Unsupervised Motion Retargeting for Human-Robot Imitation

Louis Annabi, Ziqi Ma, Sao Mai Nguyen

Unsupervised Motion Retargeting for Human-Robot Imitation

Louis Annabi
U2IS, ENSTA Paris, IP Paris
Palaiseau, France
louis.annabi@gmail.com

Ziqi Ma
U2IS, ENSTA Paris, IP Paris
Palaiseau, France
ziqi.ma@ensta-paris.fr

Sao Mai Nguyen
U2IS, ENSTA Paris, IP Paris
IMT Atlantique, Lab-STICC, UMR
CNRS 6285
France
nguyensmai@gmail.com

ABSTRACT

This early-stage research work aims to improve online human-robot imitation by translating sequences of joint positions from the domain of human motions to a domain of motions achievable by a given robot, thus constrained by its embodiment. Leveraging the generalization capabilities of deep learning methods, we address this problem by proposing an encoder-decoder neural network model performing domain-to-domain translation. In order to train such a model, one could use pairs of associated robot and human motions. Though, such paired data is extremely rare in practice, and tedious to collect. Therefore, we turn towards deep learning methods for unpaired domain-to-domain translation, that we adapt in order to perform human-robot imitation.

CCS CONCEPTS

• **Computer systems organization** → **Robotics**.

KEYWORDS

imitation, neural networks, motion retargeting

ACM Reference Format:
Louis Annabi, Ziqi Ma, and Sao Mai Nguyen. 2024. Unsupervised Motion Retargeting for Human-Robot Imitation. In *Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction (HRI '24 Companion), March 11–14, 2024, Boulder, CO, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3610978.3640588

1 INTRODUCTION

In many human-robot interaction scenarios, robots need to be able to imitate human motions [3, 20]. For example, imitating human motions can be used to reproduce human demonstrated motions [9, 23], for coordination purposes in human-robot collaboration [26], or even to provide feedback to patients in physical rehabilitation scenarios [4]. This imitation is not a simple one-to-one mapping from human joint angle to motor angle, as the embodiment of humans and robots differ in sizes, proportions, velocities, forces, dynamics. Finding this relational mapping is referred to as the *correspondence problem* [19]. While the correspondence problem between a single demonstrator and an imitator has been addressed in the human-robot interaction and the machine learning literature, the correspondence problem is the more acute when the robot needs to interact with different people. To our knowledge, a humanoid robot solving the correspondence problem of whole body movements from different demonstrators carrying out several tasks has not been addressed. While humanoid robots and different demonstrators can have the same skeletal structure as humans, their bone length and joint amplitudes vary from person to person. One difficulty in addressing this problem is the lack of complete dataset with all demonstrators carrying out the same set of tasks and synchronized and paired with a robot execution of the same set of tasks.

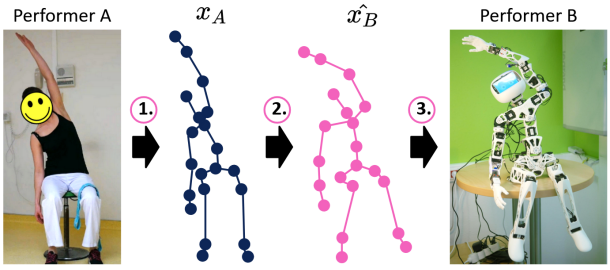

Figure 1: The steps of the human-robot imitation process: (1) pose estimation outputs from a video from performer A a sequence of joint positions x_A, (2) motion retargeting translates joint positions x_A into joint positions x_B for performer B, (3) robot control sends the low level control.

We separate the motion imitation process in three steps: pose estimation, motion retargeting, and robot control, as represented in figure 1. Pose estimation algorithms predict a sequence of skeleton joint positions from the human demonstrator given a sensor input. The motion retargeting step translates this sequence of joint positions towards a domain of joint positions achievable by the robot, i.e. from the human embodiment to the robot embodiment spaces. Finally, one can use these sequences of joint positions as targets for motor control (e.g. dynamic movement). While there is a flourishing literature on pose estimation, few works have addressed the question of motion retargeting for human-robot imitation. In this work, we will examine the correspondance problem in terms of bone length and flexibility, while considering that the skeletal structure is identical.

2 RELATED WORK

2.1 Whole-body imitation

Indeed, research in whole-body imitation of a human by a humanoid robot have proposed offline or real-time optimizations to

HRI '24 Companion, March 11–14, 2024, Boulder, CO, USA

ACM ISBN 979-8-4007-0323-2/24/03
https://doi.org/10.1145/3610978.3640588

solve inverse kinematics for the end effectors [11, 12, 18]. In [12] and [18], the authors used mainly inverse kinematics on each of the kinematic chains for static pose mapping. [11] solved the geometric and dynamic differences of the correspondence problem [19] by scaling by a predefined constant for the difference in length for body parts, and by optimizing inverse kinematics. In summary, they only focused on the end effectors as the key parts to be imitated, and supposed the retargeting is only about the end-effector positions. However, while this might be true for manipulation and task-oriented movements, in other cases, such as for rehabilitation exercises as in [8], the focus might shift on the intermediate joints. In order to take into consideration all the joints, another method is to use joint orientations as features that should be invariant to the motion performer. However, by simply copying the joint orientations, we may not properly translate meaningful features of the source motion. For instance, if the source motion contains a contact between the two hands (e.g. clapping hands), and if the target skeleton has shorter forearms than the source skeleton, the hands in the translated motion would not touch. These limitations have encouraged us to explore the use of deep learning methods for motion retargeting.

2.2 Motion retargeting

As retargeting can be seen as translating a motion from the human embodiment domain to the robot embodiment domain, it entails extracting the main characteristics from a motion that would be common to human and robot movements, or in other words, finding a common representation between both movements. Domain-to-domain translation is fundamentally a disentanglement problem, where some information content of the source data must be kept while some domain-specific content has to be transformed. For images, typically object positions and features should be kept, while appearance and style is transformed. [7] proposed shared latent variables as a common representation between movements from a robot and humans with different joint flexibility. However their algorithm does not handle more morphology differences, and needs to be retrained for each exercise : the same model can not be applied to all movements. In the case of motion retargeting, we would like to disentangle the motion features from the identity of the subject performing the motion. Thus, we hypothesize that a deep learning encoder-decoder architecture may capture such relevant features during the encoding, and properly translate them in the reconstructed motion during the decoding. In order to have a retargeting method that can work with different sources (e.g. different human subjects) and targets (e.g. different robots), we make the encoder invariant with regard to the source skeleton, and the decoder conditioned on the target skeleton. Training such an architecture with paired source and target motions would constitute an ideal supervised learning scenario, but in practice such data is very difficult to collect. One could obtain paired motions in circumstances where different subjects perform the same task (e.g. from action recognition data sets), however it is difficult to ensure that the task is completed with the same motion by different subjects. For instance, for two motions labeled as "picking up a phone", the hand picking up the phone (left or right), the speed of the motion, the initial position of the phone, the pose of the subject (sitting or standing) are features that may not be consistent.

The other option is to work with unpaired data. In this case, we have access to one dataset of motions for the source skeleton and one dataset of motions for the target skeleton, without any known correspondence. Models working with unpaired data have been proposed in the field of image-to-image translation [30], which we try to adapt in this work to perform unpaired cross-domain motion translation between several source and target embodiments.

In order to train a domain-to domain translation without paired data, the CycleGAN method [30] leverages a cycle-consistency constraint together with adversarial training. The cycle-consistency criterion ensures that translating back the prediction in the source space yields the initial source image (or motion in our case). The adversarial training constrains the prediction to belong to the target domain. In the UNIT method presented in [14], the authors use a similar approach with an encoder-decoder architecture. Using the assumption that both source and target domains share a same latent space, they build a cycle-consistency criterion in the latent space (output of the encoder). Another recent approach [21] replaces this cycle-consistency constraint using contrastive learning techniques. While these methods have been tested successfully for image-to-image translation, our goal is to adapt them to motion retargeting.

In order to use these methods for motion retargeting, we need to build neural network models that can properly process skeleton data. We can take inspiration from the literature in skeleton-based action recognition, which has seen several breakthroughs over the last decade. Initially centered around recurrent and convolutional neural networks [10, 13], the field increasingly uses graph convolutions [28]. Graph convolutions constitute a more natural operation to perform on skeleton data as they allow to properly exploit their underlying structure. Many variants have been proposed, using directed skeleton graphs, using wider kernel size convolutions with k-adjacency matrices, or using learnable graph adjacency matrices [15, 24]. Performing even better are neural network models using self-attention mechanisms, that can be seen as a special case of graph convolutions where the graph is dynamically constructed based on attention coefficients [22, 25].

In the motion retargeting literature, the second step of the human-robot imitation process we described is often considered lightly. Existing approaches typically focus on some joints of interest for which they define a transformation, for instance homothetic [29], or obtained by resizing the bones while keeping the joint angles invariant [6]. In comparison, our retargeting method has more freedom in the range of transformations it can apply to the source skeleton. Most related to our work is the deep motion retargeting model proposed in [1]. They also address the problem of unpaired cross-domain translation of motions. They use an architecture consisting of encoders, decoders and discriminators. However, their approach uses as input joint orientations (as quaternions) instead of positions, which has troublesome practical implications for deep learning [27], as well as struggles to capture some meaningful motion features such as joint contacts (e.g. in the clapping hand motion, hand contacts are not reflected in joint orientations information). Moreover, our current raw data are videos, which are processed by deep learning algorithms to extract joint positions [2, 5] with

Annabi, L., Ma, Z., and Nguyen, S. M. (2024). Unsupervised Motion Retargeting for Human-Robot Imitation. Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction(587–591). Association for Computing Machinery. https://doi.org/10.1145/3610978.3640588

performance comparable to to RGB-D cameras like the Kinect [17], whereas joint orientations are not directly obtained.

3 METHODS

In this section, we describe the retargeting algorithm. As depicted in figure 2, the algorithm consists of a pair of encoder-decoder to extract the main characteristics of a motion and to generate a motion in the target joint position space, and a discriminator to challenge the decoder. We then describe our training process.

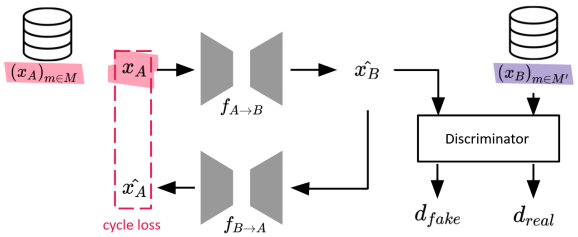

Figure 2: Algorithm architecture with an encoder-decoder and a discriminator.

3.1 Encoder and decoder

We present here the proposed encoder-decoder model for motion retargeting. The encoder takes as input x_A, a sequence of joint positions in the domain of motions performed by a performer A. It infers a latent variable z from this input motion, disentangling information about the motion itself from the information about the performer, as represented in figure 3. Then the decoder takes as input this latent variable z as well as the lengths of the performer B bones, that we denote l_B, and outputs a prediction $\hat{x_B}$ corresponding to the motion translated to the domain of motions performed by the performer B. In this work, we assume that source and target skeletons share the same structure, but that bone lengths vary between the two.

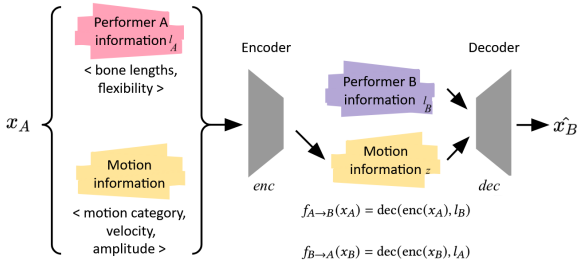

Figure 3: Details of the encoder-decoder.

The encoder and the decoder both comprise three layers of graph convolutions mixed with three layers of temporal 1D convolutions. The graph convolutions allow to process information on the skeleton dimension, while the temporal convolutions allow to process information on the temporal dimension.

For the encoder, the temporal convolutions use a kernel size of 3 and a stride of 2. For the decoder, we experimented with : option 1: transposed convolutions; option 2: using upsampling with a factor of 2, followed by convolutions of kernel size 3 and stride 1. While the transposed convolutions seem at first like the right choice to have a symmetrical architecture for the encoder and the decoder, it can create artifacts on the generated data, that we do not observe using the second option.

For the encoder, we use standard directed graph convolutions, where we have different sets of weights for the parent nodes and for the children nodes (denoted respectively W_p and W_c). The graph convolution performs the following operation on an input x of shape (N, d) where N is the number of nodes in the graph (here the number of joints), and d is the number of features for each node i:

$$x_i \leftarrow \sigma\Big(W_r \cdot x_i + \sum_{j\in C(i)} W_c \cdot x_j + \sum_{j\in \mathcal{P}(i)} W_p \cdot x_j\Big) \quad (1)$$

where $\mathcal{P}(i)$ and $C(i)$ denote respectively the parents and children of node i in the graph, σ is the sigmoid activation function, and W_r corresponds to a third set of weights. Bias coefficients are omitted for simplicity, here and in the rest of the methods section.

The decoder needs to include the bone lengths of the target performer as additional input. We choose to include those as edge features in the graph, and instead perform two-steps graph convolutions in the decoder. The first step updates edge features e_{ij} based on adjacent nodes' features x_i and x_j, and the second step updates node features based on parent and children edges' features, as expressed in the following equations:

$$e_{ij} \leftarrow \sigma\Big(W_e \cdot e_{ij} + W_- \cdot x_i + W_+ \cdot x_j\Big) \quad (2)$$

$$x_i \leftarrow \sigma\Big(W_r \cdot x_i + \sum_{j\in C(i)} W_c \cdot e_{ij} + \sum_{j\in \mathcal{P}(i)} W_p \cdot e_{ji}\Big) \quad (3)$$

where W_e, W_+, W_- denote sets of weights for the first step. This graph convolution operation makes it possible for the decoder to generate motion conditioned on the provided bone lengths.

3.2 Discriminator

The discriminator takes as input a predicted motion $\hat{x_B}$ and target performer bone lengths l_B and outputs a real number scoring how well the predicted motion fits the distribution of target motions. It is trained with adversarial training, using positive real samples from the distribution $\mathcal{D}_B$ and fake samples generated by the encoder-decoder network. Its structure is similar to the encoder, with an additional dense layer at the end to output the score. It combines temporal 1D convolutions and two-steps graph convolutions to include the bone lengths l_B in the processing.

3.3 Training

Following the Least Squares GAN method [16], the discriminator is trained with the following loss function:

$$\mathcal{L}_D = \frac{1}{2}\mathbb{E}_{x_B\sim\mathcal{D}_B}\Big[\big(D(x_B)-1\big)^2\Big] + \frac{1}{2}\mathbb{E}_{x_A\sim\mathcal{D}_A}\Big[D\big(f_{A\to B}(x_A)\big)^2\Big] \quad (4)$$

where D denotes the discriminator, and $f_{A\to B}(x_A) = \mathrm{dec}(\mathrm{enc}(x_A), l_B)$ the encoder-decoder network translating from subject A to B.

We train the encoder-decoder network (working as the generator as well) with a combination of several loss functions, following the CycleGAN [30] and UNIT [14] methods for unpaired domain-to-domain translation. Both methods use a loss function coming from the adversarial training:

$$\mathcal{L}_G = \frac{1}{2}\mathbb{E}_{x_A\sim\mathcal{D}_A}\Big[\big(D\big(f_{A\to B}(x_A)\big)-1\big)^2\Big] \quad (5)$$

Annabi, L., Ma, Z., and Nguyen, S. M. (2024). Unsupervised Motion Retargeting for Human-Robot Imitation. Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction(587–591). Association for Computing Machinery. https://doi.org/10.1145/3610978.3640588

The cycleGAN model combines it with a cycle consistency loss:

$$\mathcal{L}_{cycle} = \mathbb{E}_{x_A \sim \mathcal{D}_A}\left[\left\|x_A - f_{B\to A}(f_{A\to B}(x_A))\right\|_2^2\right] \quad (6)$$

where $f_{B\to A}(x_B) = \text{dec}(\text{enc}(x_B), l_A)$ denotes the encoder-decoder network translating from subject B to subject A.

In comparison, the UNIT model combines it with a variational auto-encoder loss and a cycle consistency loss on the latent space:

$$\mathcal{L}_{vae} = \mathbb{E}_{x_A \sim \mathcal{D}_A}\left[\left\|x_A - \text{dec}(\text{enc}(x_A), l_A)\right\|_2^2 + \left\|\text{enc}(x_A)\right\|_2^2\right] \quad (7)$$

$$\mathcal{L}_{cycleUNIT} = \mathbb{E}_{x_A \sim \mathcal{D}_A}\left[\left\|\text{enc}(x_A) - \text{enc}(f_{A\to B}(x_A))\right\|_2^2\right] \quad (8)$$

where enc and dec denote the encoder and decoder networks.

We experiment with both methods, and train our models on a dataset of animated motions called Mixamo (https://www.mixamo.com/). This dataset advantageously contains the same motions performed by different animated characters. We create a training and testing set from motions exported from the website, with 800 motions distributed among 25 characters for the training set (unpaired data), and 110 same motions for 4 other characters in the test set. The Mixamo dataset comprise fbx files, a type of 3D model file containing mesh, material, texture, and skeletal animation data, which can be easily 3d joint positions.

With this separation, we ensure that the training data do not contain any motion performed by two different characters, and on the contrary, that testing data contain corresponding motions for different characters, which allows us to compute a prediction error. However, the motions are automatically adapted to the animated characters and the implementation of this adaptation is not available. We can see that some motions present impossible body configurations (e.g. arm going through head). Consequently, we will also perform visual inspection in order to validate our method.

4 EARLY RESULTS

We present here early results. Figure 4 displays an example motion x_A, the corresponding predicted motion obtained with the UNIT model after training $\hat{x_B}$, and the ground truth motion performed by performer B taken from the Mixamo test set x_B. Figure 4 only shows one time frame of the full motion.

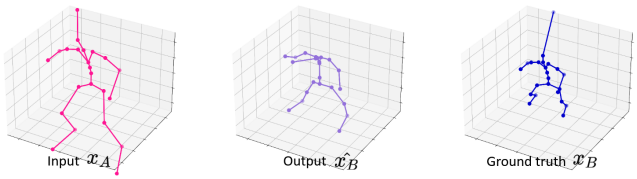

Figure 4: Example of motion retargeting using UNIT.

While we can observe that the predicted skeleton is of comparable size with the ground truth skeleton (which indicates that there is to some extent a translation of the motion to the target domain), the pose is visually very different from the ground truth.

We measure the average retargeting prediction error on the test set using the two methods described above, and compare them with two simple baselines consisting respectively in copying the source joint positions, and copying the source joint orientations (and computing the joint positions using the target skeleton bone lengths). The results are displayed in table 1.

Table 1: Comparison of the different retargeting methods.

Method	Reconstruction error (train)	Reconstruction error (test)	Retargeting error (test)
Position copy	0 mm	0 mm	195 mm
Rotation copy	0 mm	0 mm	**79 mm**
CycleGAN	70 mm	182 mm	243 mm
UNIT	48 mm	164 mm	209 mm

The results argue so far against our initial hypothesis that deep learning methods can successfully perform unpaired motion retargeting, as a simple method copying joint orientations (rotation copy in the table) can better retarget motions.

More experiments will be conducted in order to properly identify the causes of this failure. While unsupervised domain-to-domain translation has been successfully applied to images, the mechanisms behind this success remain unclear. Indeed, the unsupervised domain-to-domain translation constitutes an ill-posed problem, where many solutions can satisfy the criteria we optimize for, while not performing the translation we expect. For instance, an image-to-image style translation problem in an unsupervised setting could in principle accept as a solution a function $f_{A\to B}$ correctly applying the desired style translation $f^*_{A\to B}$, but also applying at the same time a bijection b on the image space for which the style is invariant. The candidate solutions $f_{A\to B} = f^*_{A\to B} \circ b$ and $f^*_{A\to B}$ are as good at minimizing the loss function, yet only the later is the solution we wish to reach. Investigating this problem, and how it was addressed for image-to-image translation, could help us find better network designs or initialization strategies leading to the desired solution.

5 CONCLUSION

This early-stage research has shown that deep learning unsupervised motion retargeting is feasible yet not accurate enough to replace simpler naive methods. Still, as explained in the introduction, such naive methods are limited in that they cannot capture and translate some meaningful features of the motion, and we think that more effort is needed to improve motion retargeting models.

Future work will extend the current study in three directions:

- Further investigating the failure of the current method, as explained in the last section.
- Creating a dataset of paired motion data from human-human imitation or robot-human imitation. We hypothesize that humans can perform accurate imitation, and thus participate in building a dataset of paired motions. A first step would be to have enough paired data to replace the Mixamo test set. In a second step, if enough paired data is available, it might become possible to train the retargeting models in a supervised learning setting, largely simplifying the problem.
- Improving the model architecture in order to obtain more accurate retargeting predictions. As hinted by their success in the field of action recognition, skeleton self-attention layers, as well as temporal self-attention layers (transformers) could also help our model better capturing important features of the motion and generating accurate predictions.

ACKNOWLEDGMENTS

This project is partially funded by Institut Carnot and AID Project ACoCaTherm.

Annabi, L., Ma, Z., and Nguyen, S. M. (2024). Unsupervised Motion Retargeting for Human-Robot Imitation. Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction(587–591). Association for Computing Machinery. https://doi.org/10.1145/3610978.3640588

REFERENCES

[1] Kfir Aberman, Peizhuo Li, Dani Lischinski, Olga Sorkine-Hornung, Daniel Cohen-Or, and Baoquan Chen. 2020. Skeleton-aware networks for deep motion retargeting. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 62–1.

[2] Valentin Bazarevsky, Ivan Grishchenko, Karthik Raveendran, Tyler Zhu, Fan Zhang, and Matthias Grundmann. 2020. BlazePose: On-device Real-time Body Pose tracking. *CoRR* abs/2006.10204 (2020).

[3] Aude Billard and R. Siegward. 2004. Robot Learning from Demonstration. *Robotics and Autonomous Systems* 47, 2-3 (June 2004), 65–67.

[4] Agathe Blanchard, Sao Mai Nguyen, Maxime Devanne, Mathieu Simonnet, Myriam Le Goff-Pronost, and Olivier Rémy-Néris. 2022. Technical Feasibility of Supervision of Stretching Exercises by a Humanoid Robot Coach for Chronic Low Back Pain: The R-COOL Randomized Trial. *BioMed Research International* 2022 (mar 2022), 1–10.

[5] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. 2019. OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2019).

[6] Sungjoon Choi and Joohyung Kim. 2019. Towards a natural motion generator: A pipeline to control a humanoid based on motion data. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 4373–4380.

[7] Maxime Devanne and Sao Mai Nguyen. 2019. Generating Shared Latent Variables for Robots to Imitate Human Movements and Understand their Physical Limitations. In *Computer Vision – ECCV 2018 Workshops*, Laura Leal-Taixé and Stefan Roth (Eds.). Springer International Publishing, Cham, 190–197.

[8] Maxime Devanne, Sao Mai Nguyen, Olivier Remy-Neris, Béatrice Le Gales-Garnett, Gilles Kermarrec, and André Thepaut. 2018. A Co-design Approach for a Rehabilitation Robot Coach for Physical Rehabilitation Based on the Error Classification of Motion Errors. In *IEEE International Conference on Robotic Computing (IRC)*. 352–357.

[9] Auke Jan Ijspeert, Jun Nakanishi, and Stefan Schaal. 2002. Movement imitation with nonlinear dynamical systems in humanoid robots. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, Vol. 2. IEEE, 1398–1403.

[10] Qiuhong Ke, Mohammed Bennamoun, Senjian An, Ferdous Sohel, and Farid Boussaid. 2017. A new representation of skeleton sequences for 3d action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3288–3297.

[11] Seungsu Kim, ChangHwan Kim, Bumjae You, and Sangrok Oh. 2009. Stable whole-body motion generation for humanoid robots to imitate human motions. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2518–2524. https://doi.org/10.1109/IROS.2009.5354271

[12] Jonas Koenemann, Felix Burget, and Maren Bennewitz. 2014. Real-time imitation of human whole-body motions by humanoids. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*. 2806–2812. https://doi.org/10.1109/ICRA.2014.6907261

[13] Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang. 2016. Spatio-temporal lstm with trust gates for 3d human action recognition. In *European conference on computer vision*. Springer, 816–833.

[14] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. 2017. Unsupervised image-to-image translation networks. *Advances in neural information processing systems* 30 (2017).

[15] Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. 2020. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 143–152.

[16] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. 2017. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2794–2802.

[17] Aleksa Marusic, Sao Mai Nguyen, and Adriana Tapus. 2023. Evaluating Kinect, OpenPose and BlazePose for Human Body Movement Analysis on a Low Back Pain Physical Rehabilitation Dataset. In *Companion of the 2023 ACM/IEEE International Conference on Human-Robot Interaction (HRI '23)*. Association for Computing Machinery, New York, NY, USA, 587–591.

[18] S. Nakaoka, A. Nakazawa, K. Yokoi, H. Hirukawa, and K. Ikeuchi. 2003. Generating whole body motions for a biped humanoid robot from captured human dances. In *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, Vol. 3. 3905–3910 vol.3. https://doi.org/10.1109/ROBOT.2003.1242196

[19] Chrystopher L. Nehaniv and Kerstin Dautenhahn. 2002. *The Correspondence Problem*. MIT Press, Cambridge, MA, USA, 41–61.

[20] Chrystopher L Nehaniv and Kerstin Dautenhahn. 2007. *Imitation and Social Learning in Robots, Humans and Animals: Behavioural, Social and Communicative Dimensions*. Cambridge Univ. Press, Cambridge.

[21] Taesung Park, Alexei A Efros, Richard Zhang, and Jun-Yan Zhu. 2020. Contrastive learning for unpaired image-to-image translation. In *European conference on computer vision*. Springer, 319–345.

[22] Chiara Plizzari, Marco Cannici, and Matteo Matteucci. 2021. Skeleton-based action recognition via spatial and temporal transformer networks. *Computer Vision and Image Understanding* 208 (2021), 103219.

[23] Stefan Schaal. 1997. Learning from Demonstration. In *Advances in Neural Information Processing Systems 9*, M. C. Mozer, M. I. Jordan, and T. Petsche (Eds.). MIT Press, 1040–1046.

[24] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. 2019. Skeleton-based action recognition with directed graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7912–7921.

[25] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. 2019. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 12026–12035.

[26] Likun Wang, Guoyan Wang, Shuya Jia, Alison Turner, and Svetan Ratchev. 2022. Imitation learning for coordinated human–robot collaboration based on hidden state-space models. *Robotics and Computer-Integrated Manufacturing* 76 (2022), 102310. https://doi.org/10.1016/j.rcim.2021.102310

[27] Sitao Xiang and Hao Li. 2020. Revisiting the continuity of rotation representations in neural networks. *arXiv preprint arXiv:2006.06234* (2020).

[28] Sijie Yan, Yuanjun Xiong, and Dahua Lin. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-second AAAI conference on artificial intelligence*.

[29] Haodong Zhang, Weijie Li, Yuwei Liang, Zexi Chen, Yuxiang Cui, Yue Wang, and Rong Xiong. 2021. Human-Robot Motion Retargeting via Neural Latent Optimization. *CoRR* (2021).

[30] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2223–2232.

Annabi, L., Ma, Z., and Nguyen, S. M. (2024). Unsupervised Motion Retargeting for Human-Robot Imitation. Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction(587–591). Association for Computing Machinery. https://doi.org/10.1145/3610978.3640588