Deep Convolutional Networks for Marker-less Human Pose Estimation from Multiple Views

Matthew Trumble
Centre for Vision, Speech and Signal Processing
University of Surrey
Guildford, UK

Andrew Gilbert
Centre for Vision, Speech and Signal Processing
University of Surrey
Guildford, UK

Adrian Hilton
Centre for Vision, Speech and Signal Processing
University of Surrey
Guildford, UK

John Collomosse
Centre for Vision, Speech and Signal Processing
University of Surrey
Guildford, UK

ABSTRACT

We propose a human performance capture system employing convolutional neural networks (CNN) to estimate human pose from a volumetric representation of a performer derived from multiple view-point video (MVV). We compare direct CNN pose regression to the performance of an affine invariant pose descriptor learned by a CNN through a classification task. A non-linear manifold embedding is learned between the descriptor and articulated pose spaces, enabling regression of pose from the source MVV. The results are evaluated against ground truth pose data captured using a Vicon marker-based system and demonstrate good generalisation over a range of human poses, providing a system that requires no special suit to be worn by the performer.

CCS Concepts

•**Computing methodologies** → **Motion capture; Neural networks;** *Volumetric models;*

Keywords

Deep Learning, Pose Estimation, Multiple Viewpoint Video

1. INTRODUCTION

Performance capture is widely used in the entertainment and biomechanics industries for the capture of human motion. The state of practice in the commercial domain, for example, the Vicon and Optitrack systems, require a special suit with markers to be worn by the performer and specialist cameras and specific lighting conditions in which to work. These requirements restrict the process to dedicated motion capture studios, which is often an undesirable location for the principal footage of a film, for example, thereby requiring multiple shoots and additional expense. The technique we propose in this paper aims to estimate human pose from principle footage using multiple, synchronised video cameras, thereby avoiding the inconvenience and expense of specialist equipment.

The presented techniques employ CNNs to the task of human pose estimation in two different modes. CNN_1 learns a rotationally invariant pose descriptor and CNN_2 regresses pose directly. We demonstrate the efficacy of the CNNs for pose classification and regression; the latter by both direct regression and by learning a manifold embedding for CNN_1 space using Gaussian Processes [14]. Both networks share the same volumetric data input format, reconstructed through triangulation of performer silhouettes within the MVV sequence. Without loss of generality, we recover a probabilistic visual hull (PVH) [6], a coarse voxel representation of the performer from which we extract a multi-scale polar representation ('spherical histogram') used as input to the first CNN layer.

2. RELATED WORK

Human pose estimation (HPE) is concerned with identifying the skeletal position from visual data in terms of limb location/orientation or a probability map of their locations in the source material. The first challenge of such systems is identifying people in the source images or video, commonly achieved by techniques such as background subtraction [27, 2] or sliding window classifiers [23, 4]. Existing approaches can then be split into two broad categories; methods that use a top-down approach to fit an articulated model to the source data and those that do not rely on an explicit a proiri model and identify body parts in a bottom-up approach.

Bottom-up HPE is driven by image parsing to isolate components e. g. Srinivasan and Shi [21] use graph-cut to parse a subset of salient shapes from an image and group these into a model of a person. However, the approach is sensitive to clutter which interferes with the segmentation. Mori and

CVMP '16, December 12-13, 2016, London, United Kingdom

DOI: http://dx.doi.org/10.1145/2998559.2998565

Malik identified the position of individual joints in a 2D image. Scale and symmetry constraints were used to establish the correspondence between a 2D query image and training images annotated *a priori* with joint positions [16]. Ren *et al.* recursively split Canny edge contours into segments, classifying each as a putative body part using cues such as parallelism [20]. Ning *et al.* [7] apply vector quantization to learn codewords for body zone labelling. Ren et al. also use BoVW for implicit pose estimation as part of a pose similarity system for dance video retrieval [19].

Top-down approaches to 2D HPE fit an articulated limb model to data incorporating kinematics into the optimisation to bias toward possible configurations. Lan and Huttenlocher [13] provide such a model using joint angles and considering the conditional independence of parts; Inter-Limb dependencies (e. g. symmetry) are not considered. A more global treatment is proposed in [10] using linear relaxation but performs well only on uncluttered scenes. Pictorial structures [5] encode a graphical model of limb components that are fitted by decomposing the objective function across edges and nodes in a tree. Spatio-temporal tracking of pictorial structures is applied to HPE in [12], and the fusion of pictorial structures with Ada-Boost shape classification was explored in [3]. Agarwal and Triggs used non-linear regression to estimate pose in 2D silhouette images [1].

Studies have begun to leverage the power of convolutional neural networks to pose estimation, following in the wake of the eye-opening results of Krizhevsky et al. [11]. Toshev and Szegedy [22], in the DeepPose system, use a cascade of networks to estimate 2D pose in images. Descriptors learned by a CNN have also been used in 2D pose estimation from very low resolution images [17]. The more general challenge of 3D object pose estimation [26, 25] has also benefited from deep learning approaches. The challenge of estimating 3D human pose from MVV is currently less explored, although related work has been undertaken aligning pairs of 3D human body poses [24], and estimating pose via a tracked 4-D mesh of a human performer from video reconstruction [9]. However mesh tracking is subject to frequent manual correction and is reliant upon strong surface texture cues and absence of surface deformation e. g. due to clothing. Our approach is unique in adopting a data-driven, machine learning approach to 3D pose estimation, using only a coarse visual hull (PVH) derived from the MVV to drive learning of the CNN descriptor and manifold embedding.

3. MARKERLESS POSE ESTIMATION

Our approach accepts a multiple viewpoint video (MVV) sequence as input, shot using synchronised calibrated cameras surrounding the performance. The volume visible in all camera views is the effective zone of capture (the 'capture volume'); here, approximately 6×4 meters. We first reconstruct a coarse geometric proxy within the capture volume from each frame of the sequence (subsec. 3.1). The proxy is then resampled about its centroid into a log-polar representation at multiple scales ('spherical histogram'). We employ this representation to train two separate proposed convolutional neural networks,

- A convolutional neural network (CNN_1) configured for a supervised classification task. CNN_1 is trained using labelled examples of several distinct poses exercising the full range of typical human motion (subsec. 3.1.1). Descriptors are extracted from the first fully connected layer of the network, and a non-linear manifold embedding learned over a combined space of the CNN descriptors and joint angle estimates (subsec. 3.3). The manifold enables pose regression from descriptors derived from each MVV frame.

- Alternatively CNN_2 adapts the network structure of CNN_1 to introduce a Euclidean loss layer(subsec. 3.3.4) as the final layer to allow the network to learn to regress the estimated human pose.

Figure 1 illustrates the full pipeline.

3.1 Volumetric Representation

The capture volume is observed by C camera views $c = [1, C]$ for which extrinsic parameters $\{R_c, COP_c\}$ (camera orientation and focal point) and intrinsic parameters $\{f_c, o_c^x, o_c^y\}$ (focal length, and 2D optical centre) are known. A geometric proxy of the performer is built via an adapted form of Grauman et al.'s probabilistic visual hull (PVH) [6] computed from soft foreground mattes extracted from each camera image I_c using a bluescreen chroma key. To compute the PVH we coarsely decimate the capture volume into a set of voxels at locations $\mathcal{V} = \{V_1, \ldots, V_m\}$; a resolution of 5cm^3 is used in our experiments. The probability of the voxel being part of the performer in a given view c is:

$$p(V|c) = B(I_c(x[V_i], y[V_i])) \tag{1}$$

where $B(.)$ is a simple blue dominance term derived from the RGB components of $I_c(x, y)$, i. e. $1 - \frac{B}{R+G+B}$, and (x, y) is the point within I_c that V_i projects to:

$$x[V_i] = \frac{f_c v_x}{v_z} + o_c^x \tag{2}$$

$$\text{and} \quad y[V_i] = \frac{f_c v_y}{v_z} + o_c^y, \tag{3}$$

$$\text{where} \quad \begin{bmatrix} v_x & v_y & v_z \end{bmatrix} = COP_c - R_c^{-1} V_i. \tag{4}$$

The overall probability of occupancy for a given voxel $p(V)$ is:

$$p(V_i) = \prod_{i=1}^{C} 1/(1 + e^{p(V|c)}). \tag{5}$$

We compute $p(V_i)$ for all $V_i \in \mathcal{V}$ to create a volumetric representation of the performer for subsequent processing. In practice, $B(.)$ is computed in parallel via a GPU-equipped PC attached to each camera, yielding a soft matte at 25 frames/second. This allows for independent solution of eqs.1-4 prior to aggregating results on a single machine to solve eq. 5 yielding a streaming PVH.

3.1.1 Log-Polar Representation

$\mathcal{V}$ is next resampled into a log-polar representation, quantizing longtitude and latitude into N regular intervals, and yielding a 2D signal $S(\phi, \theta) \in \Re^{N^2}$ derived from a sub-volume of interest in $\mathcal{V}$ local to the weighted centroid $\mu(\mathcal{V})$ of the PVH

$$\mu(\mathcal{V}) = \sum_{V_i \in \mathcal{V}} V_i p(V_i). \tag{6}$$

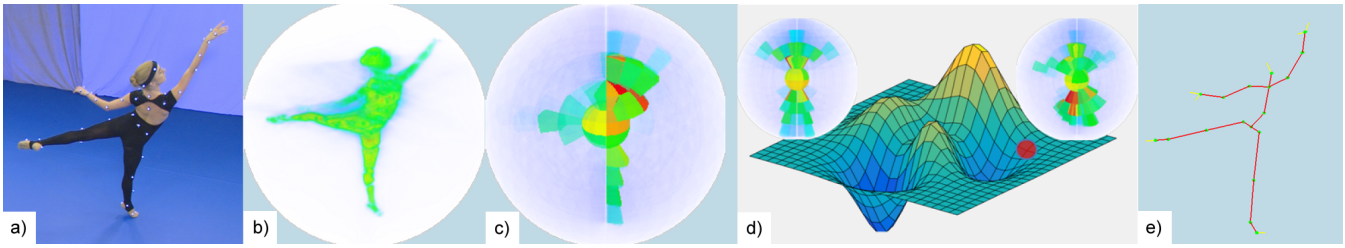


Figure 1: Overview of proposed technique. MVV is captured (a) and a geometric proxy / PVH built (b). The PVH is sampled into log-polar form at multiple scales and passed through a CNN to learn a rotationally invariant descriptor (c). A non-linear manifold embedding of the combined CNN and joint angle space (d) is learned under supervision to regress a pose estimate (e). Note markers visible in (a) are not used by our approach; they provide ground-truth for quantitative comparison against a commercial system.

Each sample $S(\phi,\theta)$ aggregates $p(V_i)$ from the subvolume of the PVH within interval $\left[\frac{2\pi}{\phi},\frac{2\pi}{\phi+1}\right]$ and $\left[\frac{2\pi}{\theta},\frac{2\pi}{\theta+1}\right]$ within a particular distance interval from μ. Writing voxels within that interval $\mathcal{V}'$:

$$S(\phi,\theta)=\sum_{V_i\in\mathcal{V}'}1/(1+e^{-(p(V_i)-D(\mathcal{V}))}) \tag{7}$$

where $D(\mathcal{V})$ is a dynamic threshold computed for the voxel distribution that serves to normalise against larger $p(V_i)$ occuring due to appearance variation (e. g. difference in clothing colour) between performers. D is computed using a cumulative count of voxels with occupancy probability exceeding τ for all $\tau=[0,1]$, yielding a 1D signal $s(\tau)$:

$$s(\tau)=\sum_{V_i\in\mathcal{V}}p(V_i)>\tau \tag{8}$$

from which D is derived from the turning point in $s(\tau)$

$$D(\mathcal{V})=\underset{\tau}{\operatorname{argmax}}\left|\frac{\delta s(\tau)}{\delta\tau}\right|. \tag{9}$$

The use of a log-polar representation follows successes in prior work on human 3D mesh alignment [8] and general 3D object retrieval [15] that employ spherical histogram representations to match on coarse shape. Our work extends these to compute a set $\mathcal{S}$ of $S(\theta,\phi)$ each computed from a radial interval $[R_r,R_{r+1}]\in\mathcal{R}=\{0.2,0.4,0.6,0.8,1.0\}$.

We explored two approaches to deriving robust descriptors invariant to performer viewpoint (i. e. longitudinal rotation), and evaluate these in Sec. 4.2.

3.1.2 *Frequency Domain Variant*

We investigated the removal of phase information in θ by computing a frequency domain (DFT) representation of each row $S\in\Re^{N\times1}$ and considering only the complex i. e. frequency magnitude. Similar to classical Fourier Descriptors this results in a shorter descriptor in $\Re^{N\times\frac{N}{2}}$ invariant to rotation of signal in θ. Such a descriptor space with inbuilt invariance is preferable to the optimisation strategies of prior 3D pose matching work using log-polar voxel descriptors for mesh alignment[8], which try all possible rotations to evaluate pair-wise shape similarity and so are both slow and incompatible with a machine learning approach to pose estimation.

3.2 Learning the Pose Descriptor

We investigated the use of either $\mathcal{S}$ or its above-described frequency domain variant, as source data to train a convolutional neural network (CNN_1) to perform a supervised pose classification task (the relative performance of each data source type is evaluated in Sec. 4.2). Similar to modern image classification work, which now extensively employs CNNs, we sample a high-dimensional descriptor from the first fully connected layer following training convergence. Our CNN adapts the architecture of [11] and is illustrated in Figure 2.

Our first convolutional layer consists of 96 kernels of size $11\times11\times5$ and filters image data of size 227×227 with four stride pixels. The local response normalisation (LRN) layer with local size 5×5 follows, after max pooling with 3×3 with two stride pixel size. The output of max pooling becomes the input of the second layer with 256 kernels of size $5\times5\times48$. After second layer, another normalisation with the same local size of previous LRN and max pooling with 3×3 with two stride pixel size is performed. The third layer, fourth, and fifth layers have 384,384 and 256 kernels of size 3×3 respectively. Max pooling exists between the fifth and sixth layers only. The two final layers are fully connected. We evaluate (Sec. 4.2) fully connected layers of 1024 (1K) and 4096 (4K) leading to descriptors of similar dimension. A softmax layer is appended with 20 outputs in line with the supervised training task we now describe. Fig. 2 illustrates the CNN architecture.

3.2.1 *CNN Training and Data Augmentation*

CNN_1 was trained using a purpose-built dataset of labeled MVV footage comprising $\sim25k$ multiple-view frames (and so, $\mathcal{V}$) from 8 cameras. 25 individuals in a variety of clothing (shorts, trousers, dresses) were filmed executing repetitions of 20 distinct poses following the Vicon "Range of Motion" (ROM) sequence used to calibrate commercial motion capture equipment to exercise all major modes of human pose variation. Examples of the resulting PVH in a range of poses are given in Figure 3.

Soft-max loss was used to train the CNN using 80% of this data to recognize the 20 poses, subject to data augmentation. Sec. 4.2 reports the results of training the CNN using either the spatial or the frequency domain, representations of $\mathcal{S}$ under two data augmentation (DA) strategies:

- **DA1:** Longtitude Jitter. $\mathcal{S}(\phi,\theta)$ was subject to random rotation of $\theta=[0,2\pi]$.
- **DA2:** As DA1 with the addition of Gaussian noise

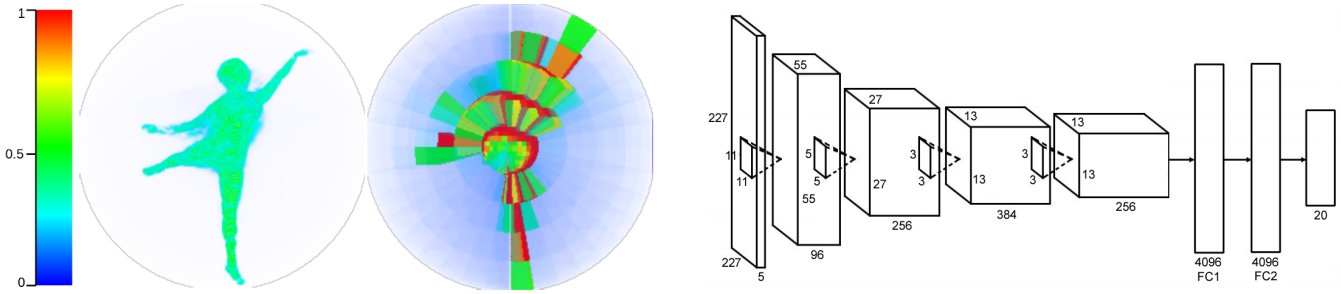


Figure 2: Architecture of the proposed CNN_1 classifier, (right) operating over the multi-scale log-polar representation parsed from MVV (left) and normalised against appearance variation via the dynamic thresholding operation (middle). Far-left shows the colour key for voxel occupancy.

Figure 3: Visualising a subset of PVH responses ($\mathcal{V}$) for CNN classifier training on the *ROM* dataset (comprising $\sim 25k$ MVV video frames of 25 performers in 20 diverse poses). Voxel colour key given in Fig. 2.

and blur at random scale.

Training proceeded over 100 epochs in our experiments, using a mini-batch size of 200. At test time, the CNN is truncated at the second fully connected layer yielding a vector of convolutional feature responses $\mathcal{C}$ that serves as our pose descriptor.

3.3 Joint Manifold Embedding

We perform human pose estimation via a supervised learning approach in which a correlation is learned between exemplar pairs of descriptors (in CNN_1 space $\mathcal{C}$) and a vector of 21 skeletal joint angles expressed in quaternion form as ordered pairs (we denote this space $\mathcal{Q} \in \Re^{21\times 4}$). The joint angles are measured by professional motion capture engineers using a marker-based commercial system (Vicon Blade). A dataset of correspondences $c_i \mapsto q_i$ between $c_i \in \mathcal{C}$ and $q_i \in \mathcal{Q}$ are gathered as a one-off process using a short training sequence exercising degrees of motion likely to be encountered at test time. The training sequence comprises simultaneously captured MVV and skeletal joint angles estimated by a commercial marker-based motion capture solution (Sec. 4.1). We investigate four approaches to the generalisation of these sparse training correspondences to a dense mapping $\mathcal{C} \mapsto \mathcal{Q}$ suitable for inferring performer pose $P \in \mathcal{Q}$ from a query point $c \in \mathcal{C}$ derived from MVV at test time.

3.3.1 Nearest Neighbour (Baseline)

The naïve approach to creating a dense mapping is to snap a query pose descriptor to closest $c_i \in \mathcal{C}$ i. e. perform a nearest neighbour lookup to obtain pose estimate P_{nn}

$$P_{nn} = q_j, \text{where} \quad j = \underset{i}{\text{argmin}} \left| c - c_i \right|. \tag{10}$$

This can be implemented in real-time (i. e. 25 frames/second) using a kd-tree pre-built over c_i. Under this approach no constraints are imposed to guard against invalid poses, since no generalisation beyond training is performed.

3.3.2 Piecewise linear embedding (Baseline)

A linear subspace model is learned local to each c_i based on the local K most proximate training samples c'_j where $j = \{1..K\}$. We construct this model as an undirected graph connecting c_i to c'_j, forming a piecewise linear manifold over $\mathcal{C}$ covering likely poses and (linear) interpolations between similar poses. In our experiments $K = 5$ providing a balanced trade-off between speed and accuracy. We estimate the pose P_{ple} under this model as:

$$P_{ple} = \sum_{j \in J} d(c, c_j) q_j \tag{11}$$

where $d(a, b)$ is a value proportional to geodesic distance between two points on the graph manifold, and J is the set of K nearest neighbours to c in $\mathcal{C}$.

3.3.3 Non-linear embedding

Gaussian processes (GP) [18] are a popular approach for creating smooth non-linear mappings between continuous spaces of differing dimension. We adopt the Gaussian Process Latent Variable Model (GP-LVM) [14] as an supervised means for learning a non-linear manifold embedding within joined space $\mathcal{C} \times \mathcal{Q}$ i. e. to model the manifold upon which vectors $[c_i\ q_i]$ lie. The essence of the GP is to model the probability of a point in space $c_i \in \mathcal{C}$ as the covariance of a kernelized Gaussian in $\mathcal{Q}$; as with prior work we adopt an RBF kernel with hyper-parameters α and γ:

$$\kappa(x_a, x_b) = \alpha e^{\left(-\frac{1}{2}\gamma |x_i - x_j|^2\right)} \tag{12}$$

i. e. given kernel $\kappa(.)$ the probability of c_i is:

$$p(c_i, q_i) = \mathcal{N}(c_i, \kappa(x_a, x_b)). \tag{13}$$

Given N training pairs $\{c_i, q_i\}$ and writing the sets of these data C and Q respectively we train the GP-LVM model by maximising:

$$P(C|Q) = \prod_{i}^{N} \mathcal{N}\left(c_i, \kappa(Q,Q) + \sigma^2 I\right), \tag{14}$$

where I is the identity matrix, by seeking appropriate values of α, γ, and noise parameter σ. Once trained, creating a pose estimate P_{nle} from c is a straightforward matrix multiplication:

$$P_{nle} = \kappa(c,Q)\left[\kappa(Q,Q)\right]^{-1} C \tag{15}$$

3.3.4 CNN Pose Regression

CNN_1 was adapted to enable direct regression from log-polar representation $\mathcal{S}$ (see Sec. 3.1.1) to pose estimate P_{reg}, output as joint locations in 3D space. This new network, CNN_2, changes only the final layer to become length 63, to encapsulate 21 joint coordinates. A Euclidean loss function is used for network training, and DA2 data augmentation is applied, as described in section Sec. 3.2.1.

4. EXPERIMENTS AND DISCUSSION

We report experiments evaluating our proposed technique under two distinct tasks. First, we investigate the efficacy of several variants of our pose descriptor (Sec. 3.1.1) at pose classification. The *ROM* dataset of $\sim 25k$ MVV frames is used to train and test the descriptors using both CNN and Support Vector Machine (SVM) classifiers (described in Sec. 3.2.1). The most promising descriptor for pose classification is carried forward to the second experiment, pose estimation, which compares the efficacy of three proposed manifold embedding approaches (P_{nn}, P_{ple}, and P_{nle}) in that descriptor's space. We use a specially captured hybrid dataset *Ballet* to evaluate the accuracy of pose estimate against a ground-truth for all methods (including P_{reg}). All of our datasets will be publicly released under a Creative Commons non-commercial attribution license.

4.1 Hybrid Dataset for Pose Estimation

We captured a hybrid dataset *Ballet* comprising five MVV sequences each of which were accompanied by ground-truth measurement of 21 skeletal joint angles, produced by a professional motion capture engineering using a Vicon motion capture system. Obtaining this ground-truth required visible markers to be worn, however, these were not used in our pose estimation process, and the size of these markers (0.5cm^3) was negligible relative to the coarse (5cm^3) volume decimation of the visual hull. Fig. 6 (top row) illustrates sample frames. The dataset was captured in an indoor studio with nine video cameras on a ring gantry suspended at approximately 2.5 metres, surrounding a 8×4 metre capture volume. The Vicon system used was a T-Series with 12 infra-red cameras positioned at similar locations on the ring. The five sequences total 9434 MVV frames.

To evaluate performance on *Ballet*, two metrics are used. The first is the the total angular error θ_{error} between the estimated quaternion angle of each joint q_e and the groundtruth joint angle q_g. The quaternion rotating q_e to q_g is in vector form:

$$[q_r \quad \overrightarrow{q_v}] = q_e^{-1} q_g. \tag{16}$$

Table 1: Classification accuracy of DFT and CNN based descriptors

Descriptor type	Classifier	MAP(%)
DFT	SVM	75.62
4K CNN_1+DA1	CNN_1	77.55
4K CNN_1+DA2	CNN_1	80.97
4K CNN_1+DA2	SVM	**87.99**

the components of which may be converted into axis-angle form to extract the angular error:

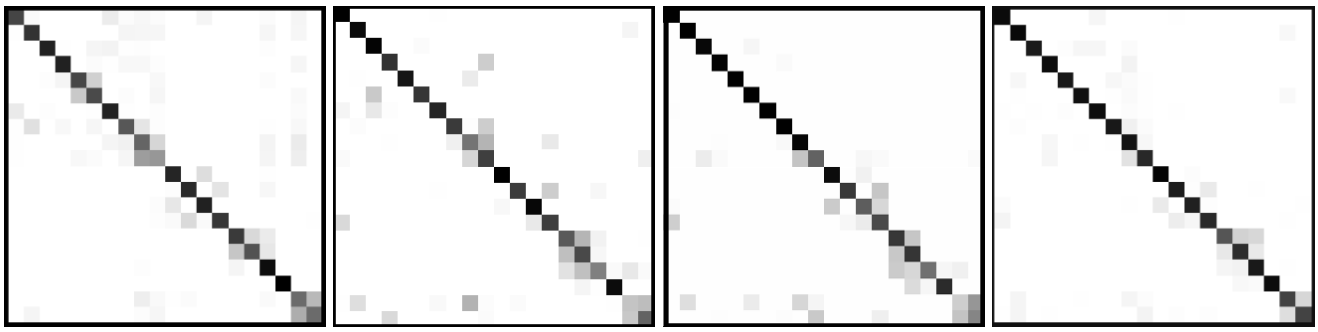
$$\theta_{error} = 2abs\left[\arctan(\frac{|\overrightarrow{q_v}|}{q_r})\right] \tag{17}$$

The average angle error is accumulated over all 21 joints, to provide a per frame error quaternion angle measured in degrees. We also present the total cumulative error in joint positive via a simple Euclidean distance between the estimated joint and ground-truth joint locations in 3D space, expressed in millimetres (mm). Note that errors in the absolute position of joints are compounded by error in the joint preceeding it within the articulation.

4.2 Pose Classification Experiments

We evaluated the proposed CNN_1 architecture of Sec. 3.1.1 under the two proposed training (data augmentation) variants: DA1, and DA2. We trialled the direct classification of the MVV frames via CNN_1 as well as extraction of the pose descriptor from layer FC2 and subsequent classification using a non-linear SVM. As a baseline, we compare against the direct use of log-polar representation for inter-frame pose matching proposed by [8], for fairness of comparison adapting this to a form invariant to longitudinal rotations via the DFT approach described in Sec. 3.1.1. For these tests, the *ROM* dataset was split into training and test datasets under leave-one-out cross-validation with the poses of 20 people in the training set (a total of $\sim 20,000$ frames) and the remaining five people (~ 5000 frames) reserved for the test dataset. The mean average precision (MAP) score over the test data is shown in Table 1.

Comparing augmentation strategy **DA2** for CNN_1 descriptor of 4K length (and directly applying the CNN as classifier) against the **DFT** encoding of the log-polar representation, performance increases around 3%. However when the FC2 layer is used as the descriptor in conjunction with an SVM, there is a 12% increase in classification performance on the *ROM* dataset. This improvement can be visualised in the class confusion matrices, as shown in Fig 4, which shows the classification matrix of the DFT descriptor and the states of the augmented CNN. Increasing the volume and diversity of data augmentation, therefore, reduces class confusion as does the use of the raw descriptor (FC2) combined with an SVM classifier. Much of the remaining limited confusion occurs between left and right variants of the pose classes. This ensures that for a single hand labelled dataset, the CNN_1 descriptor can efficiently encode the log-polar information for effective discrimination of poses. The superior performance of FC2 derived descriptors implies CNN_1 has not only learned strong pose discrimination but that we can use a truncated form of the resulting network to produce a descriptor for pose estimation.



(a) DFT+SVM (b) DA1 (c) DA2 (d) DA2+SVM

Figure 4: Class wise classification confusion matrix, for the four descriptor options on the *ROM* dataset)

4.3 Pose Estimation Experiments

4.3.1 Joint Manifold Embedding Results

The optimal descriptor (4k CNN_1+DA2) learned on the *ROM* dataset for the purpose of pose classification is now applied, in conjunction with manifold learning, as the basis for markerless pose estimation. We applied the three proposed manifold learning techniques and CNN_2's direct regression of the pose to the hybrid *Ballet* MVV dataset to estimate the angles/locations of the joints of the pose. The dataset was split so that 4 of the 5 MVV sequences in Ballet (Sec. 4.1) were trained on with the remainder used for testing ($\sim$ 1400 frames). We perform comparative evaluation of proposed non-linear P_{nle} and the direct regression P_reg approaches with the baselines P_{nn}, and P_{ple}. We explore variations on the CNN architectures with the FC2 layer at **1k** and **4k** dimensionality and assess the importance of the proposed dynamic thresholding approach to improving cross-dataset encoding of the log-polar representation from the PVH, regardless of appearance change (e. g. differences in performer). Table 2 shows the average per joint pose estimation error in quaternion angles (degrees) and overall displacement in joint location (mm) of the four approaches (only joint position has been computed for P_{reg}), for different descriptor dimensionality and with or without the dynamic thresholding (**DynThrs**).

The use of a dynamic threshold on the representation uniformly reduces the error both regarding the joint angle and joint location. This is to be expected given the diversity of data used to train both the CNN and manifold learning (i. e. variation in illumination, noise due to the quantity and quality of cameras, and performer appearance).

Without appropriate data scaling via this method, the log-polar representation tends to encode poorly extremities of the performer containing expressive arm and leg movements. Figure 5 illustrates this problem and the difficulty of manually applying a threshold or corrective scaling to the log-polar data versus our dynamic technique.

Although a general trend rewarding higher dimensionality is observed, this is not true for P_{nle} where a dimensionality of 1k (with dynamic thresholding) proves to be the best performing configuration. Direct regression of pose using CNN_2, P_{reg}, performs better than any other technique, even without dynamic threshold applied to the input. The best configuration, **4K+DynThrs**, significantly outperforms the other methods. Figure 6 quantifies per frame error for each of the techniques over this best-performing descriptor. It shows direct CNN regression is more accurate on average, but poorer in terms of temporal consistency compared to other methods, with non-linear embedding, P_{nle} the strongest in this regard. Figure 7 provides qualitative comparisons via representative examples of pose estimates from each of the approaches.

Experiments were also undertaken using the learned weights from CNN_1 for fine-tuning CNN_2 in its regression training process, although negligable difference in final accuracy was achieved.

5. CONCLUSION

We have presented and evaluated a system for human pose estimation from multiple view-point video that demonstrates the potential for deep learning within the field. Our experiments show that we are able to train a CNN on volumetric data to discriminate between a range of poses covering the natural range of human motion. The affine-invariant pose descriptor extracted from the trained CNN performs well in both pose classification and estimation. Furthermore, we demonstrated that a similar CNN architecture can be used for direct regression of pose with significantly improved accuracy.

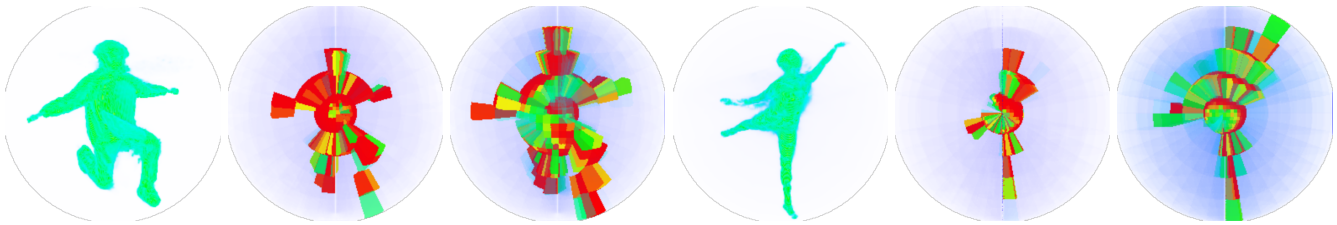
The direction for future work will include investigating further CNN models and fine-tuning strategies, and the fusion of additional sources of data that may compensate for the limitations of our silhouette-based representation, for example using inertial sensors to detect on-axis limb rotations and other subtle movements. We also intend to explore the benefit of using explicit kinematic constraints to compliment the implicit prior built into the trained system.

Acknowledgements

The work was supported by the REFRAME project, InnovateUK grant agreement 101854. The Ballet dataset is courtesy of the EU FP7 RE@CT project.

6. REFERENCES

[1] A. Agarwal and B. Triggs. 3D human pose from silhouettes by relevance vector regression. In *Proc. CVPR*, 2004.



(a) *ROM* Seq. frame (b) Manual Thresh. (c) Dynamic Thresh. (d) *Ballet* Seq frame (e) Manual thresh. (f) Dynamic thresh.

Figure 5: Examples of log-polar representations from *ROM* and *Ballet* sequences without dynamic thresholding (a-b) and with (c-d). Voxel colour key given in Fig. 2.

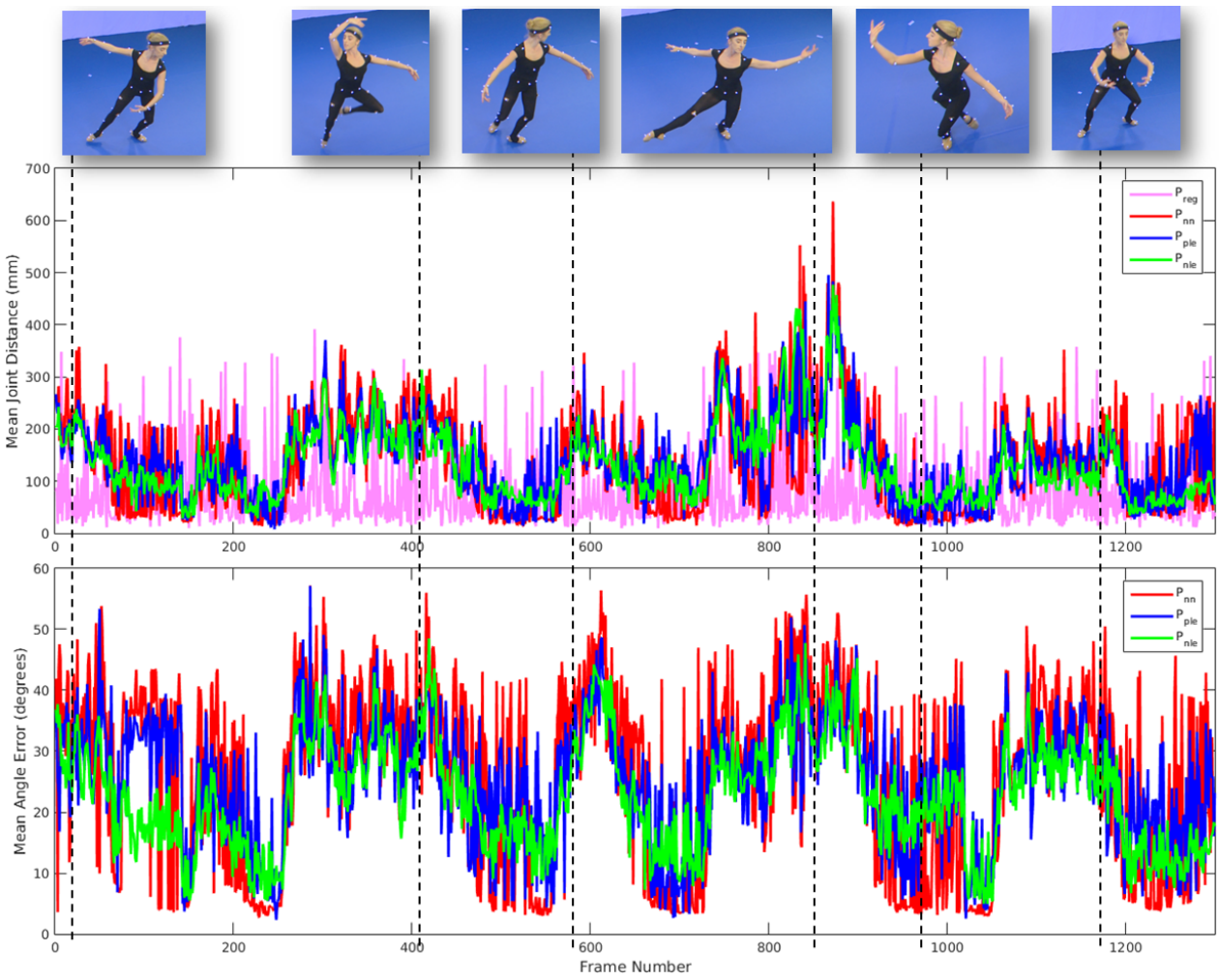


Figure 6: Comparing total joint position and angular error of the three manifold embedding techniques and the direct regression of CNN_2 over the *Ballet* dataset.

[2] A. Aggarwal, S. Biswas, S. Singh, S. Sural, and A. Majumdar. Object tracking using background subtraction and motion estimation in MPEG videos. In *Proc. Asian Conf. on Computer Vision (ACCV)*, volume 3852. Springer LNCS, 2006.

[3] M. Andriluka, S. Roth, and B. Schiele. Pictoral structures revisited: People detection and articulated pose estimation. In *Proc. Computer Vision and Pattern Recognition*, 2009.

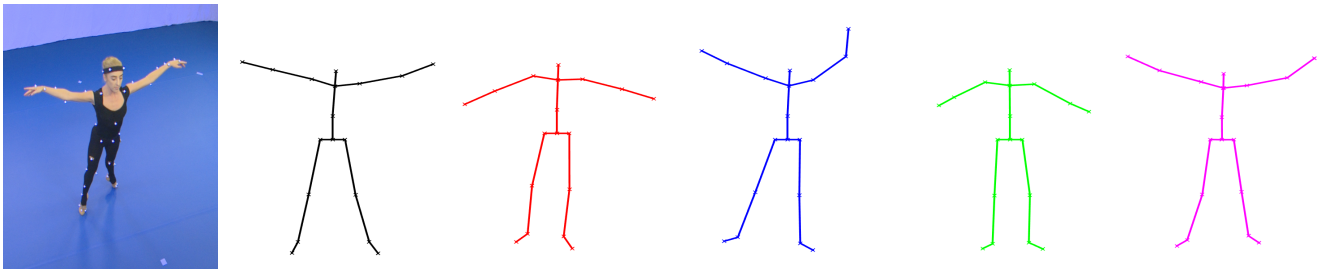
[4] M. Eichner and V. Ferrari. Better appearance models for pictorial structures. In *Proc. British Machine Vision Conf. (BMVC)*, 2009.

[5] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object detection. *Intl. Journal on Computer Vision*, 61, 2003.

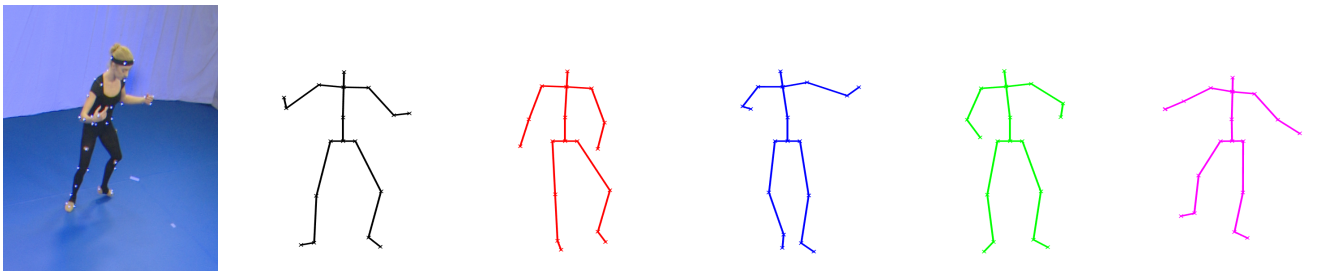
[6] K. Grauman, G. Shakhnarovich, and T. Darrell. A

Table 2: Pose estimation error (Average angular error in degrees and joint location error in millimetres)

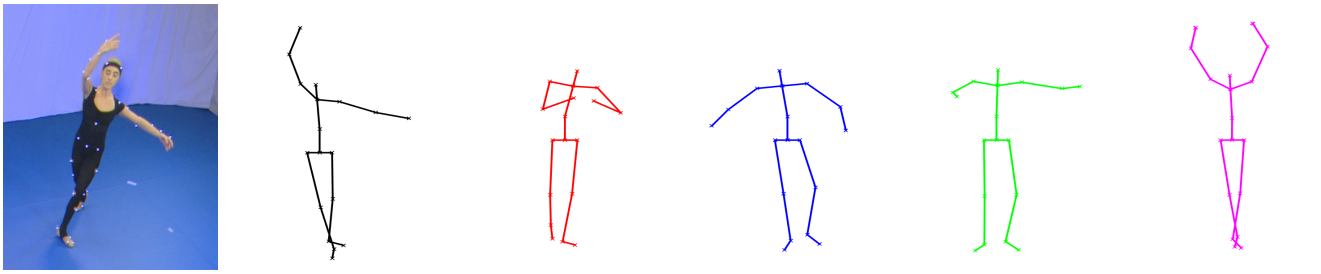
Descriptor	P_{nn}		P_{ple}		P_{nle}		P_{reg}
	Angle Err	Joint Err	Angle Err	Joint Err	Angle Err	Joint Err	Joint Err
1K	26.1	154.5	25.3	152.6	22.8	139.2	121.3
4K	23.2	136.1	22.9	140.1	21.4	132.2	120.2
1K+DynThrs	22.5	128.1	21.2	124.5	20.3	122.9	79.2
4K+DynThrs	21.7	124.9	21.5	127.5	20.8	129.8	**78.5**



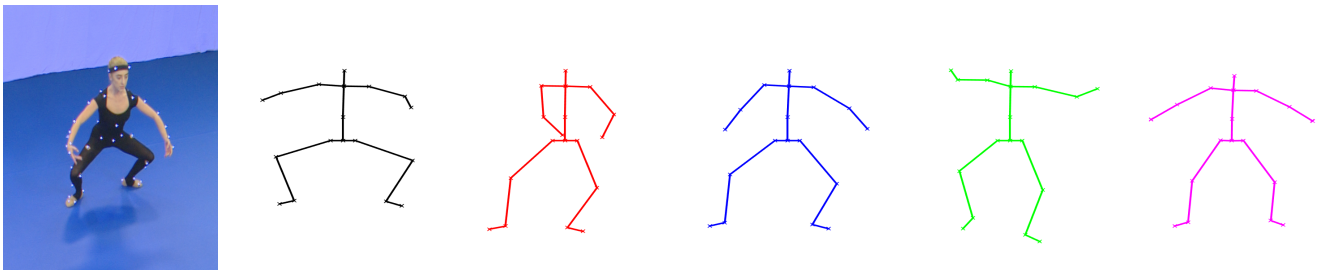
(a)



(b)



(c)



(d)

Figure 7: Representative examples of source data and corresponding pose estimations. From left to right: Source MVV frame; Ground truth (Vicon); Nearest-neighbour (P_{nn}); Piecewise linear embedding (P_{ple}); Non-linear GP-LVM embedding (P_{nle}); Direct CNN_2 regression (P_{reg}). Frames sampled at a) 439, b) 633, c) 755 and d) 1186 from *Ballet*.

bayesian approach to image-based visual hull reconstruction. In *Proc. CVPR*, 2003.

[7] Y. G. H. Ning, W. Xu and T. Huang. Discriminative learning of visual words for 3D human pose estimation. In *Proc. CVPR*, 2008.

[8] P. Huang, A. Hilton, and J. Starck. Shape similarity for 3d video sequences of people. *Intl. Journal of Computer Vision*, 2010.

[9] P. Huang, M. Tejera, J. Collomosse, and A. Hilton. Hybrid skeletal-surface motion graphs for character animation from 4d performance capture. *ACM Transactions on Graphics (ToG)*, 2015.

[10] H. Jiang. Human pose estimation using consistent max-covering. In *Intl. Conf. on Computer Vision*, 2009.

[11] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. NIPS*, 2012.

[12] X. Lan and D. Huttenlocher. A unified spatio-temporal articulated model for tracking. In *Proc. Computer Vision and Pattern Recognition*, volume 1, pages 722–729, 2004.

[13] X. Lan and D. Huttenlocher. Beyond trees: common-factor model for 2d human pose recovery. In *Proc. Intl. Conf. on Computer Vision*, volume 1, pages 470–477, 2005.

[14] N. Lawrence. Probabilistic non-linear principal component analysis with gaussian process latent variable models. *Journal of Machine Learning Research*, 6:1783–1817, 2005.

[15] A. Makadia and K. Daniilidis. Spherical correlation of visual representations for 3d model retrieval. *Intl. Journal of Computer Vision*, 2009.

[16] G. Mori, X. Ren, A. Efros, and J. Malik. Recovering human body configurations: Combining segmentation and recognition. In *Proc. Computer Vision and Pattern Recognition*, pages 326–333, 2004.

[17] D. Park and D. Ramanan. Articulated pose estimation with tiny synthetic videos. In *Proc. CHA-LEARN Workshop on Looking at People*, 2015.

[18] C. E. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[19] R. Ren and J. Collomosse. Visual sentences for pose retrieval over low-resolution cross-media dance collections. *IEEE Transactions on Multimedia*, 2012.

[20] X. Ren, E. Berg, and J. Malik. Recovering human body configurations using pairwise constraints between parts. In *Proc. Intl. Conf. on Computer Vision*, volume 1, pages 824–831, 2005.

[21] P. Srinivasan and J. Shi. Bottom-up recognition and parsing of the human body. In *Proc. Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[22] A. Toshev and C. Szegedy. Deep pose: Human pose estimation via deep neural networks. In *Proc. CVPR*, 2014.

[23] P. Viola and M. Jones. Robust real-time object detection. *Intl. Journal of Computer Vision*, 2(57):137–154, 2004.

[24] L. Wei, Q. Huang, D. Ceylan, E. Vouga, and H. Li. Dense human body correspondences using convolutional networks. *CoRR*, abs/1511.05904, 2015.

[25] P. Wohlhart and V. Lepetit. Learning descriptors for object recognition and 3d pose estimation. In *Proc. CVPR*, 2015.

[26] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3D shapenets: A deep representation for volumetric shapes. In *Proc. CVPR*, 2015.

[27] T. Zhao and R. Nevatia. Bayesian human segmentation in crowded situations. In *Proc. Computer Vision and Pattern Recognition*, volume 2, pages 459–466, 2003.