Semi-Supervised Gait Generation With Two Microfluidic Soft Sensors

Dooyoung Kim, Min Kim, Junghan Kwon, Yong-Lae Park, and Sungho Jo

***Abstract*—Nowadays, the use of deep learning for the calibration of soft wearable sensors has addressed the typical drawbacks of the microfluidic soft sensors, such as hysteresis and nonlinearity. However, previous studies have not yet resolved some of the design constraints such as the sensors are needed to be attached to the joints and many sensors are needed to track the human motion. Moreover, the previous methods also demand an excessive amount of data for sensor calibration which make the system impractical. In this letter, we present a gait motion generating method using only two microfluidic sensors. We select appropriate sensor positions with consideration of the deformation patterns of the lower-limb skins and mutual interference with soft actuators. Moreover, a semi-supervised deep learning model is proposed to reduce the size of calibration data. We evaluated the performance of the proposed model with various walking speeds. From the experiment, the proposed method showed a higher performance with smaller calibration dataset comparing to the other methods that are based on the supervised deep learning.**

***Index Terms*—Wearable robots, soft robot applications, deep learning in robotics and automation.**

I. Introduction

RECENTLY, soft sensors have been widely used in wearable robotics research due to their easy wearability and high stretchability [1], [2]. For example, soft wearable sensors and their calibration methods have been developed for hands [3]–[5], lower-limbs [6], shoulders [7], and a full-body [8].

Although previous research has presented the feasibility of motion sensing using soft sensors, there are still several limitations to be addressed for practical. First, most of the studies have attached the soft sensors to the major joints, such as a hip, knees, and ankles [6], [7], [9]. This approach maximizes the strain sensitivity of the attached sensor [2] and thus provides accuracy in estimation of joint. However, this approach requires many sensors to track the complex human motions, such as gaits, which make it hard to protect the sensors from physical damages. The excessive amount of sensors also causes mutual interference between the sensors and the actuators when we develop a soft robot that consists of soft sensors and actuators [10]. There has been an effort to address this issue by attaching multiple sensors to the thigh for gate analysis [10]. However, this method has a limitation in mapping the sensor signals and the thigh motions.

Another limitation comes with the inherent characteristics of soft microfluidic sensors, such as nonlinearity and hysteresis [1], [2], which make it difficult to calibrate the soft sensors [11]. To address this issue, a supervised neural network model has been proposed for the calibration of the wearable sensors [8]. The study used long short-term memory networks [12] to calibrate the full-body motions with a soft sensing suit that consists of 20 soft strain sensors.

Despite their achievements in overcoming the limitations of the soft sensor calibration, the supervised learning methods require an extensive calibration dataset to train the model and thus it is difficult to be used in practical applications of wearable robots [13]. Furthermore, the users have to collect a new calibration everytime they wear the sensing suit [8]. Therefore, reducing both the size and the number of the calibration datasets is one of the critical problems to be solved for soft sensors to be used in wearable robots and devices with practicality.

In this letter, we address the challenges of the soft wearable sensors by taking the advantage of semi-supervised learning [14], and decided to reconstruct the gait motion of the wearer as a target application (Fig. 1), which has recently been spotlighted in soft robotics area [15], [16]. We first determined the sensor positions based on easy wearability and accurate recognition of the gait patterns, and then generated a natural gait motion in three dimensional space using only two soft strain sensors.

We used a deep auto encoder (AE) [17] to embed the gait motion to a latent motion manifold, and analyzed its representation qualities. With this manifold, we generated a natural human gait motion using the sensor outputs. The performance evaluation showed that the proposed method was able to reduce the need for the calibration dataset and also improves the calibration performances compared with the previous methods.

The rest of this letter is organized as follows. Section II presents the design and the fabrication process of the gait

Manuscript received October 16, 2018; accepted March 10, 2019. Date of publication April 1, 2019; date of current version April 8, 2019. This letter was recommended for publication by Associate Editor C.-H. Yeow and Editor K.-J. Cho upon evaluation of the reviewers' comments. This work was supported by the National Research Foundation funded by the Korean Government (MSIT) under Grant NRF-2016R1A5A1938472. *(Corresponding author: Sungho Jo.)*

D. Kim, M. Kim, and S. Jo are with the School of Computing, Korea Advanced Institute of Science and Technology, Daejeon 305-701, South Korea (e-mail: dykim07@kaist.ac.kr; m-kim@kaist.ac.kr; shjo@kaist.ac.kr).

J. Kwon and Y.-L. Park are with the Department of Mechanical Engineering and Aerospace Engineering and Institue of Advanced Machines and Design, Seoul National University, Seoul 08826, South Korea (e-mail: jhkwon@snu.ac.kr; ylpark@snu.ac.kr).

This letter has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors. This video demonstrates the performance of the proposed semi-supervised deep learning model for generating human gait motions using only two micro soft sensors. It includes both data acquisition setup and gait generation result. The result shows that the proposed method presents a higher performance with smaller calibration dataset comparing to the other method.

Digital Object Identifier 10.1109/LRA.2019.2907431

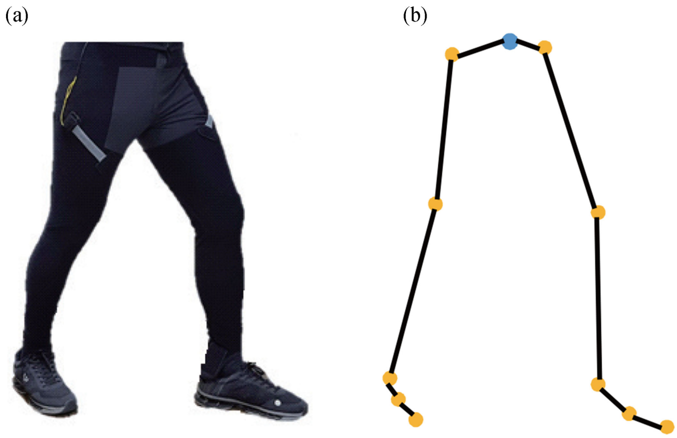

Fig. 1. A wearable gait sensing pants using two microfluidic soft sensors and the modeled 3D skeleton using the proposed motion generating method.

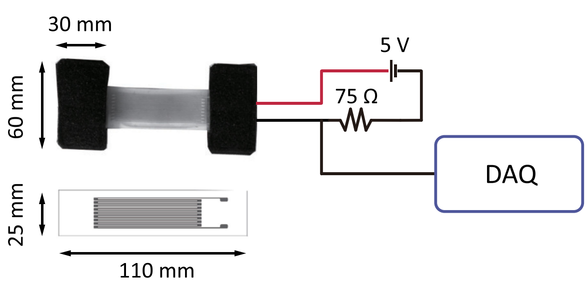

Fig. 2. Microfluidic soft sensor module and its data acquisition circuit.

sensing pants and its signal characteristics. Section III explains the proposed semi-supervised calibration method. Section. IV discusses the performance evaluation of the proposed method. Finally, we conclude our research and present future works in Section V.

II. Materials

A. Sensor Fabrication and Placement

Fig. 2 illustrates the microfluidic soft strain sensor and its data acquisition circuit for human gait motion sensing. The design and the fabrication process of the sensors were based on the previous research [8], [18]. Each sensor was connected with a simple voltage divider circuit to capture the resistance changes, and the changes were transferred to the computer through a data acquisition (DAQ) device.

The position and the stretch direction of the soft strain sensors were selected considering the following constraints. First, the major joints in the lower-limb (hip, knee, ankle) were excluded from the candidate positions to prevent the mutual interference with soft actuators. Second, we wanted to use as small number of sensors as possible to simplify the design process and the calibration effort.

In order to find the locations that meet the constraints, we focused on the relationship between the gait motion and the deformation of human skin.

Since the soft strain sensors measure a motion of the user from the skin deformation at the position where the sensors are

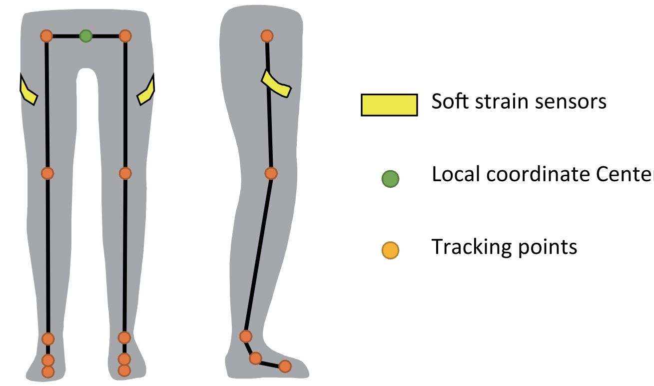

Fig. 3. The position of the soft strain sensors, and the local coordinate center and tracking points of our motion generating method.

attached, the strain sensitivity is maximized when the sensor is attached on the line of skin that has maximum deformation [6], [18]. We therefore selected the positions of the soft strain sensors based on the relationship between skin deformation and dynamic lower-limb postures [19], [20].

Fig. 3 shows the location and the direction of the soft strain sensors. We only used two soft sensors to detect and reconstruct human gait motions. The ends of each sensor were attached to the upper-side and the middle-front of the thigh, respectively. Those positions are where the deformation of the skin is maximized during walking. By attaching the sensors to those positions, we were able to obtain sufficient resistance changes of the sensor for gait pattern recognition.

B. Experimental Setup and Data Acquisition

The experiment environment for data acquisition is shown in Fig. 4. A healthy male (height: 179 cm, leg length: 92 cm) was recruited as a subject in this study. We set up a treadmill (width: 37 cm , length: 90 cm) to collect a dataset with various walking speeds: from 2 km/h to 6 km/h with an increment of 1 km/h.

In order to capture the subject's data, we selected 10 tracking points in the lower-limb: both hips, knees, ankles, feet, and toes (Fig. 3). The signal outputs of the two soft strain sensors were measured by the voltage divider circuit and the DAQ (NI USB-6353, National Instrument). At the same time, the subject's motion was collected by an optical motion capture system (Prim13, Opti-Track). The sampling rate of both the DAQ and the motion capture device was 120 Hz, and the resolution of the DAQ was 16 bit.

The notions of the dataset are as follows. A signal vector $x^{(t)} \in \mathbb{R}^S$ and a position vector $y^{(t)} \in \mathbb{R}^{3M}$ are denoted by:

$$x^{(t)} = \left\{ x_1^{(t)}, x_2^{(t)}, \ldots x_S^{(t)} \right\} \tag{1}$$

$$y^{(t)} = \left\{ y_1^{(t)}, y_2^{(t)}, \ldots, y_M^{(t)} \right\} \tag{2}$$

where S is the number of sensor, and M is the number of tracking point. In order to include the sequential phenomenon of the human motion, the sensor output x is converted to the sequence

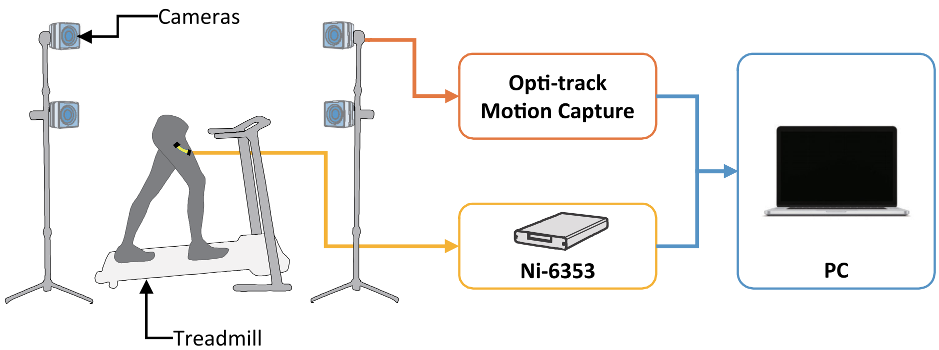

Fig. 4. Schematic representation of experiment setup.

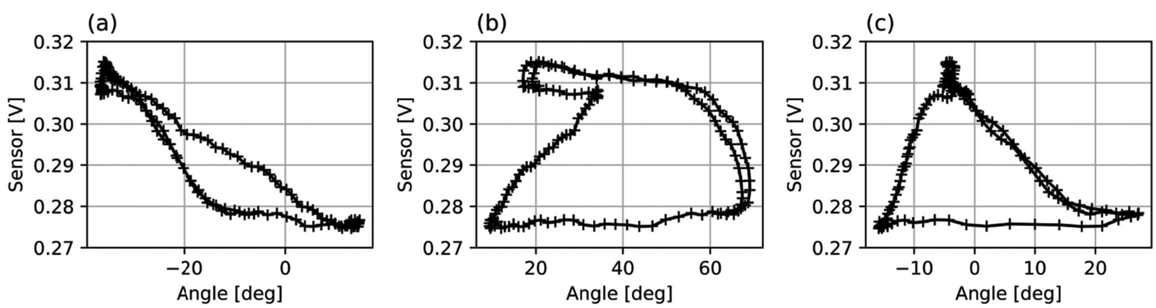

Fig. 5. Relationship between the captured joint angle on the sagittal plain and sensor outputs: (a) knee, (b) ankle, (c) toe.

of sensor outputs with time window t:

$$x^{(t-n:t)} = \left\{x^{(t-n)}, x^{(t-n+1)}, \ldots, x^{(t)}\right\} \tag{3}$$

Fig. 5 illustrates the relationship between the captured sensor signals and the joint angles at 5 km/h of walking speed. In this figure, we can observe a unique pattern in each sensor signals. However, those patterns are too complex to be fitted to the linear models that are widely used in the soft sensor calibration [1], [2]. These complex patterns are caused by the common limitations of microfludic soft sensors, such as nonlinearity and hysteresis [11] and complex deformations of the human skin [10].

III. Methods

The goal of the sensor calibration is to find a calibration model F in

$$\hat{y} = F(x|\theta) \tag{4}$$

where θ is the calibration parameters and $\hat{y}$ represents predicted target vector. We learn θ through the semi-supervised deep learning.

Two types of datasets are used for our model: a calibration dataset (labeled data) and a gait motion dataset (unlabeled data). The calibration dataset consists of the sensor output and its corresponding pose. The relationship between these two features depends on the relative position between the sensor and the target joint [7], [8], but it changes whenever the wears and takes off the sensor, since the positions of the sensors are subject to changes. This problem causes the system to recollect the calibration dataset every time making the soft wearable sensors impractical for everyday use.

On the other hand, the gait motion dataset only contains the gait pattern of the user. Thus, we do not have to collect this motion dataset every time because the motion is independent of the sensor placements. Therefore, collecting the motion dataset is easier than the calibration dataset. In this section, we present how to use this unlabeled data to simplify the calibration process of the soft wearable sensors.

The dimension of the dataset is an important factor in machine learning because the prediction quality of the model decreases as the dimension of the dataset increases when the number of the training dataset is fixed [21]. Thus, we used semi-supervised leanings that define a low-dimensional manifold that represents the hidden characteristics of the original dataset using the unlabeled data to improve the prediction accuracy.

In our case, a human gait pose is represented as high-dimensional feature vector $y \in \mathbb{R}^{30}$. However, the probability density of y in this high-dimensional space is relatively low, since the position of the joints during a gait motion is highly correlated to each other. Therefore, we assume that a low-dimensional latent motion manifold representing that the high-dimensional motion feature vector y exists, and probability density of y decreases rapidly when it moves away from the latent motion manifold.

The overall architecture of the proposed calibration model is shown in Fig. 6. The model is composed of three components: a sequential encoder network (SEN), an alignment network (AliN), and a motion representation network (MRN). The MRN defines a latent motion manifold $\mathcal{Z}$ that represents the

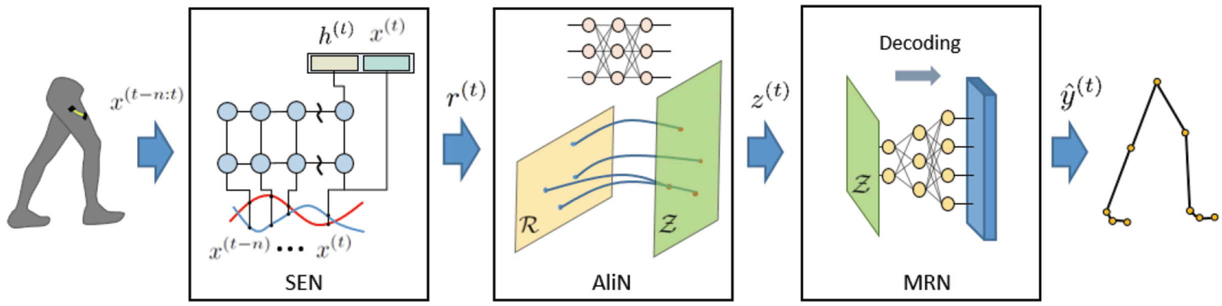

Fig. 6. Overall architecture of the proposed semi-supervised gait motion generation model.

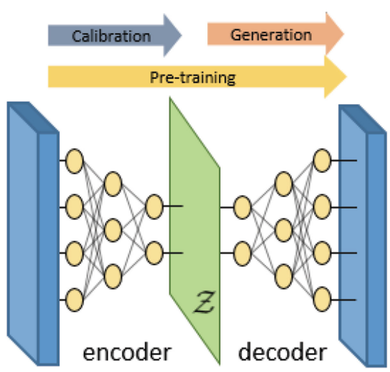

Fig. 7. Motion Representation Network: The model consists of encoder network and decoder network.

high dimensional human gait motion y in the low dimensional latent motion manifold z. We built the SEN based on the recurrent neural networks (RNN) to extract the sequential feature vector r in the sensor output. The AliN defines a mapping from the sequential feature vector r to the latent motion vector z in $\mathcal{Z}$. Details of each component are explained as follows.

A. MRN: Motion Representation Network

We build the MRN using a deep AE, a type of unsupervised machine learning methods (Fig. 7). The network consists of an encoder network g and a decoder network f as follows:

$$z = g(y|\theta_g) \tag{5}$$

$$\hat{y} = f(z|\theta_f) \tag{6}$$

where θ, is the model parameter. We define both encoder and decoder networks using three fully-connected neural network (FCNN) layers:

$$f_{(i)}(z_i; w_i, b_i) = z_i^T w_i + b_i \tag{7}$$

where z_i is an input vector of the i-th layer, w_i, b_i are weights and biases. A rectified linear unit (ReLU) is used at each layer except the last one as an activation function layer [22].

$$ReLU(z) = \max\{0, z\} \tag{8}$$

In the training step, the original motion feature y in the high-dimensional space $\mathcal{Y}$ is converted to the hidden motion feature vector $z \in \mathcal{Z}$ through the encoder network g. After that, the feature vector z is converted to the reconstructed motion feature $\hat{y}$ through the decoder network f. In order to optimize the parameters in f and g, we used a mean square error (MSE) as the cost function:

$$c_p(\hat{y}, y) = \frac{1}{3NM} \sum_i^N \sum_m^M \sum_d^3 (\hat{y}_{i,m,d} - y_{i,m,d})^2$$

where N is the number of features, and d is the dimension index of the tracking points.

B. SEN: Sequential Encoder Network

We defined the SEN based on the DFM-Net [8] to extract the sequential phenomenon r in the sensor outputs x.

$$r^{(t)} = SEN(x^{(t-n:t)}) \tag{9}$$

The difference between the original SEN and our implementation is that we used a gated recurrent unit (GRU) [23] instead of a long-short term memory (LSTM) [12]. A GRU is a type of an RNN model that reduces the computational overhead while maintaining the performance of LSTMs. GRUs merge the forget gate and input gate into a single update gate to simplify the model. The update gate a_t and reset gate e_t at time t are defined as follows:

$$a^{(t)} = \sigma(W_a x^{(t)} + U_a h^{(t-1)}) \tag{10}$$

$$e^{(t)} = \sigma(W_e x^{(t)} + U_e h^{(t-1)}) \tag{11}$$

The first step in GRUs is determining what information is useful from the previous hidden state h_t by following equation:

$$\tilde{h}^{(t)} = \tanh(W_h x^{(t)} + e^{(t)} \odot U_h h^{(t-1)}) \tag{12}$$

where $\tanh(\cdot)$ is the hyperbolic tangent function, $\odot$ is the element-wise multiplication operation, W^h, U^h is input weight and recurrent weight, respectively. Then, the new hidden state h_t is obtained from the $\tilde{h}_t$ and h_t as follows:

$$h^{(t)} = a^{(t)} \odot h^{(t-1)} + (1 - a^{(t)}) \odot \tilde{h}^{(t)} \tag{13}$$

In order to prevent the overfitting, we used dropout [24] in the first layer of the GRU with the dropout rate of 0.5. We used the sequential output of the soft sensors $x^{(t-n:t)}$ as the input of the GRU, and then extracted the temporal feature vector $h^{(t)}$. In order to represent both current and temporal features at once, we concatenated the current sensor output $x^{(t)}$ and temporal feature $h^{(t)}$ into the sequential feature vector $r^{(t)}$.

TABLE I
NETWORK MODELS AND PARAMETERS OF THE PROPOSED MODEL

Component	Layer index	Model	Parameters
SEN	1	GRU	Hidden: 256, Dropout: 0.5
	2	GRU	Hidden: 256
AliN	1	FCNN	In: 258, Out: 256, ReLU
	2	FCNN	In: 256, Out: 256, ReLU
	3	FCNN	In: 256, Out: 128
MRN	encoder 1	FCNN	In: 30, Out: 128, ReLU
	encoder 2	FCNN	In: 128, Out: 128, ReLU
	encoder 3	FCNN	In: 128, Out: 3
	decoder 1	FCNN	In: 3, Out: 128
	decoder 2	FCNN	In: 128, Out: 128, ReLU
	decoder 3	FCNN	In: 128, Out: 30

C. AliN: Alignment Network

The feature vectors r and z represent the sequential pattern of the soft sensors and the hidden characteristic of the human gait motion respectively. These two feature vectors are in different feature spaces; thus, we need a method to transform from the r to z. The AliN aims to represent a mapping from the r to the z:

$$\hat{z}^{(t)} = AliN(r^{(t)}) \tag{14}$$

The AliN consists of three FCNN layers, and ReLU is used as an activation function in every layer except the last layer.

D. Implementation Detail

The training procedure of the proposed model has two steps: pre-training step and calibration step. In the pre-training step, we trained the MRN to define the latent motion manifold with the gait motion dataset. From the MRN, we can obtain the encoder network g and decoder network f.

After that, we project the motion feature y in the calibration dataset into the latent motion manifold $\mathcal{Z}$, and get the hidden representation z of y using the encoder network g.

In the calibration step, we trained the SEN and AliN with the sensor input x and the hidden representation of the motion z at once. The cost function for the calibration step is the MSE loss function defined as follow:

$$c_c(\hat{z}, z) = \frac{1}{N}\sum_{i}^{N}(\hat{z}_i - z_i)^2$$

After training, we can directly generate a human gait motion $\hat{y}$ from the soft sensor input x using the following procedure:

$$\hat{y}^{(t)} = f\big(AliN\big(SEN\big(x^{(t-n:t)}\big)\big)\big) \tag{15}$$

First, the SEN captures the sequential characteristic r of the soft sensors. After that, the AliN projects the r into the manifold space $\mathcal{Z}$. Finally, the decoder network in the MRE generates the most feasible gait motion based on the hidden information in the latent motion feature z.

We implemented our model using PyTorch deep learning framework [25]. Detailed model structure and hyper-parameters are described in Table I. The sampling window size of the soft sensor was 120 frames. Thus, we observed previous one second. All parameters in the proposed model were initialized by Gloraot initialization algorithm [26]. The numbers of epoch in pre-training and calibration steps were 60 and 30 respectively. Both pre-training and calibration steps used Adam [27] as our optimization algorithm (learning rate: 0.001), and the mini-batch size was 512.

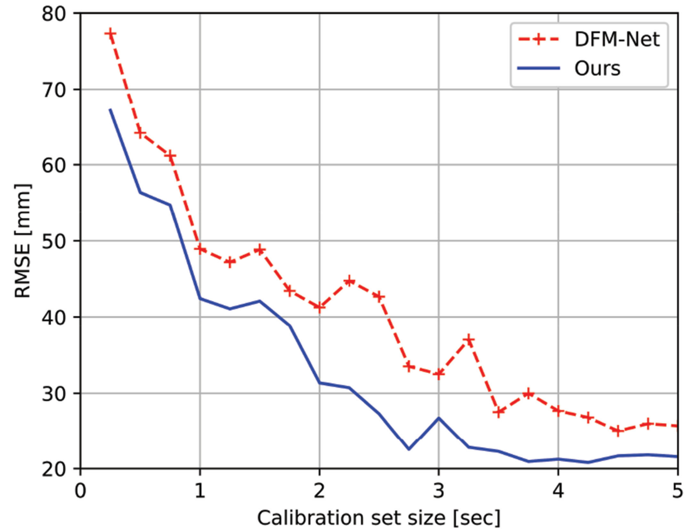

Fig. 8. Comparison of the generating accuracy with varied length of calibration datasets between the DFM-Net and the proposed method.

IV. EXPERIMENTAL RESULTS

We compared the proposed method with the DFM-Net [8], which is the state-of-the-art method for soft sensor calibration. In this experiment, we modified the structure of the DFM-Net as follow: two GRU layers and three FCNN layers with the same hyper-parameters of the proposed method, to show the best performance in our test case. In order to evaluate the proposed method, we used 1, 200 frames (10 seconds) in each speed levels as our test dataset in all of the experiments. The total number of test frame was 60,000 ($1,200 \times 5$ speed levels). As a performance evaluation metric, we used a root mean square error (RMSE) defined as follow:

$$RMSE(\hat{y}, y) = \sqrt{\frac{1}{3NM}\sum\nolimits_{i}^{N}\sum\nolimits_{m}^{M}\sum\nolimits_{d}^{3}(\hat{y}_{i,m,d} - y_{i,m,d})^2}$$

A. Demands on Calibration Dataset

In order to examine how many calibration datasets are required to train the proposed method, we sampled the calibration dataset from each walking speed levels with a specific time size. Fig. 8 shows the RMSE of the calibration results with various sizes of the calibration dataset of each speed levels. The result shows that our method required fewer amount of the calibration dataset than the DFM-Net to achieve the same quality.

The main difference between our method and the DFM-Net is the MRN. The DFM-Net trains both SEN and the kinematic decoder network at once from the calibration dataset. In contrast, the proposed method concentrates more on finding the relationship between the sensor output and the gait motion. Therefore, the proposed method requires less calibration dataset than the DFM-Net.

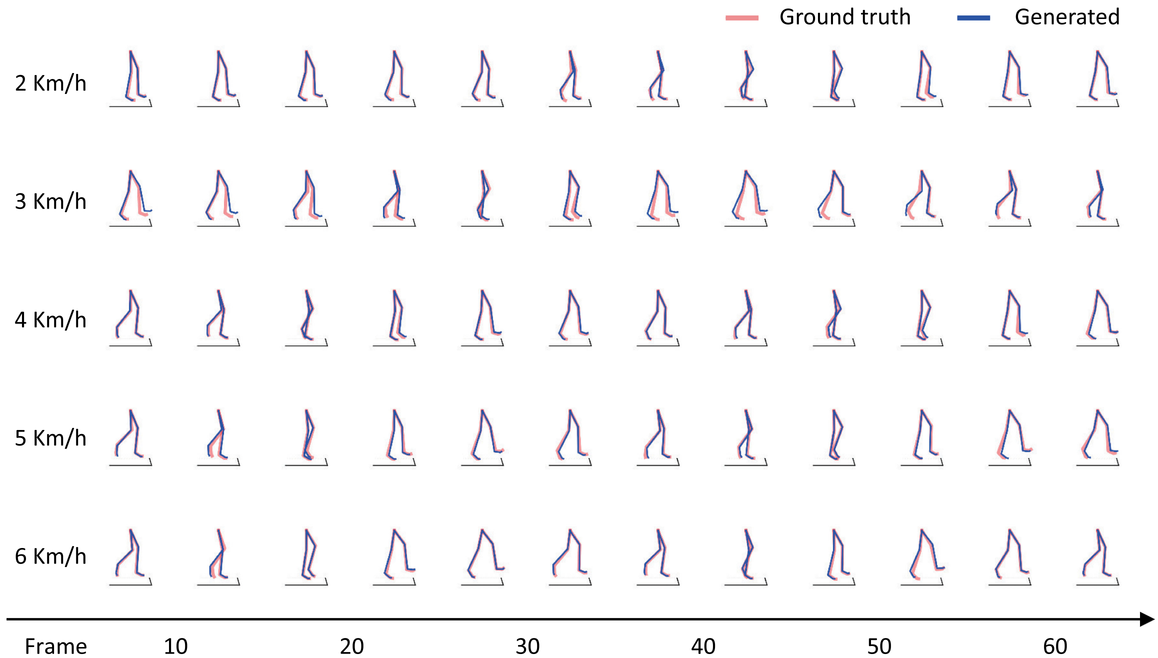

Fig. 9. Generated gait motion flow using the proposed method. The motions were picked up every fifth frame (red: ground truth motion, blue: generated motion).

TABLE II
EXPERIMENTAL RESULTS OF GAIT MOTION GENERATING AND COMPARISON OF THE OTHER SOFT SENSOR CALIBRATION METHODS

RMSE (mm)	Ours	DFM-Net	LR-4
Overall	21.22	29.00	53.50
2 km/h	19.85	37.81	53.90
3 km/h	33.30	36.85	50.07
4 km/h	17.76	16.79	44.21
5 km/h	18.01	30.14	49.80
6 km/h	23.36	35.90	66.83

TABLE III
EXPERIMENTAL RESULT OF THE JOINT'S POSITIONS

RMSE (mm)	Hip	Knee	Ankle	Foot	Toe
Overall	2.06	12.39	26.25	30.16	30.52
2 km/h	1.72	12.31	23.63	24.92	25.24
3 km/h	1.74	13.80	37.42	44.23	44.64
4 km/h	1.88	8.41	19.73	23.10	24.08
5 km/h	1.92	9.88	19.72	23.59	23.97
6 km/h	2.84	16.04	26.63	29.66	29.54

B. Quality of Generated Gait Motion

Table II presents the performance of the proposed method in comparison with the other methods. We sampled four seconds from each walking speed levels (2, 400 frames) as our calibration dataset. We then compared our method with the modified DFM-Net and a linear regression model [28], which have been used in multiple soft sensor calibration methods [29]. In order to select the polynomial order in the regression model, we changed the polynomial order from first to sixth and tested its prediction performance. According to the test, the fourth polynomial order (LR-4) showed the best performance with the given dataset and, therefore, was used to compare with the proposed method and DFM-Net.

From the result, the proposed method was able to generate the gait motion with highest accuracy (21.22 mm in overall), while the RMSE of the worst-case was only 33.30 mm. Table III shows the calibration quality of the lower-limb joint's position. As can be seen in the table, the RMSE of the terminal joints (toes) is only 30.53 mm.

We captured the motions from the generated results every fifth frame to evaluate the naturalness of the generated gait motion using the proposed method. As we can see in the Fig. 9, the generated motions almost exactly match the ground truth. This result supports that the proposed design of the motion-sensing pants and the calibration method are accurate enough to generate natural gait motions.

C. Analysis of Latent Motion Manifold

Our model acquires the high-level hidden features of gait motion, such as phase and speed, from the data from two microfluidic sensors and generates a motion based on these features. Therefore, the representation ability is an important factor of the performance.

In order to estimate the performance of the proposed MRN, we illustrated the space $\mathcal{Z}$ into the 3-D Euclidean space (Fig. 10). We projected the continuous gait motion features y into the latent space $\mathcal{Z}$ using the encoder network g, and drew them in the Euclidean space. In addition, we annotated some key phases of the gait cycle (heel strikes and mid-stance) and their corresponding motions in the figure. In this figure, we can see that the flows of the gait motions are in a closed loop in $\mathcal{Z}$, and the flow of each speed level is also obviously distinctive. This result validates our assumption of the existence of a low-dimensional latent motion manifold that represent the high-dimensional motion feature.

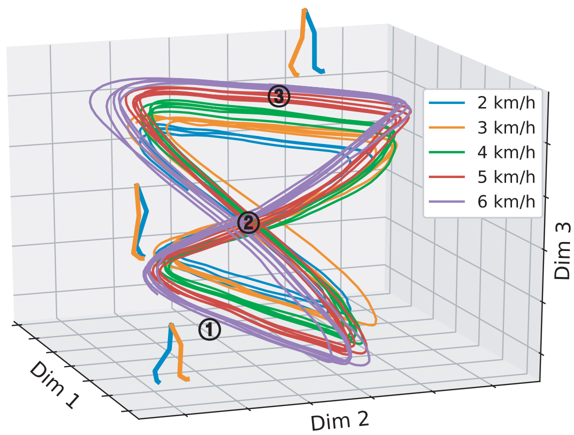

Fig. 10. Flows of latent feature vectors z in the latent motion manifold $\mathcal{Z} \in R^3$ and their gait motions: 1 right heel strike, 2 mid-stance, and 3 left heel strike.

V. Conclusion

We proposed the semi-supervised soft sensor calibration method that generates the natural human gait motions using only two soft microfluidic sensors.

Unlike previous research that required many soft sensors attached to multiple joints to measure the gait motions, we were able to reduce the number of soft sensor to two so that eliminating the interferences between the soft sensors and actuators.

Moreover, we developed a semi-supervised deep learning method to reduce the number of the calibration datasets. We defined the latent motion manifold using the deep auto encoder and the unlabeled motion dataset. Our method was able to generate more accurate gait motion even with the small size of the calibration dataset compared with the other soft sensor calibration methods that were based on supervised deep learning. Thus, the proposed method can perform the calibration more quickly when the user wears it.

To the best of our knowledge, this is the first study of generating a gait motion using only two soft microfluidic sensors. As a future work, we will improve our model to recognize the phase of the gait cycle in real-time. Furthermore, We will integrate the proposed method with soft actuators for assisting human walking. We believe that our proposed method will significantly improve the soft wearable robots especially by making the technology more functional and practical.

References

[1] Y.-L. Park, D. Tepayotl-Ramirez, R. J. Wood, and C. Majidi, "Influence of cross-sectional geometry on the sensitivity and hysteresis of liquid-phase electronic pressure sensors," *Appl. Phys. Lett.*, vol. 101, no. 19, 2012, Art. no. 191904.

[2] Y.-L. Park, C. Majidi, R. Kramer, P. Bérard, and R. J. Wood, "Hyperelastic pressure sensing with a liquid-embedded elastomer," *J. Micromech. Microeng.*, vol. 20, no. 12, 2010, Art. no. 125029.

[3] J. B. Chossat, Y. Tao, V. Duchaine, and Y.-L. Park, "Wearable soft artificial skin for hand motion detection with embedded microfluidic strain sensing," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2015, pp. 2568–2573.

[4] W. Park, K. Ro, S. Kim, and J. Bae, "A soft sensor-based three-dimensional (3-D) finger motion measurement system," *Sensors*, vol. 17, no. 2, p. 420, 2017.

[5] Y. Lee, M. Kim, Y. Lee, J. Kwon, Y. Park, and D. Lee, "Wearable finger tracking and cutaneous haptic interface with soft sensors for multi-fingered virtual manipulation," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 1, pp. 67–77, Feb. 2019.

[6] Y. Mengüç *et al.*, "Wearable soft sensing suit for human gait measurement," *Int. J. Robot. Res.*, vol. 33, no. 14, pp. 1748–1764, 2014.

[7] H. Lee, J. Cho, and J. Kim, "Printable skin adhesive stretch sensor for measuring multi-axis human joint angles," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2016, pp. 4975–4980.

[8] D. Kim, J. Kwon, S. Han, Y.-L. Park, and S. Jo, "Deep full-body motion network (DFM-net) for a soft wearable motion sensing suit," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 1, pp. 56–66, Feb. 2019.

[9] C. R. Walker and I. A. Anderson, "Monitoring diver kinematics with dielectric elastomer sensors," in *Proc. Electroactive Polymer Actuators Devices*, vol. 10163, 2017, Paper 1016307.

[10] M. Borghetti, M. Serpelloni, E. Sardini, and O. Casas, "Multisensor system for analyzing the thigh movement during walking," *IEEE Sensors J.*, vol. 17, no. 15, pp. 4953–4961, Aug. 2017.

[11] S. Han, T. Kim, D. Kim, Y.-L. Park, and S. Jo, "Use of deep learning for characterization of microfluidic soft sensors," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, pp. 873–880, Apr. 2018.

[12] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[13] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[14] X. Zhu, "Semi-supervised learning literature survey," *Comput. Sci., Univ. Wisconsin–Madison*, vol. 2, no. 3, p. 4, 2006.

[15] J. Kim *et al.*, "Autonomous and portable soft exosuit for hip extension assistance with online walking and running detection algorithm," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 1–8.

[16] Y.-L. Park *et al.*, "Design and control of a bio-inspired soft wearable robotic device for ankle–foot rehabilitation," *Bioinspiration Biomimetics*, vol. 9, no. 1, 2014, Art. no. 016007.

[17] P. Baldi, "Autoencoders, unsupervised learning and deep architectures," in *Proc. Int. Conf. Unsupervised Transfer Learn. Workshop*, 2011, vol. 27, pp. 37–50. [Online]. Available: http://dl.acm.org/citation.cfm?id=3045796.3045801

[18] Y.-L. Park, B. R. Chen, and R. J. Wood, "Design and fabrication of soft artificial skin using embedded microchannels and liquid conductors," *IEEE Sensors J.*, vol. 12, no. 8, pp. 2711–2718, Aug. 2012.

[19] H. Lee, K. Hong, and Y. Lee, "Ergonomic mapping of skin deformation in dynamic postures to provide fundamental data for functional design lines of outdoor pants," *Fibers Polymers*, vol. 14, no. 12, pp. 2197–2201, Dec. 2013.

[20] J. Choi and K. Hong, "3D skin length deformation of lower body during knee joint flexion for the practical application of functional sportswear," *Appl. Ergonom.*, vol. 48, pp. 186–201, 2015.

[21] G. V. Trunk, "A problem of dimensionality: A simple example," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-1, no. 3, pp. 306–307, July 1979.

[22] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 807–814.

[23] K. Cho *et al.*, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," in *Proc. Confe. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1724–1734.

[24] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[25] A. Paszke, S. Gross, S. Chintala, and G. Chanan, "Pytorch," 2017. [Online]. Available: https://github.com/pytorch/pytorch

[26] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256.

[27] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Int. Conf. Learn. Represent. (ICLR)*, 2015.

[28] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to Linear Regression Analysis*, vol. 821. Hoboken, NJ, USA: Wiley, 2012.

[29] M. Totaro *et al.*, "Soft smart garments for lower limb joint position analysis," *Sensors*, vol. 17, no. 10:2314, 2017.