
Intentional Dance Choreography with Semi-Supervised Recurrent VAEs

Mathilde Papillon *
UC Santa Barbara

Mariel Pettee
Lawrence Berkeley National Lab

Nina Miolane
UC Santa Barbara

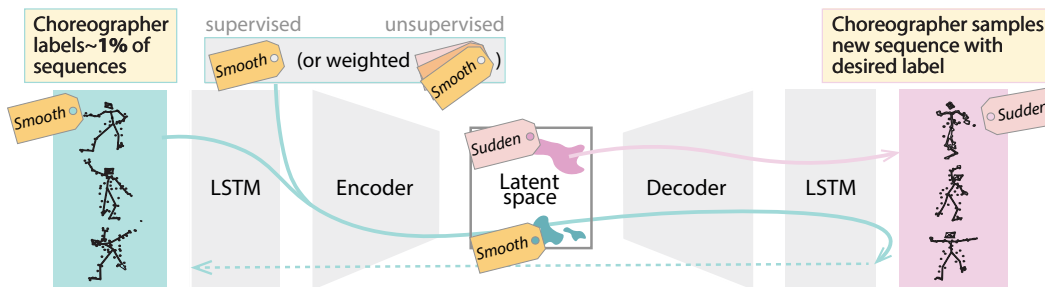


Figure 1: PIROUNET, a semi-supervised recurrent variational autoencoder, is trained (blue path) with a 1% manually labeled dataset. New dance sequences are conditionally generated (pink path) by sampling in the latent space conditionally on the desired choreographic label.

1 Introduction

Artificial Intelligence (AI) has given rise to a suite of methods that automatically generate new dance sequences. Yet, these tools continue to be largely ignored by dance practitioners because they lack the ability to create dance sequences with specific choreographic aesthetic. Two broad categories describe how these AI methods engage with an artist’s choreographic practice. The first consists of deep learning approaches that often rely on recurrent networks [1] and (variational) autoencoders [2]. These act as generators for new movement ideas, either by randomly generating sequences [3, 4, 5, 6] or by responding to a specific movement [5, 6, 7] or music prompt [8, 9, 10, 11]. Yet, the user’s creative control over these sequences is restricted to the choice of training data or of an external prompt. The second category, made up of algorithmic and non-deep methods [12, 13], makes use of Laban Movement Analysis (LMA) [14, 15], a widely recognized dance theory [16], to translate choreographic scores directly into movement. Beyond providing a static starting point for inspiration [17], this category’s methods are limited in their creative contribution. We postulate that the lack of large, user-specific dance datasets with labels meaningful to the user’s practice has restricted the development of dance generation methods with substantial creative control. Such datasets for daily human actions [18] have enabled supervised action generation methods [19, 20]. By contrast, existing annotated dance databases [21, 22, 23] are small and limited to their producers’ specific styles and creative processes.

To remedy this, we propose PIROUNET, a semi-supervised generative recurrent deep learning model that conditionally creates new dance sequences from choreographers’ aesthetic inputs. Our semi-supervised approach, combined with a suite of tools for automatic labeling, ensures that the choreographer only needs to label a very small portion of an input dataset. While we use the categorical intensities of LMA’s Laban Time Effort [24, 25, 26] as an illustrative example, PIROUNET users can implement any choice of subjective labels they wish to use.

*Corresponding email: papillon@ucsb.edu

2 Methods

PIROUNET (github.com/bioshape-lab/pirounet) features a dance encoding and generative model that uses (i) a variational autoencoder (VAE) [inspired from [6]], coupled with (ii) motion dynamics through a long-short-term-memory network [7]. For semi-supervised training [8], PIROUNET leverages (iii) a linear classifier. We propose the following conditional dance generative model: $p(y) = \text{Cat}(y | j)$; $p(z) = N(z | j, 0; I)$; $p(x | y, z; \theta) = N(f(y, z; \theta)^2)$. Here, z is a continuous variable representing the dynamics, which movement is performed, y is a categorical variable denoting the choreographer's label, how the move is performed. PIROUNET solves the inverse problem of inferring the marginally independent latent variables (from a sequence x).

Figure 1's blue path denotes the flow of information during training. In the supervised case, PIROUNET maximizes the log-likelihood $\log p(x; y)$ via the maximization of its lower bound $L(x; y)$, which resembles a VAE's regularized reconstruction loss (see App. 3). In the unsupervised case, y is missing and treated as another latent variable, in addition, over which we perform posterior inference. We maximize the marginalized log-likelihood $\log p(x)$ via the maximization of its lower bound (see App. 4). This consists of an entropy term $h(x; y)$ weighted by the classifier's confidence associated to each label. Fig. 1's pink path shows how PIROUNET conditionally generates dance. A new random latent variable representing body motion, is sampled from an approximation of the marginal distribution $q(z|y)$ where the label y is chosen by the user.

3 Results

After an extensive hyperparameter search (App. 1, 2), we validate PIROUNET on a 225-sequence benchmark set, partially depicted in Fig. 2 (see github.com/bioshape-lab/pirounet for video examples). The labeler identifies 96.0% of these sequences to be realistic and novel. Of these, 63% are determined to be in agreement with their intended label, or 83% of the labeler's self-agreement upon labeling the dataset twice. Further, this set features a diversity metric more than double that of test data. These results show PIROUNET's ability to perform as a creative tool, even with limited supervision.

Figure 2: New sequences generated by PIROUNET, demonstrating its dynamic creativity in adequation with the choreographer's desired intentions. a. Slow leg extension from V-sit. b. Continuous transition from crouched to plank position. c. Smooth weight transfer through deep plié. d. Forward weight transfer through small plié. e. Spontaneous drop to a crouched position. f. Leg swing leads sharp torso rotation. g. Pirouette with arm thrown up. h. Explosive extension of torso from plié.

4 Context within the creative practice

We hope this artist-guided, adaptable tool will act as a launching pad for engaging with previous work and developing vocabulary for new choreography, offering new insight into one's own choreographic vision. In parallel, identifying qualitative intentions on a spectrum (low to high, small to big, etc) offers paths towards exploring movement that contrasts with one's usual practice. Moreover, PIROUNET helps facilitate a conversation between old repertoire and new explorations. The artist is empowered to shape their own AI tool with a customizable keypoint labeling web-app that supports graphic-supported classification of any keypoint dataset with any annotations (see App. 5). While demonstrated on dance, the proposed method can be extended to other forms of art creation, inspiring AI-based tools tailored to the style and intuition of their artist.

5 Ethics

All training data was sourced ethically, originating from our second author. As the dataset only represents one body and one movement practice, it cannot validate PROOFNET for all dancers or practices. Moreover, it is important to underline that LMA was designed for and mostly used by the Western modern dance community, meaning the results presented should not be generalized to all forms of dance annotations. The open sourced nature of the project and low data barrier were designed to encourage many more dance practitioners to train their own version of PROOFNET, with or without LMA labels.

References

- [1] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.
- [2] Max Welling and Diederik P Kingma. Auto-encoding variational bayes. *ICLR*, 2014.
- [3] Alexander Berman and Valencia James. Kinetic imaginations: Exploring the possibilities of combining ai and dance. In *Proceedings of the 24th International Conference on Artificial Intelligence*, page 2431–2437. AAAI Press, 2015.
- [4] Zimo Li, Yi Zhou, Shuangjiu Xiao, Chong He, Zeng Huang, and Hao Li. Auto-conditioned recurrent networks for extended complex human motion synthesis, 2017.
- [5] Agnese Augello, Emanuele Cipolla, Ignazio Infantino, Adriano Manfre, Giovanni Pilato, and Filippo Vella. Creative robot dance with variational encoder, 2017.
- [6] Mariel Pettee, Chase Shimmin, Douglas Duhaime, and Ilya Vidrin. Beyond Imitation: Generative and Variational Choreography via Machine Learning, 2019.
- [7] Alexander Berman and Valencia James. Learning as Performance: Autoencoding and Generating Dance Movements in Real Time. *pages 256–266*. 01 2018.
- [8] Ruilong Li, Shan Yang, David A. Ross, and Angjoo Kanazawa. Ai choreographer: Music conditioned 3d dance generation with aist++, 2021.
- [9] Hsin-Ying Lee, Xiaodong Yang, Ming-Yu Liu, Ting-Chun Wang, Yu-Ding Lu, Ming-Hsuan Yang, and Jan Kautz. Dancing to music, 2019.
- [10] Omid Alemi, Jules Franoise, and Philippe Pasquier. Groovenet: Real-time music-driven dance movement generation using artificial neural networks. 08 2017.
- [11] Jiaman Li, Yihang Yin, Hang Chu, Yi Zhou, Tingwu Wang, Sanja Fidler, and Hao Li. Learning to generate diverse dance motions with transformer, 2020.
- [12] Worawat Choensawat and Kozaburo Hachimura. Generating stylized dance motion from labanotation by using an autonomous dance avatar. *GRAAPP/IVAPP*, 2012.
- [13] Shun Zhang, Qilei Li, Tao Yu, XiaoJie Shen, Weidong Geng, and Pingyao Wang. Implementation of a notation-based motion choreography system. *pages 495–503*, 10 2006.
- [14] Lynn Matluck Brooks. Harmony in Space: A Perspective on the Work of Rudolf Laban. *Journal of Aesthetic Education*, 27(2):29–41, 1993. Publisher: University of Illinois Press.
- [15] Ed Groff. Laban Movement Analysis: Charting the ineffable domain of human movement. *Journal of Physical Education, Recreation & Dance*, 66(2):27–30, 1995.
- [16] Vera Maletic. Body-Space-Expression: The development of Rudolf Laban's movement and dance concepts. volume 75. Walter de Gruyter, 2011.
- [17] Kristin Carlson, Thecla Schiphorst, and Philippe Pasquier. Scuddle: Generating Movement Catalysts for Computer-Aided Choreography. *ICCC*, pages 123–128, 2011.
- [18] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. NTU RGB+ D: A large scale dataset for 3d human activity analysis. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1010–1019, 2016.
- [19] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2Motion: Conditioned Generation of 3D Human Motion. *CoRR*, abs/2007.15240, 2020.

- [20] Mathis Petrovich, Michael J. Black, and Gül Varol. Action-Conditioned 3D Human Motion Synthesis with Transformer VAE, 2021.
- [21] Shuhei Tsuchida, Satoru Fukayama, Masahiro Hamasaki, and Masataka Goto. AIST Dance Video Database: Multi-Genre, Multi-Dancer, and Multi-Camera Database for Dance Information Processing. In *ISMIR*, volume 1, page 6, 2019.
- [22] Dohyung Kim, Dong-Hyeon Kim, and Keun-Chang Kwak. Classification of K-Pop dance movements based on skeleton information obtained by a Kinect sensor. *Sensors* 17(6):1261, 2017.
- [23] Katerina El Raheb and Yannis Ioannidis. Annotating the captured dance: reflections on the role of tool-creation. *International Journal of Performance Arts and Digital Media* 17(1):118–137, 2021.
- [24] E. Davies. Beyond dance: Laban's legacy of movement analysis. January 2006.
- [25] Vanessa Ewan and Kate Sagovsky. *Laban's Efforts in Action: A Movement Handbook for Actors with Online Video Resources*. Bloomsbury Publishing, 2018.
- [26] Heather Knight and Reid Simmons. Expressive motion with x, y and theta: Laban effort features for mobile robots. In *The 23rd IEEE international symposium on robot and human interactive communication* pages 267–273. IEEE, 2014.
- [27] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation* 9(8):1735–1780, 1997.
- [28] Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editor, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.
- [29] Meinard Müller, Tido Röder, Michael Clausen, Bernhard Eberhardt, Björn Krüger, and Andreas Weber. Documentation Mocap database HDM05. 2007.
- [30] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* pages 1653–1660, 2014.
- [31] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision* pages 483–499. Springer, 2016.
- [32] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. RMPE: Regional multi-person pose estimation. In *Proceedings of the IEEE international conference on computer vision* pages 2334–2343, 2017.
- [33] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, editor, *Advances in Neural Information Processing Systems* 32, pages 8024–8035. Curran Associates, Inc., 2019.
- [34] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, 2014.
- [35] Lukas Biewald. Experiment Tracking with Weights and Biases, 2020. Software available from wandb.com.
- [36] Vinod Nair and Geoffrey E. Hinton. Rectified Linear Units Improve Restricted Boltzmann Machines. In *ICML*, pages 807–814, 2010.
- [37] James M. Joyce. *Kullback-Leibler Divergence* pages 720–722. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
- [38] Shammamah Hossain, C Calloway, D Lipka, D Niederhut, and D Shupe. Visualization of bioinformatics data with Dash bio. In *Proceedings of the 18th Python in Science Conference* pages 126–133, 2019.

A Appendix

A.1 Datasets

Like most deep learning tools for dance, we leverage motion data in keypoint format. This format, often included in large movement datasets [18, 29], represents the body as a cloud of 3D points, each representing a unique joint. Available and high-performing pose-estimation software [30, 31, 32] makes keypoint format accessible to smaller, homemade datasets as well. In this spirit, we use half of Pettee's keypoint dataset [6].

The dataset [6] is comprised of 36,396 poses extracted from six uninterrupted dances captured at a rate of 35 frames per second. This amounts to about 20 minutes of real-time movement of an experienced dancer. Each pose features 53 joints captured in three dimensions, normalized such that the dance fits within inside a unit box. The dancer's barycenter is fixed to one point on the (x,y) plane. From the pose data, we extract 36,356 sliding sequences of 40 continuous poses, and manually label 350 of these sequences (0.96% of the dataset) which do not share any of the same poses. This takes an experienced dancer (the principal author) about 3 hours, identifying if the movement's Laban Time Effort (characterizing how sudden or urgent a movement feels) is Low, Medium, or High. Using our label augmentation toolkit (see App. 3), we apply two techniques to get 9,167 labeled sequences (representing 25.2% of our unlabeled dataset) in total, split between 45% Low, 34% Medium, and 21% High Efforts. (i) We automatically label all sequences between sequences that share a same Effort. For example, if two back-to-back sequences are deemed to have Low Time Effort, all sequences that are a combination of the poses in these two sequences are also labeled with Low Time effort. (ii) We extend every label to all sequences starting within 6 frames (0.17 seconds) before or after its respective sequence.

A.2 Training

PirouNet is built using the PyTorch library [33] and run on a server with two Nvidia A30 GPUs and two CPUs, each with 16 cores. We train using an Adam optimizer with standard hyperparameters [34]. We present results for the PirouNet architecture resulting from a hyperparameter search using Wandb [35] on batch size, learning rate, number of LSTM and dense layers, as well as hidden variable sizes. PirouNet uses 5 LSTM layers with 100 nodes in both the encoder and the decoder. The classifier features 2 ReLU-activated [36] linear layers with 100 nodes. The latent space is 256-dimensional, which is approximately 25 times smaller than the 6360-dimensional initial space. We train for 500 epochs with a learning rate of 10^{-4} and a batch size of 80 sequences. For unsupervised training, we use 35,538 40-pose sequences, with the remaining sequences being reserved for testing. For supervised training, PirouNet is trained on 79% (16.6% of entire training set) of the labeled sequences. We reserve 11% of the labeled sequences for validation, and 10% for testing.

A.3 Derivation of Lower Bound in the Labeled Case

This appendix presents a step-by-step derivation for the evidence lower bound (ELBO) used for training PirouNet on labeled input data. As shown below, the ELBO provides a computable quantity that is smaller or equal to the log-likelihood (which is, itself, intractable due to the integral on the hidden latent variables). The goal is to find the "most likely" parameters of our model, i.e. the parameters that maximize the model's log-likelihood. Instead of maximizing the (intractable) log-likelihood, we will maximize its ELBO.

In the case of input data that have labels, the log-likelihood is $\log p(x; y)$ by definition. Let $q(z|x; y)$ be the approximate posterior for the continuous latent variable z . The law of total probability helps us write the log-likelihood integrating on the hidden latent variable

$$\begin{aligned} \log p(x; y) &= \log \int_Z p(x; y; z) dz \\ &= \log \int_Z q(z|x; y) \frac{p(x; y; z)}{q(z|x; y)} dz \end{aligned}$$

For a concave log, Jensen's inequality states that

$$\log E_{q(z|x,y)} f(z) = E_{q(z|x,y)} \log f(z);$$

for any function f . Using this:

$$\log p(x,y) = \int q(z|x,y) \log \frac{p(x,y;z)}{q(z|x,y)} dz$$

The definition of conditional probability gives $p(x,y;z) = p(x|y,z) p(y,z)$. Assuming that z (which movement is performed) and y (how the movement is performed) are independent random variables, we get $p(y,z) = p(y) p(z)$ and thus $p(x|y,z) = p(x|z) p(y)$.

$$\begin{aligned} \log p(x,y) &= \int q(z|x,y) \log \frac{p(x|z) p(y)}{q(z|x,y)} dz \\ &= \int q(z|x,y) [\log p(x|z) + \log p(y)] dz + \int q(z|x,y) \log \frac{p(z)}{q(z|x,y)} dz \\ \text{By definition of the KL-divergence } KL(q||p) &= \int q \log \frac{q}{p} \quad [37]: \\ &= \int q(z|x,y) [\log p(x|z) + \log p(y)] dz - KL(q(z|x,y)||p(z)) \\ &= E_{q(z|x,y)} [\log p(x|z) + \log p(y)] - KL(q(z|x,y)||p(z)) \\ &= L(x,y) \end{aligned}$$

What does the first term represent? Invoking the decoder generative model for some input x_i , we get:

$$\begin{aligned} x_i &= \text{Dec}(z_i; y_i) + \epsilon_i, \text{ for some } \epsilon_i \sim N(0, \sigma^2) \\ \text{Therefore:} \\ \log p(x|y,z) &= \log N(x_i | \text{Dec}(z_i; y_i), \sigma^2) \\ &= \log \frac{1}{\sigma \sqrt{2\pi}} \exp \left\{ -\frac{(x_i - \text{Dec}(z_i; y_i))^2}{2\sigma^2} \right\} \\ &= \log \frac{1}{\sigma \sqrt{2\pi}} - \frac{(x_i - \hat{x}_i)^2}{2\sigma^2} \end{aligned}$$

Dropping the constants (shifting and scaling), we see that this is just the plain absolute difference squared between the input and the VAE's reconstructed output. Maximizing the ELBO $L(x,y)$ means maximizing, in part, $\log p(x|y,z)$ and thus minimizing the reconstruction loss $\|x_i - \hat{x}_i\|^2$.

The second and third terms represent a regularization with respect to this reconstruction loss.

A.4 Derivation of Lower Bound in the Unlabeled Case

In this section we walk through the derivation for obtaining the evidence lower bound (ELBO) used for training on unlabeled data. In this case, we seek to find an expression smaller or equal to the log-likelihood $\log p(x)$ over unlabeled inputs. We will make use of a uniform prior over all categorical labels, as well as a normalized probability distribution over the continuous latent variable z .

Applying the law of total probability to the discrete probability distribution for categorical labels and then to the continuous probability distribution over latent variables we get:

$$\begin{aligned}
\log p(x) &= \log \int_X p(x; y) \\
&= \log \int_X \int_Z p(x; y; z) dz \\
&= \log \int_X \int_Z q(z; y | x) \frac{p(x; y; z)}{q(z; y | x)} dz:
\end{aligned}$$

By the definition of conditional probability $q(z; y | x) = q(y | x) q(z | x; y)$:

$$\log p(x) = \log \int_Y q(y | x) \int_Z q(z | x; y) \frac{p(x; y; z)}{q(z; y | x)} dz$$

Invoking Jensen's inequality stating that $E_{q(y|x)} f(y) > E_{q(y|x)} \log f(y)$ for any function f :

$$\log p(x) > \int_Y q(y | x) \log \int_Z q(z | x; y) \frac{p(x; y; z)}{q(z; y | x)} dz:$$

Invoking Jensen's inequality again, $E_{q(z|x;y)} h(z) > E_{q(z|x;y)} \log h(z)$ gives:

$$\log p(x) > \int_Y q(y | x) \int_Z q(z | x; y) \log \frac{p(x; y; z)}{q(z; y | x)} dz:$$

Using the definition of conditional probability $q(z; y | x) = q(z | x; y) q(y | x)$, we get:

$$\log p(x) > \int_Y q(y | x) \int_Z q(z | x; y) \log \frac{p(x; y; z)}{(q(z | x; y) q(y | x))} dz \quad (1)$$

$$= \int_Y q(y | x) \int_Z q(z | x; y) \log \frac{p(x; y; z)}{q(z | x; y)} dz - \log q(y | x) \quad (2)$$

$$= \int_Y q(y | x) (L(x; y)_{\text{unlabeled}} - \log q(y | x)) \quad (3)$$

$$= E_{q(y|x)} L(x; y)_{\text{unlabeled}} + H(q(y | x)); \quad (4)$$

where the last line uses the definition of the entropy $H(q) = - \int_Y q(y) \log q(y)$, and $L(x; y)_{\text{unlabeled}}$ is the ELBO for the log-likelihood of the labeled case computed previously.

Our lower bound in this case includes the same regularized reconstruction loss $L(x; y)_{\text{unlabeled}}$ as the labeled case, except now it is weighted by the encoding probability for each possible label. This term also features a new term which is defined as the entropy of the classification.

A.5 Web-app for annotations

We propose a tool for easy manual labeling of an input dance dataset. This locally hosted [Dash](#) app can be used by a choreographer wishing to classify their own movement. The graphical-user-interface, displayed in Fig. 3 is easy to navigate and allows multi-session labeling with a choice of starting pose index. The open source project is built such that plotting functions are easy to switch out and customize for any given keypoint dataset.

Figure 3: Screen capture of web labeling app. User enters amount of poses per sequence in (1), and the index of the starting pose of the first sequence in (2). Upon clicking "Get Dance," an animation of the fully connected skeleton appears in (3). The user can zoom and rotate the animation directly, and click into specific frames with (4). User enters Laban Effort (or any chosen label) in (5) and clicks "Save label" to record the inputs to a CSV. A spot is reserved in (6) for an infographic with instructions to encourage consistent labeling.