between two different poses, the distance between them can be estimated using χ^2 measurement as shown in Eq. 4.

4.2. Silhouette Value (SV)

SV analysis measures cluster quality and consistency. Each pose in a cluster is assigned a normalized similarity value (S_i) reflecting similarity to poses in the same cluster when compared to the poses in the other cluster (binary clustering, we only have two clusters). The similarity (S_i) depends on two main criteria: (1) Average Euclidian distance from this pose to all other poses in the same cluster (a_i). (2) Average Euclidian distance from this pose to poses from the other cluster (b_i). Using these criteria, the SV for a given pose is defined by:

$$S_i = (b_i - a_i)/\max(a_i, b_i) \tag{5}$$

Where the range of S_i is between [-1,1]. Since each pose in a cluster is assigned a separate SV, we use the average SV for all poses within a cluster $(c = k)$ as a cluster SV Such that:

$$S_c(k) = \frac{\sum\limits_{c(i)=k} S_i}{N_k} \tag{6}$$

Where $S_c(k)$ is the cluster $(c = k)$ silhouette value, N_k is the number of instances in the cluster $(c = k)$. Cluster SV analysis is used as a criterion in the hierarchical binary clustering (section [5]). It decides which cluster is the relevant one and which cluster has to be discarded.

5. Hierarchical Binary Clustering (HBC)

Highlighting cluster consistency gives clues about the main characteristics of the pose configuration within that cluster. Using a suitable feature vector (like 3D shape context histogram), we are able to cluster poses iteratively. The HBC approach (see Fig. 5) uses a χ^2 based kmeans clustering algorithm. The poses are iteratively separated into two clusters. One of them (which has a higher consistency SV) is selected to be the cluster of interest, while the other one is discarded. The HBC procedure is repeated on the selected cluster until the cluster consistency $S_c(k)$ exceeds a threshold δ (see Algorithm 1). Obtaining a consistent representative cluster provides a subset of related real poses to the query keyword. Please note, that this approach might result in multiple representations of a single pose query, which are closely related in the 3D space.

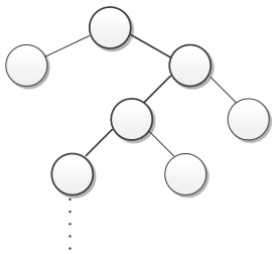


Figure 5: HBC approach. At each level poses are divided into two clusters, one of them (which is more consistent) is selected while the other one is discarded.

To measure the amount of scatter $\mathcal{S}$ in the human-pose database $\mathcal{DB}$ which includes the related 3D shape contexts, we estimate the distances between all 3D shape contexts such that:

$$\begin{aligned} \mathcal{S}(\mathcal{DB}) &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} cost(i,j) \\ &= \frac{1}{2}\sum_{k=1}^{2}\sum_{c(i)=k}[\sum_{c(j)=k} cost(i,j) + \sum_{c(j)\neq k} cost(i,j)] \end{aligned} \tag{7}$$

where N is the number of instances in the $\mathcal{DB}$ (i.e. 3D shape contexts features). Since we have only two clusters $(k = 2)$, the scatter amount can be explained as the scatter between instances in the same cluster, adding to that the scatter between instances in different clusters. The internal scatter amount between a cluster members can be rewritten as:

$$\begin{aligned} int(C) &= \frac{1}{2}\sum_{k=1}^{2}\sum_{c(i)=k}\sum_{c(j)=k} cost(i,j) \\ &= \frac{1}{4}\sum_{k=1}^{2}\sum_{\substack{c(i)=k \\ c(j)=k}}[\sum_{m=1}^{|\mathcal{L}|}\sum_{n=1}^{|\mathcal{L}|}\frac{[\mathcal{H}_i(m,n) - \mathcal{H}_j(m,n)]^2}{\mathcal{H}_i(m,n) + \mathcal{H}_j(m,n)}] \end{aligned} \tag{8}$$

Where $int(C)$ is the internal scatter for cluster C. Increasing a cluster consistency can be obtained by reducing the

scatter amount between the cluster instances such that the cluster instances are close together. The external cluster scatter $ext(C)$ can be rewritten as:

$$ext(C) = \frac{1}{2}\sum_{k=1}^{2}\sum_{c(i)=k}\sum_{c(j)\neq k} cost(i,j) \quad (9)$$

$$= \frac{1}{4}\sum_{k=1}^{2}\sum_{\substack{c(i)=k\\c(j)\neq k}}[\sum_{m=1}^{|\mathcal{L}|}\sum_{n=1}^{|\mathcal{L}|}\frac{[\mathcal{H}_i(m,n)-\mathcal{H}_j(m,n)]^2}{\mathcal{H}_i(m,n)+\mathcal{H}_j(m,n)}]$$

Applying HBC iteratively produces two new clusters at each iteration. After many iterations the instances within a cluster tends to be closer (i.e. poses with similar skeletons).

Algorithm 1 Hierarchical Binary Clustering Approach (HBC)

$\mathcal{DB} \leftarrow$ Human-pose features database
$k \leftarrow 2$ {Number of clusters}
repeat
  $[clus_1, clus_2] \leftarrow \mathbf{kmeans}(\mathcal{DB}, k)$
  $S_1 \leftarrow SV(clus_1)$ $\{Sc(1)\}$
  $S_2 \leftarrow SV(clus_2)$ $\{Sc(2)\}$
  if $(S_1 \geq S_2)$ **then**
    $\mathcal{DB} \leftarrow clus_1$
  else
    $\mathcal{DB} \leftarrow clus_2$
  end if
until $((Sc(1) > \delta)||(Sc(2) > \delta))$

Algorithm. 1 shows the main steps in HBC. The process is repeated iteratively until the poses cluster exceeds the threshold δ.

6. Experiments

We report results of 3D pose modeling using query keywords. For the experiments we used 4 different query keywords (standing pose, sitting pose, warrior pose, and tree pose). We used google images to search these keywords. All the retrieved images for each query ("sitting pose" = 917 images, "standing pose" = 924 images, "warrior pose" = 836 images, and "tree pose" = 892 images) were processed to extract the poses skeletons in these images.

6.1. 2D Human-Pose Estimation

Applying the pose estimator [4] on the retrieved images only reveals many limitations of it. Some examples from the retrieved results are shown in Fig. 6. We noticed some wrong segmentation (failing to separate the pose from the background) for many images (see 6(b), 6(e)). Another limitation happens in pose estimation even with correct segmentation, where the estimator localizes wrongly some of the pose joints (see 6(c), 6(f)). However, the estimator was able to segment and estimate many poses correctly like ones shown in 6(a), 6(d).

(a) (b) (c) (d) (e) (f)

Figure 6: Some examples of pose estimation results when applying approach in [4] on retrieved images by google for different pose conformations. The query keyword for figures (a), (c) is "standing", for figures (b), (e) is "sitting", and for figures (d), (f) is "warrior".

For the sake of clarity we can attribute the problems of extracting pose process into four main problems, false positive retrieved images (does not include poses), segmentation problem, estimation problem, and finally, overlapping problem (as much as the overlapping amount between a pose's joints increases, the quality of estimated pose decreases). Any of these demands for the clustering processing. Due to these problems, we can not trust using only single image for 3D reconstruction, and using a google image data set of retrieved images can solve this problem as shown in the results in Fig. 8.

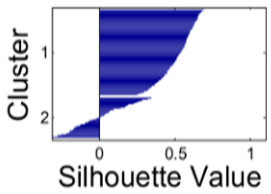
6.2. 3D Shape Contexts Based Distances

To visualize the effect of the pose based 3D shape context, we used a projected database of standing poses similar to the one used in Fig. 3(b) which contains 100 poses. We added additional sets with different amount of noise. As a result we have many data sets (each one has 100 poses) related to the same 3D standing pose, but each set has a different amount of noise. We reconstructed the approximate 3D poses, and applied 3D shape context to build the poses features. Fig. 7 shows the average χ^2 distance between the 3D shape contexts of each set and the shape context of the original standing pose. The results shows the effect of added noise on the average χ^2 distances, which increases along with the added noise amount.

6.3. Filtered 3D Human-pose

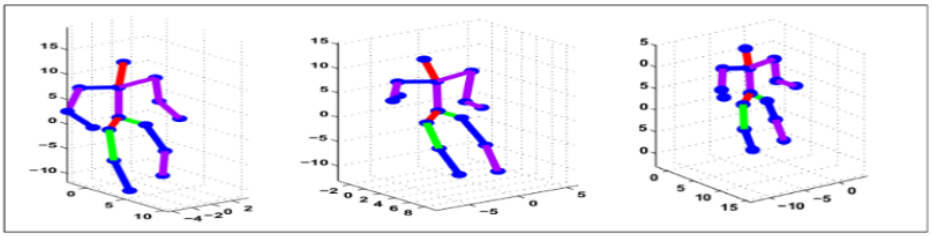
Fig. 8 shows the result, i.e. the final cluster members for each query keyword ("Standing pose", "Sitting pose", "Warrior pose", and "Tree pose"). For "standing pose", the HBC has performed 8 iterations, and the final cluster consists of 3 poses as shown in Fig. 8(d). The filtered poses are sufficiently represent a valid standing pose. A specific pose keyword might have different variations in real life (see the real life variations in Fig. 8(c), 8(g), 8(k), 8(o)). For example, in "sitting pose" the pose might sit on different types of chairs or might sit on the floor. Such variations provide us with different valid representations which are suitable to be

(a) First 20 retrieved images by google for Standing pose.



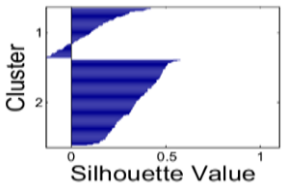
(b) SV in HBC iter(1).

(c) Real life variations



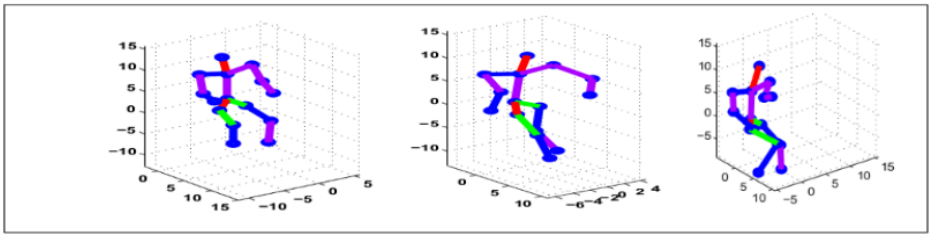
(d) The representative 3D models for standing pose.

(e) First 20 retrieved images by google for Sitting pose.



(f) SV in HBC iter(1).

(g) Real life variations



(h) The representative 3D models for sitting pose.

modeled in 3D space. However, in the HBC we focus on one of these valid representations. Throughout clustering poses in 3D space, the HBC generates 2 clusters iteratively and both of them might include some of the valid variations.

In the case of "sitting pose", the estimated 2D poses were weak. The problem with "sitting pose" is the amount of the overlap between human skeleton's joints (i.e. overlapped legs, or overlapped arms). The overlap between pose joints increases the ambiguity in the poses estimator's deformable model, as a result it increases the probability of getting wrong 2D pose estimation. Fig. 8(h) shows one pose which is close to a valid sitting pose, while the rest are weak valid sitting poses.

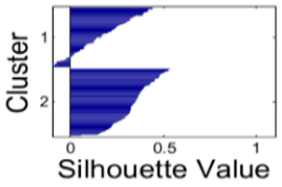
The benefit of using a rotation invariant representation like 3D shape context, appears clearly in the case of the warrior pose (Fig. 8(l)) where the generated poses are closely related, however they have different rotations. For "tree pose", the HBC returned 4 poses which are closely related. Silhouette values (SV) are shown for the first iteration (i.e. iter(1)) in the HBC for all queries (see Fig. 8(b), 8(f), 8(j), 8(n)).

Part of the retrieved google images are shown in Fig. 8(a), 8(e), 8(i), 8(m). The first 20 retrieved images are shown for each query. Although these images have some strongly related images for each query, the human-poses appearance and estimation makes it difficult to use small number of images. Hence, we use the whole retrieved images in order to get a subset of images having closely related poses.

7. Conclusion

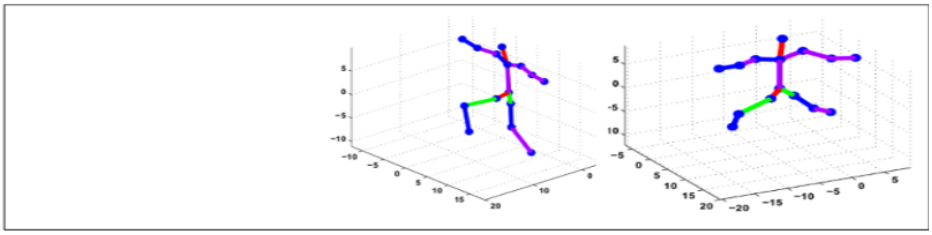
In this paper, we propose a 3D shape context for human-pose. We used such pose description jointly with hierarchical binary clustering approach (HBC). The motivation of this approach is to allow pose related query keyword to be translated into a 3D pose model. Using such approach a query keyword might have multiple representations. Abil-

(i) First 20 retrieved images by google for Warrior pose.



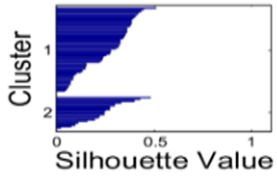
(j) SV in HBC iter(1).

(k) Real life variations



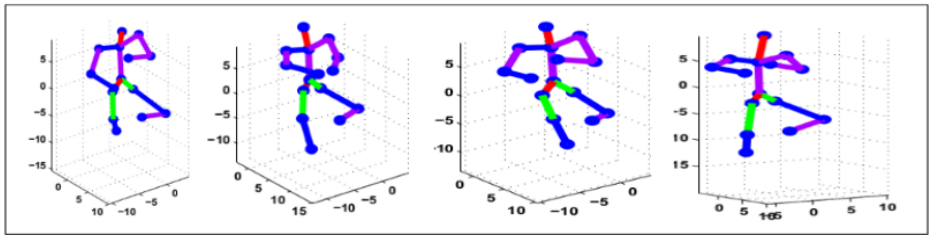
(l) The representative 3D models for warrior pose.

(m) First 20 retrieved images by google for Tree pose.



(n) SV in HBC iter(1).

(o) Real life variations



(p) The representative 3D models for tree pose.

Figure 8: The selected poses by using the HBC approach. The clustering algorithm applied until the cluster consistency exceeds a threshold value δ. Silhouette value (SV) is used to measure a cluster consistency, and to decide which cluster to select as well.

ity to translate a query keyword into 3D representations bridges the gap between 2D pose estimation in 2D images, and reconstructing a 3D pose using 2D image with landmarks. This kind of 3D modeling opens the door to extend many applications (e.g. for humanoid robots to adopt self-learning approaches to recognize the surrounding environment). In future work, we will focus on how to decide the best 3D pose representation among a set of available models, such that it adapts well to the current surrounding environment.

References

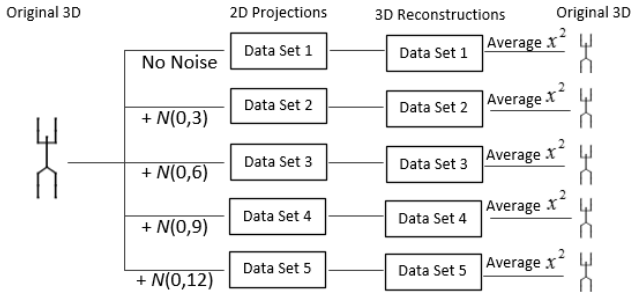
[1] M. Al-Hami and R. Lakaemper. Sitting pose generation using genetic algorithm for nao humanoid robot. In *IEEE Workshop on Advanced Robotics and its Social Impacts (ARSO), 2014*, pages 137–142. IEEE, 2014.

[2] M. Al-Hami and R. Lakaemper. Towards human pose semantic synthesis in 3d based on query keywords. In *Proceedings of the 10th International Conference on Computer Vision Theory and Applications - Volume 3: VISAPP, (VISIGRAPP 2015)*, pages 420–427. INSTICC, SciTePress, 2015.

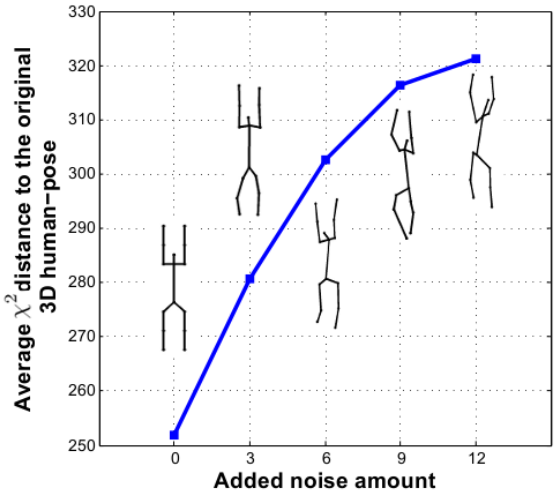
[3] M. Eichner and V. Ferrari. Human pose co-estimation and applications. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(11):2282–2288, 2012.

[4] M. Eichner, M. Marin-Jimenez, A. Zisserman, and V. Ferrari. 2d articulated human pose estimation and retrieval in (almost) unconstrained still images. *International Journal of Computer Vision*, 99(2):190–214, 2012.

[5] V. Ferrari, M. Marin-Jimenez, and A. Zisserman. Progressive search space reduction for human pose esti-



(a)



(b)

Figure 7: Average χ^2 distance between 3D pose sets having different levels of added noise and the original 3D pose. We use the χ^2 to calculate distances between the related 3D shape contexts. The added noise follows normal distribution $N(0, \sigma^2)$ where σ^2 is the added noise amount.

mation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008*, pages 1–8. IEEE, 2008.

[6] D. Gavrila. Pedestrian detection from a moving vehicle. In *Computer Vision ECCV 2000*, pages 37–49. Springer, 2000.

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

[8] K. Jokinen and G. Wilcock. Multimodal open-domain conversations with the nao robot. In *Natural Interaction with Robots, Knowbots and Smartphones*, pages 213–224. Springer, 2014.

[9] X. Lan and D. P. Huttenlocher. A unified spatio-temporal articulated model for tracking. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2004*, volume 1, pages I–722. IEEE, 2004.

[10] X. Lan and D. P. Huttenlocher. Beyond trees: Common-factor models for 2d human pose recovery. In *Tenth IEEE International Conference on Computer Vision (ICCV), 2005*, volume 1, pages 470–477. IEEE, 2005.

[11] G. Mori, S. Belongie, and J. Malik. Efficient shape matching using shape contexts. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (11):1832–1837, 2005.

[12] G. Mori and J. Malik. Estimating human body configurations using shape context matching. In *Computer Vision ECCV 2002*, pages 666–680. Springer, 2002.

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

[14] D. Ramanan. Learning to parse images of articulated bodies. In *Advances in Neural Information Processing Systems*, pages 1129–1136, 2006.

[15] B. Sapp and B. Taskar. Modec: Multimodal decomposable models for human pose estimation. In *IEEE Conference onComputer Vision and Pattern Recognition (CVPR), 2013*, pages 3674–3681. IEEE, 2013.

[16] P. Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10, 1966.

[17] C. J. Taylor. Reconstruction of articulated objects from point correspondences in a single uncalibrated image. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2000.*, volume 1, pages 677–684. IEEE, 2000.

[18] X. K. Wei and J. Chai. Modeling 3d human poses from uncalibrated monocular images. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1873–1880. IEEE, 2009.

[19] B. Yao and L. Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010*, pages 17–24. IEEE, 2010.

[20] T.-H. Yu, T.-K. Kim, and R. Cipolla. Unconstrained monocular 3d human pose estimation by action detection and cross-modality regression forest. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3642–3649. IEEE, 2013.