











# Self-SLAM: A Self-supervised Learning Based Annotation Method to Reduce Labeling Overhead

Alfiya M. Shaikh<sup>1</sup>(✉) , Hrithik Nambiar<sup>1</sup> , Kshitish Ghate<sup>2</sup>,  
Swarnali Banik<sup>1</sup> , Sougata Sen<sup>1</sup> , Surjya Ghosh<sup>1</sup> ,  
Vaskar Raychoudhury<sup>3</sup> , Niloy Ganguly<sup>4</sup> , and Snehanshu Saha<sup>1</sup> 

<sup>1</sup> Computer Science and Information Systems and APPCAIR, BITS Pilani K K Birla  
Goa Campus, Zuarinagar, India

{2023proj027, f20190100g, p20210016, sougatas, surjyag,  
snehanshus}@goa.bits-pilani.ac.in

<sup>2</sup> Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA  
15213, USA

kghate@andrew.cmu.edu

<sup>3</sup> Computer Science and Software Engineering, Miami University, Oxford, OH 45056,  
USA

raychov@miamioh.edu

<sup>4</sup> Computer Science and Engineering, IIT Kharagpur, Kharagpur, India  
niloy@cse.iitkgp.ac.in

**Abstract.** In recent times, Deep Neural Networks (DNNs) have been effectively used to tackle various tasks such as emotion recognition, activity detection, disease prediction, and surface classification. However, a major challenge in developing models for these tasks requires a large amount of labeled data for accurate predictions. The manual annotation process for a large dataset is expensive, time-consuming, and error-prone. Thus, we present SSLAM (Self-supervised Learning-based Annotation Method) framework to tackle this challenge. SSLAM is a self-supervised deep learning framework designed to generate labels while minimizing the overhead associated with tabular data annotation. SSLAM learns valuable representations from unlabeled data that are applied to the downstream task of label generation by utilizing two pre-text tasks with a novel *log – cosh* loss function. SSLAM outperforms supervised learning and Value Imputation and Mask Estimation (VIME) baselines on two datasets - Continuously Annotated Signals of Emotion (CASE) and wheelchair dataset. The wheelchair dataset is our novel unique surface classification dataset collected using wheelchairs showcasing our framework's effectiveness in real-world scenarios. All these results reinforce that SSLAM significantly reduces the labeling overhead, especially when there is a vast amount of unlabeled data compared to labeled data. The code for this paper can be viewed at the following link: <https://github.com/Alfiya-M-H-Shaikh/SSLAM.git>

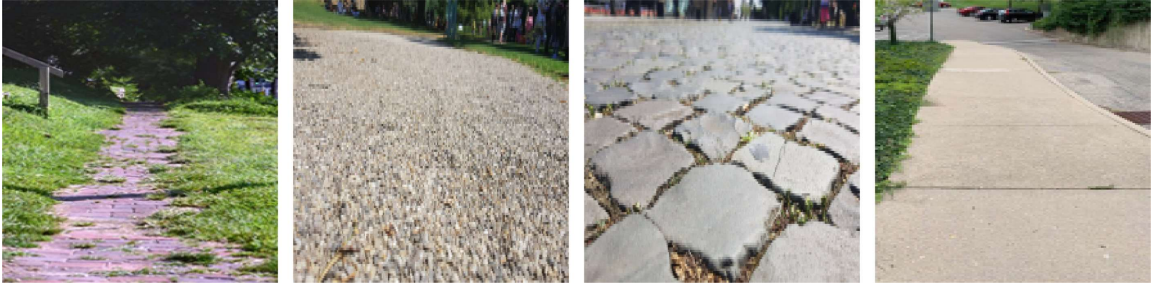
**Supplementary Information** The online version contains supplementary material available at [https://doi.org/10.1007/978-3-031-70378-2\\_8](https://doi.org/10.1007/978-3-031-70378-2_8).

**Keywords:** Self-supervised Learning · Annotation Overhead ·  
*log - cosh* loss function · Limited Labelled Data

## 1 Introduction

Recently, Deep Neural Networks (DNN) have been found effective in different domains including healthcare, activity recognition, surface classification, and human behavior understanding (e.g., emotion recognition, elderly monitoring) [7, 15, 22]. These systems collect data from the physiological signals and IMU (Inertial Measurement Unit) sensor and then employ a DNN model for the intended task. However, a major challenge in achieving optimal performance by utilizing DNN models is the requirement for a substantial volume of labeled data as the manual annotation process is fatigue-inducing, error-prone, and time-consuming [1, 9, 12]. At present time, we are surrounded by a large number of pervasive devices (e.g., smartphones, smartwatches, IoT devices) that generate a lot of data; a majority of which remains unlabeled. Consequently, despite the abundance of data, we are unable to fully leverage the potential of this extensive dataset due to the substantial overhead involved in annotation. Hence, the development of efficient strategies for annotating large volumes of data is essential.

In this paper, we aim to address the problem of automatic annotation of a large volume of continuous sensor data streams for socially relevant problems such as detecting wheelchair-accessible path characteristics from the built environment using smartphone-embedded motion sensors. Wheelchair users while undertaking their daily activities, will move through various built surfaces, such as concrete sidewalks, asphalt, granite tiles, cobblestones, etc. in the outdoor environment and carpet, linoleum, mosaic, etc. in the indoor environment. We captured the vibration generated by different surfaces through the accelerometer and gyroscope sensors in the user's smartphone and then used a specialized AI framework to classify the surfaces based on their characteristic vibration patterns. The data collection process for this unique dataset has been extensively documented in our previous work [25]. Often sidewalks are not accessible by wheelchair users depicted in Fig. 1 due to obstacles such as broken/uneven surfaces, steep slopes, high-pile slippery cobblestones (with deep gaps in between) as well as sidewalks with no access ramps. E.g., cobblestones are recognized as grossly inaccessible while concrete sidewalks are considered accessible. However, this problem is a challenging one given the numerous different types of surfaces available in different countries as well as the different types of wheelchairs used by the people. Several wheelchair-related parameters (such as manual or power, tire material, weight, number of wheels, height from the ground at which the smartphone is attached, etc.) are responsible for producing different vibration data streams for the same surface type. Moreover, the user's body weight, height, and disability type can also impact the nature of vibration. Overall, it is non-trivial to manually annotate the different types of data collected in this project across 6 different countries on 3 different continents from 48 different surfaces using 50 wheelchair users on 6 different manual and power wheelchairs.



**Fig. 1.** Non-accessible sidewalks; surface classification required

Various annotation strategies are proposed in the existing literature. First, self-report or expert-driven techniques are utilized wherein the signal fragments are annotated by an (or a group of) experts [23], and distinct unification approaches (e.g., majority voting [13]) are applied to come up with a single rating (or label). For example, the dataset named CASE (*Continuously Annotated Signals of Emotion*) [19] involved participants who used a joystick to provide continuous annotations of their emotions, specifically *valence* and *arousal*, based on the Circumplex Model of emotion [16]. These approaches demand significant user effort and are not easily adaptable to larger scales. The second approach to annotate the signal used an auxiliary modality from a given modality [3]. In this paper, signals from an IMU sensor are annotated automatically, leveraging the availability of acoustic data. However, the dependency on another modality is the major drawback of these approaches. The third approach uses a human-in-the-loop annotation strategy that includes the concept of Active Learning. For example, in [9, 18], a human annotator is included in the loop who recognizes a group of seed samples (with available annotations) to train a base model, which gives outcome for all the remaining unlabeled instances. Next, the outcome from the model is considered depending on the model’s confidence, or the human expert is conferred for the annotation. The key challenges include seed instance identification, involvement of human experts, and lack of clarity (by the human expert) in understanding the problem encountered by the learner [9, 18].

However, we can design an intelligent annotation approach leveraging the apriori knowledge from the domain experts and the *intrinsic properties* of the dataset clusters to reduce human engagement significantly. Thus, we propose the Self-SLAM (SSLAM) annotation framework to label datasets with minimal expert intervention. The framework constitutes a self-supervised algorithm that employs two pretext tasks developed using a contrastive sampling method [24]. We employ pretext tasks to train the encoder in a self-supervised manner, optimize the resultant representations using a parameterized activation function, and then apply a label-noise resilient *log-cosh* loss function for reconstruction. Though this function is similar in structure to the standard loss functions like Mean Squared Error (MSE) and Mean Absolute Error (MAE), it has a desirable analytical property called Lipschitzness that helps to deal with the label noise. This makes the proposed framework robust, and essential to ensure label quality.

The performance of SSLAM was evaluated for two different use cases: (a) emotion annotation and (b) surface classification. First, we evaluated label generation performance to annotate emotion continually on publicly available continuous emotion CASE [19] dataset. The dataset consists of continuous valence-arousal annotations of emotional and physiological responses measured through multiple sensors. SSLAM provides more accurate valence and arousal predictions than a supervised approach leveraging unlabeled data and minimal labeled samples. It outperforms another self-supervised learning framework (VIME) [24] on the same number of labeled and unlabeled data by 20.8% and 17.7% (for valence and arousal, respectively).

We evaluated SSLAM on a subset of our surface vibration dataset collected from wheelchair users. This dataset includes manual wheelchair-induced vibration data from 47 participants across 15 distinct indoor and outdoor surfaces in the USA and China. In this dataset, SSLAM outperforms a supervised learner and VIME by 4.25% and 7.9%, respectively, with the same amount of labeled and unlabeled data. In summary, our paper demonstrates that SSLAM outperforms classical machine learning algorithms such as Logistic Regression, Multi-layer Perceptron (MLP), and XGBoost, and a self-supervised learning algorithm (VIME). In summary, our paper’s key contributions are:

- We proposed a self-supervised framework SSLAM to significantly reduce annotation overhead and demonstrate improvements over the existing baselines using a parameterized Elliot activation function and a new loss function.
- We collected and shared a novel and unique wheelchair-induced surface vibration dataset that enriches the available resources and facilitates further research.
- We present a new reconstruction loss called  $\log - \cosh$  in the SSLAM encoder setup, provide an explanation of its suitability as a viable alternative to MSE loss, and highlight its implications of being robust to label noise and outliers and its relevance to the SSLAM framework.
- We provide empirical evidence on both wheelchair and publicly available CASE datasets to demonstrate that the proposed method is applicable for different use cases such as surface classification and continuous emotion annotation respectively.

## 2 Dataset Description

### 2.1 Wheelchair

As described in Sect. 1, the wheelchair dataset is a collection of surface-induced vibration data caused by the movement of manual wheelchairs in both built and natural environments. Data is collected using an Android smartphone attached tightly to the handrest of a collapsible manual wheelchair. When participants self-propelled the wheelchair across various surfaces, the accelerometer and gyroscope sensors capture the vibration at a sampling rate of 100 Hz. We collected





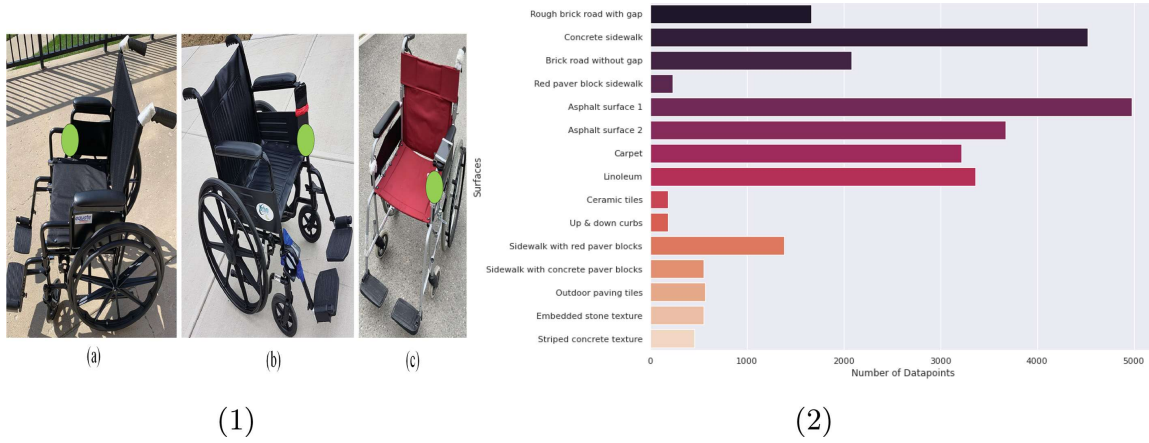
**Fig. 2.** Surfaces used for data collection: **in the USA:** (a) Rough brick road with gap, (b) Concrete sidewalk, (c) Brick road without gap, (d) Red paver block sidewalk, (e) Asphalt surface 1, (f) Asphalt surface 2, (g) Carpet, (h) Linoleum, (i) Ceramic tiles, (j) Up & down curbs **in China:**, (k) Sidewalk with red paver blocks, (l) sidewalk with concrete paver blocks, (m) Outdoor paving tiles, (n) Embedded stone texture, (o) Striped concrete texture (Color figure online)

data from 16 different surfaces in the USA and China as depicted in Fig. 2. Our data collection involved 42 participants and 2 wheelchairs in the USA and 5 participants and 1 wheelchair in China. The manual wheelchairs used in the USA and China for data collection are presented in Fig. 3.1.

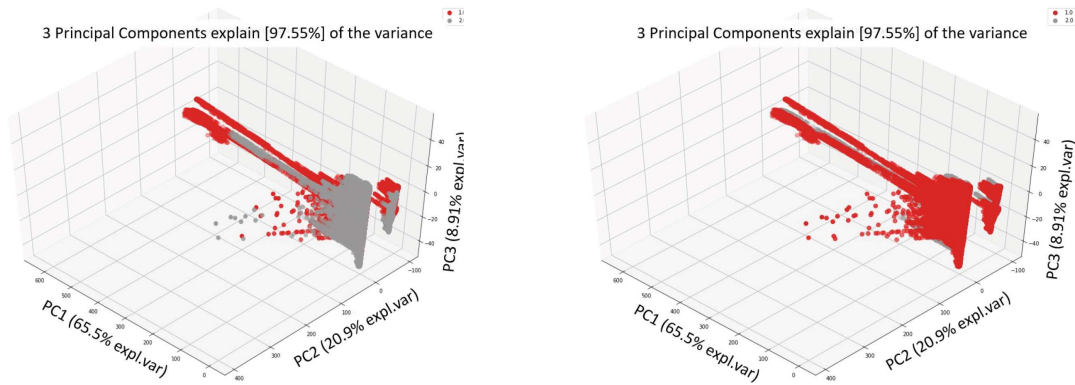
The final clean dataset includes 22 time-domain features representing vibrational and gyroscopic data. Overall, we have collected 27,000 data points that can be used for further analyses of surface classification. This dataset includes 15 surface types/classes, of which, 3 classes have a relatively less number of data points as displayed in Fig. 3.2, making the classification task challenging. Also, since the dataset is manually annotated, there is a possibility that the dataset contains some amount of label noise.

## 2.2 CASE

The Continuously Annotated Signals of Emotion (CASE) [19] dataset contains continuous emotion annotations provided by the participants while watching various videos. This dataset also includes participant’s recorded physiological reactions to the videos. These physiological measurements were synchronized and sampled at 1000 Hz from Electrocardiograph (ECG), Blood Volume Pulse (BVP), Galvanic Skin Response (GSR), Respiration (RSP), Skin Temperature (SKT), and Electromyography (EMG) sensors. This dataset is based on the 2D circumplex model of emotion that depicts different valence and arousal levels on the coordinate X-axis and Y-axis respectively. The participants used a novel Joystick-based Emotion Reporting Interface (JERI) on this 2D plane to report annotations sampled at 20 Hz. The participants included 15 males and 15 females aged between 22 and 37 from different cultural backgrounds.



**Fig. 3.** (1) Chairs (a) and (b) were used in the USA, and (c) in China. Green dots indicate where the smartphone was attached. (2) Wheelchair dataset class distribution. (Color figure online)



**Fig. 4.** CASE dataset distribution based on classes 1 and 2, representing low and high levels of (a) arousal and (b) valence respectively.

Our final dataset consists of 8 real-valued features corresponding to the physiological reactions of the participants and has two classes valence and arousal with low ( $\leq 5$ ) and high ( $> 5$ ) levels. Also, we have converted the raw annotation scores to low and high valence and arousal values such that they map to one of the four quadrants of the circumplex plane [16]. Though the dataset contains outliers as shown in Fig. 4, we demonstrate our method to be robust to label noise and outliers.

### 3 SSLAM: Self-supervised Label Generation Framework

Our proposed framework incorporates a novel activation function and loss function as an improvement over the current state-of-the-art self-supervised framework for tabular data (VIME) [24]. We employ two pretext tasks that are, feature vector estimation and mask vector estimation to train an encoder in a self-supervised manner as shown in Fig. 5. These tasks employ two predictors using the input vector’s encoder representations. The task of the first predictor

model is to recover the original input feature vector from its corrupted variant produced using a mask vector. The task of the second predictor is to predict the mask vector. The pretext tasks are solved using the below models,

(i) Mask vector estimator,  $s_m : \mathcal{Z} \rightarrow [0, 1]^d$ , takes the encoder embedding  $\mathbf{z}$  as input and predicts a mask vector  $\hat{\mathbf{m}}$ .

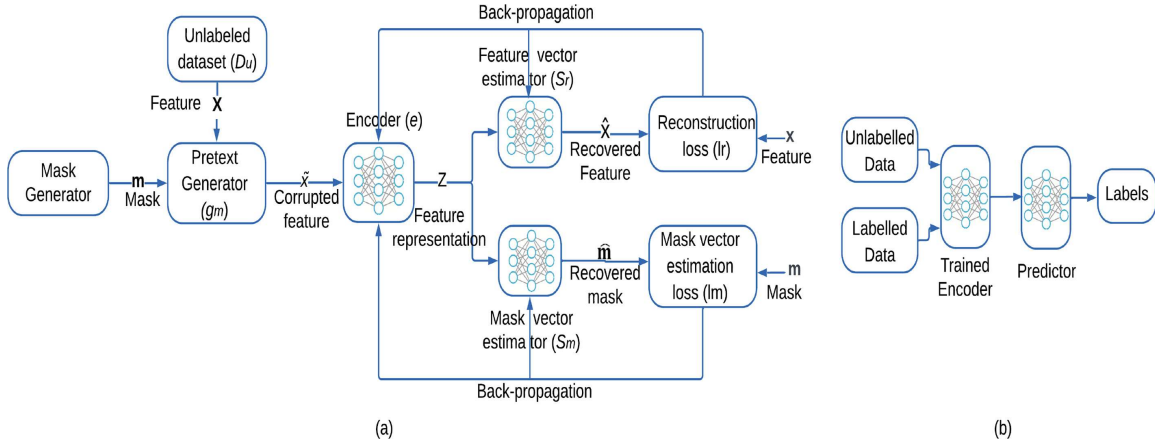
(ii) Feature vector estimator,  $s_r : \mathcal{Z} \rightarrow \mathcal{X}$ , takes the encoder embedding  $\mathbf{z}$  as input and predicts  $\hat{\mathbf{x}}$  for the input feature vector  $\mathbf{x}$ .

Mask vector estimation task uses a mask vector generator to produce a binary mask vector  $\mathbf{m} = [m_1, \dots, m_d]^\top \in \{0, 1\}^d$  where  $m_i$  is randomly sampled from a Bernoulli distribution with a probability  $p_{mask}$ . The pretext generator  $g_m : \mathcal{X} \times \{0, 1\}^d \rightarrow \mathcal{X}$  utilizes a mask vector  $\mathbf{m}$  and samples  $\mathbf{x}$  from the large unlabeled dataset  $\mathcal{D}_u$  as input, and generates a corrupted sample  $\tilde{\mathbf{x}}$ . The corrupted feature is given by,  $\tilde{\mathbf{x}} = g_m(\mathbf{x}, \mathbf{m}) = \mathbf{m} \odot \bar{\mathbf{x}} + (1 - \mathbf{m}) \odot \mathbf{x}$  where the  $j$ -th feature of  $\bar{\mathbf{x}}$  is sampled from the empirical distribution  $\hat{p}_{X_j} = \frac{1}{N_u} \sum_{i=N_l+1}^{N_l+N_u} \delta(x_j = x_{i,j})$ . The pretext generator  $g_m$  is also a stochastic function whose randomness comes from  $\bar{\mathbf{x}}$ . Together this randomness makes reconstructing  $\mathbf{x}$  from  $\tilde{\mathbf{x}}$  a difficult task for the neural networks. The following optimization problem,  $\min_{e, s_m, s_r} \mathbb{E}_{\mathbf{x} \sim p_X, \mathbf{m} \sim p_{\mathbf{m}}, \tilde{\mathbf{x}} \sim g_m(\mathbf{x}, \mathbf{m})} [l_m(\mathbf{m}, \hat{\mathbf{m}}) + \alpha \cdot l_r(\mathbf{x}, \hat{\mathbf{x}})]$  where  $\hat{\mathbf{m}} = (s_m \circ e)(\tilde{\mathbf{x}})$  and  $\hat{\mathbf{x}} = (s_r \circ e)(\tilde{\mathbf{x}})$ , is used to train the encoder  $e$  and the pretext predictive models.

$$l_m(\mathbf{m}, \hat{\mathbf{m}}) = -\frac{1}{d} \left[ \sum_{j=1}^d m_j \log \left[ (s_m \circ e)_j(\tilde{\mathbf{x}}) \right] + (1 - m_j) \log \left[ 1 - (s_m \circ e)_j(\tilde{\mathbf{x}}) \right] \right]$$

is the first loss function which is the sum of the binary cross-entropy losses for each dimension of the mask vector. The second loss function  $l_r$  is the proposed novel *log - cosh* reconstruction loss,  $l_r(\mathbf{x}, \hat{\mathbf{x}}) = \frac{1}{d} \left[ \sum_{j=1}^d \text{logcosh} \left( x_j - (s_r \circ e)_j(\tilde{\mathbf{x}}) \right) \right]$ . We propose a parameterized version of the Elliot activation function to be used in the hidden layers of the encoder to yield a better representation, which is computed as follows:  $f(\omega_j^{\text{in}}, \beta_j, x_i, \lambda) = k_1 + \frac{k_2 \cdot (\omega_j^{\text{in}} \cdot x_i + \beta_j) \cdot \lambda}{1 + |(\omega_j^{\text{in}} \cdot x_i + \beta_j) \cdot \lambda|}$  where  $\lambda$  is the slope of the function and  $k_1$  and  $k_2$  are the parameters learned through back-propagation during the training of this network.

The encoder here is a neural network that maps the input data to a fixed-length vector representation. The multilayer neural network used by SSLAM encoder framework has one hidden layer with a novel activation function. The difficulty of the pretext tasks can easily be controlled through the multiple hyper-parameters of the framework such as the probability  $p_{mask}$  can be tuned to adjust the proportion of the corrupted features. The hyper-parameter  $\alpha$  is also tuned to weigh the loss from the two pretext tasks. VIME [24] has proposed the optimal values for these parameters using cross-validation. Due to the way the encoder has been trained, the representations  $\mathbf{z}$  contain information about imputing corrupt features and identifying the corrupted features. This informative representation of the input data, reduces model complexity to minimize the losses in comparison to the raw input feature data, resulting in more accurate predictions.



**Fig. 5.** (a) Block diagram illustrating the SSLAM framework. (b) Generating labels employing a trained encoder and predictor network.

The availability of labeled and unlabeled data is the primary consideration for our proposed framework. We elaborate on the explanation of our framework using the CASE dataset. The CASE dataset has 1.5 million instances of annotated emotion data for valence and arousal classes. Significant expenses are associated with annotating these data points, which we aim to reduce using our label-generation framework. A large proportion of unlabeled data is required for our method, thus, for our study, we split our dataset in the ratio of 1:9 labeled and unlabeled data points. The SSLAM framework utilizes these data in the following way: the encoder takes unlabeled data as input and converts it into informative homogeneous representations. It is then trained to minimize the cross-entropy and reconstruction loss functions associated with the mask and feature vector estimation tasks respectively. To adequately recover the input features  $x$ , we require the encoder to output latent representation  $z$ . To achieve this, the correlation between the input features of  $x$  needs to be captured. This is exactly what the encoder does.  $s_m$  can utilize the inconsistencies between feature values to identify the masked features, while  $s_r$  can learn from the correlated non-masked features to attribute the masked features. The encoder, therefore, learns that if a particular feature has a different correlation from the others, it may be masked and corrupted.

This information is useful for the next downstream task of transforming the remaining labeled data points into better homogeneous and informative representations. These transformed representations are then fed into the predictive model, to better predict the class labels of the input test data. We, thus apply this framework to our split of unlabeled and labeled data to generate new labeled data points. These artificially generated labels can be added back to the original labeled set and the process can be iterated to produce more annotated data.

To summarise, our data goes through the following steps in the framework:

- Acquire labeled and unlabeled data points where the proportion of unlabeled data points is considerably larger.



- The encoder is fed with unlabeled data points to learn better representations of the data by solving two pretext tasks.
- Post training, the encoder is fed with labeled data to generate a homogeneous and informative representation for further downstream classification tasks.
- The encoder representations train a predictive model using the labeled data.
- The learned representations are then utilized to predict new class labels on test data.
- This newly generated labeled data can be mixed with the original labeled dataset and the process can be iterated over to produce more labels.

### 3.1 Log-Cosh Loss in SSLAM Framework

**Mathematical Framework:** We now justify the proposal of Loss,  $L(x) = \log(\cosh(x))$  in an encoder setup. Using  $\log - \cosh$  as the reconstruction loss in the encoder setup is supported by additional analytical properties, such as convexity, smoothness, robustness to outliers, etc. Let  $L(x)$  be the loss function with  $x$  being the input to the loss function. Then for the symmetric version of the loss function,  $L(x) = \log(\cosh(x))$ .

**MAE and MSE as Siblings:** The expression for the loss function is as follows:  $E(x, y) = \sum_{i=1}^m \log(\cosh(y_j - w^T x_j))$  for training examples  $(x_j, y_j)$  for  $j = 1$  to  $m$ , where  $y_j$  is the actual value of the  $j^{th}$  training example from the dataset. Using Taylor Series approximation it can be shown that  $E(x, y) = \sum_{i=1}^m \log(\cosh(y_j - w^T x_j))$  is mathematically equivalent to Mean Absolute Error (MAE) and Mean Squared Error (MSE) respectively for large  $x$  away from 0 or for small  $x$  nearer to 0. Since, MSE is the preferred reconstruction loss in VIME, we show the impact of the proposed loss function in comparison to the current SoTA, VIME. Since, we know that MAE is 1- Lipschitz [11]. For large  $x$ , our loss function behaves like MAE, thus we can argue that like MAE,  $\log - \cosh$  is robust to outliers. Our loss function therefore inherits identical robustness to label noise as MAE. For small  $x$ ,  $\log - \cosh(x)$  inherits properties of MSE. Consequently, our proposed  $\log - \cosh$  combines the smoothness of MSE and the robustness of MAE, making itself highly suitable for machine learning applications such as the self-supervised approach proposed here. This establishes  $\log - \cosh$  as a suitable alternative for MSE ((similar to VIME) in the encoder setup of the SSLAM framework.

**How did the Loss Function Come About?** The preceding discussion establishes the  $\log - \cosh$  function and explains its effectiveness in various contexts. However, it does not necessarily prove its relevance as a reconstruction loss in the encoder setup. The primary objective of deep learning is to gain knowledge about the manifold structure present in the data (i.e. natural high dimensional data that converges to a non-linear low dimensional manifold). It also involves understanding the probability distribution associated with the manifold. An encoder learns low dimensional data and represents data as a parametric manifold i.e. a piece-wise linear map from latent to the ambient space.

**Logcosh(x) in VAE - A Distributional Insight:** We define the encoder and decoder as:

- Encoder  $\varphi : \chi \rightarrow F$  maps  $\Sigma$  to its latent representation  $D = \varphi(\Sigma)$  homeomorphically.
- Decoder  $\psi: F \rightarrow \varphi$  maps  $z$  to reconstruction  $\tilde{x} = \psi(z) = \psi \circ \varphi(x)$

$\varphi \circ \psi = \operatorname{argmin}_{\varphi, \psi} \int_{\chi} L(x, \psi \circ \varphi(x)) dx$ , where  $\chi$  is the ambient space,  $F$  is the latent space,  $L$  is the loss function and  $\Sigma$  is a topological space  $\Sigma \subset \bigcup_{\alpha} U_{\alpha}$ . We constructed distributions using pseudo-hyperbolic Gaussian, resulting in the reconstruction loss for Variational AutoEncoders (VAEs), defined as  $\log\cosh(x)$ , serving as our loss function.

**Pseudo-Hyperbolic Gaussian:** The strategy to generate the pseudo-hyperbolic Gaussian ((Wrapped gaussian distribution  $G(\mu, \Sigma)$  on hyperbolic space  $\mathbf{H}$ )) is as follows:

- Sample a  $\mathbf{v}$  from normal distribution  $N(0, \Sigma)$  defined over  $\mathbf{R}^n$ .
- Interpret  $\mathbf{v}$  as an element of  $T_{\mu}\mathbf{H}^n \subset \mathbf{R}^{n+1}$  by rewriting  $\mathbf{v}$  as  $\mathbf{v} = [0, \mathbf{v}]$ .
- Parallel transport vector  $v$  to  $u \in T_{\mu}\mathbf{H}^n \subset \mathbf{R}^{n+1}$  along the geodesic from  $\mu_0$  to  $\mu$ .
- Map  $u$  to  $\mathbf{H}^n$  using  $\exp(u) = \cosh(\|u\|_L) + \sinh(\|u\|_L) \frac{u}{\|u\|_L}$

Reconstruction Loss is thus  $-\mathbf{E}_{q_{z|x}} \log(p_{\theta}(x|z))$ . Replacing  $p_{\theta}(x|z)$  with pdf of Hyperbolic secant distribution:  $= -\log(\frac{1}{2} \operatorname{sech}(\frac{\pi x}{2})) = \log(2 \cosh(\frac{\pi x}{2})) = \log(\cosh(y))$  where  $y = \frac{\pi x}{2}$ . Since the metric at the tangent space coincides with the Euclidean metric, several distributions can be produced by applying the construction strategy such as  $\log\cosh(x)$ .

## 4 Evaluation

### 4.1 Experimental Configuration

To evaluate the performance of SSLAM, we test it on two tabular datasets from the domain of affective computing: CASE and wheelchair. For all our experiments, we randomly divide our dataset into an - (a) 85-15% and (b) 80-20% train-test split. Later, we split our training data into 10-90% labeled and unlabeled data. We evaluate our proposed model against four baseline models, three of which were used as baselines in VIME. SSLAM is different from a classical supervised classification problem and therefore most of the SOTA baselines don't apply to the setting proposed here. The first baseline model is a simple MLP trained using only labeled data in a supervised manner. Our second baseline is a simple logistic regression model. XGBoost, a tree-based classification method is our third baseline. Our final baseline is a self-supervised model VIME [24] with state-of-the-art performance results for classification tasks in the tabular domain. The self-supervised models are pre-trained on the unlabeled data and used along with the labeled data for classification tasks.

We have used the same encoder architecture in SSLAM as in VIME as depicted in Table 1. Based on the experiments conducted in VIME we have

**Table 1.** Architecture details of SSLAM

Module	Layer Details	Layer Dimensions
Input	-	$8 \times 1$
Encoder	[dense] $\times$ 1 + Parameterized Elliot	(8,8)
Feature vector estimator	[dense] $\times$ 1 + Linear	(8,8)
Mask vector estimator	[dense] $\times$ 1 + Sigmoid	(8,8)
Predictor	[dense] $\times$ 1 + ReLu	(8,100)
	[dense] $\times$ 4 + ReLu	(100,100)
	[dense] $\times$ 1 + Sigmoid	(100,2)
Output	-	2

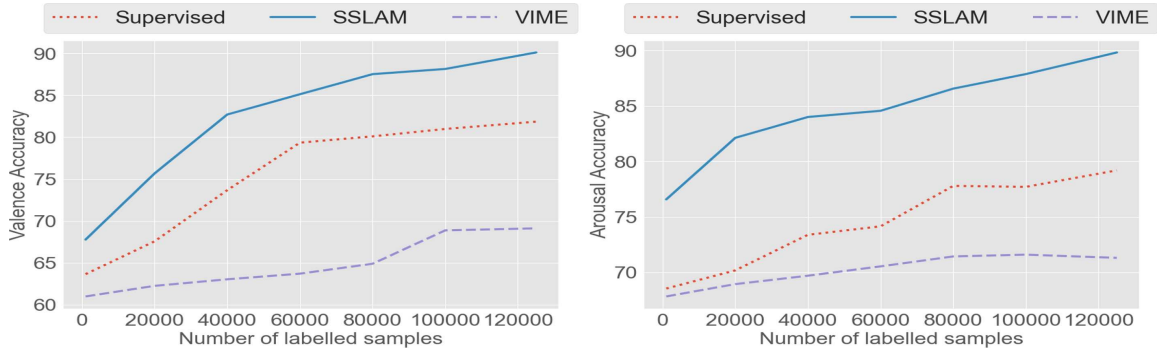
tuned the model parameters  $p_{mask}$  and  $\alpha$  to 0.3 and 2 respectively. The activation function corresponding to the feature vector estimation is set to linear activation function whereas the layer corresponding to mask estimation has a sigmoid activation function. The reconstruction loss and mask estimation loss are the novel  $log - cosh$  and binary cross-entropy losses respectively. The encoder is trained using an RMSprop optimizer with a learning rate of 0.001 on both loss functions. Our baseline Multilayer Perceptron (MLP) and feedforward neural network (FNN) used in the predictor network has five hidden layers each with hidden dimension 100. These hidden layers are set to have a ReLU activation function while the output layer has a Softmax activation function. Both are trained using an Adam optimizer with a learning rate of 0.001 on the categorical cross-entropy loss function. The supervised feedforward neural model is fine-tuned with early stopping (patience 50), and we allocate 10% of the training data as the validation split. All models are trained with a batch size of 128. We train the feedforward neural predictor network for 100 epochs and the encoder for 10 epochs. To enhance our model’s performance, we utilized a parameterized Elliot activation function in the encoder.

**Fine-Tuning Hyperparameters:** We have performed several experiments with varying the hyperparameters of the predictor FNN on both datasets. First, we conducted experiments to vary the dimensions of the hidden layers in the FNN to 100 and noted its performance for the classification tasks as shown in Appendix A.1. We found that for both datasets used in this study 100 neurons in the hidden layer is the best choice. To avoid incurring high computational costs, we do not exceed this number.

Next, for each dataset, we experiment with the number of hidden layers in FNN. Figure in Appendix A.1 and Table 2 indicate that 5 is the optimal choice for both CASE and wheelchair datasets. Again, we restrict our experiments to 5 layers. Ultimately, this study aims to illustrate the comparative performance of the proposed model over the baselines on both datasets, which is achieved by

**Table 2.** Performance comparison for different number of layers for each of the datasets

Datasets	Number of Hidden Layers				
	1	2	3	4	5
CASE: Valence	77.2329	82.1172	86.0457	87.5545	<b>90.0178</b>
CASE: Arousal	76.8658	82.1246	85.4126	87.2266	<b>89.7985</b>
Wheelchair	61.6047	64.7465	65.8745	66.1679	<b>71.6529</b>

**Fig. 6.** Comparison of Accuracy of predictions of (a)Valence and (b)Arousal across different sizes of labeled CASE dataset made by SSLAM and other baselines.

our experimental setup. We have executed each of the models 5 times utilizing different random train/validation/test splits and seeds and the average of these results has been reported. As with previous studies on tabular data, we use accuracy as our evaluation metric in all experiments.

## 4.2 Results: CASE Dataset

In this section, we will be discussing the results of the SSLAM on the CASE dataset, and we will also be comparing its performance with the baselines that we had defined earlier. The classification results of Valence and Arousal accuracy on 85-15% train-test split are indicated in Table 3. The MLP baseline produces 81.3% accuracy for Valence, which is better than logistic regression, XGBoost and VIME. But, our proposed framework vastly outperforms these baselines, producing an accuracy of 90.01%. Also, SSLAM outperforms all of these baselines including VIME for Arousal by generating an accuracy of 89.79%. We obtain similar results on the 80-20% split as displayed in Table 3 with SSLAM outperforming all baselines on both Valence (89.49%) and Arousal (88.84%). In Fig. 6, we compare the performance of (valence and arousal prediction accuracy) of a supervised MLP, VIME and SSLAM against the increasing number of labeled data points (x-axis). The proposed approach (SSLAM) outperforms both the baselines, i.e. the supervised MLP and the self-supervised VIME on the CASE dataset.



**Table 3.** Comparison of Accuracy for predicting Valence and Arousal on CASE dataset using the 85-15% and 80-20% splits

Model Type	Accuracy using 85-15% split		Accuracy using 80-20% split	
	Valence	Arousal	Valence	Arousal
MLP	0.8130 $\pm$ 0.0040	0.7993 $\pm$ 0.0013	0.8034 $\pm$ 0.0021	0.7911 $\pm$ 0.0057
Logistic Regression	0.6901 $\pm$ 0.0012	0.6806 $\pm$ 0.0010	0.6877 $\pm$ 0.0028	0.6899 $\pm$ 0.0025
XGBoost	0.7330 $\pm$ 0.0009	0.7339 $\pm$ 0.0022	0.8467 $\pm$ 0.0041	0.7343 $\pm$ 0.0015
VIME	0.6917 $\pm$ 0.0051	0.7213 $\pm$ 0.0042	0.7197 $\pm$ 0.0027	0.7093 $\pm$ 0.0027
SSLAM	<b>0.9001 <math>\pm</math> 0.0024</b>	<b>0.8979 <math>\pm</math> 0.0045</b>	<b>0.8949 <math>\pm</math> 0.0046</b>	<b>0.8884 <math>\pm</math> 0.0073</b>

**Table 4.** Comparison of Accuracy for predicting Valence and Arousal on Wheelchair dataset using the 85-15% and 80-20% splits

Model Type	85-15% split		80-20% split	
	Accuracy	F1 score	Accuracy	F1 score
MLP	0.6740 $\pm$ 0.0211	0.6704 $\pm$ 0.0054	0.6347 $\pm$ 0.0294	0.6309 $\pm$ 0.0231
Logistic Regression	0.4463 $\pm$ 0.0020	0.4063 $\pm$ 0.0012	0.4307 $\pm$ 0.0054	0.4237 $\pm$ 0.0073
XGBoost	0.6305 $\pm$ 0.0089	0.6304 $\pm$ 0.0029	0.6216 $\pm$ 0.0019	0.6193 $\pm$ 0.0147
VIME	0.6366 $\pm$ 0.0393	0.6065 $\pm$ 0.0381	0.6283 $\pm$ 0.0317	0.6267 $\pm$ 0.0318
SSLAM	<b>0.7165 <math>\pm</math> 0.0054</b>	<b>0.7120 <math>\pm</math> 0.0067</b>	<b>0.7074 <math>\pm</math> 0.0059</b>	<b>0.7040 <math>\pm</math> 0.0093</b>

### 4.3 Results on Wheelchair Dataset

We have used the same experimental setup for the Wheelchair dataset analysis as for the CASE dataset. The comparison outcomes of our model with the baselines on the wheelchair data for the 85-15% and 80-20% splits are presented in Table 4. We observe that the SSLAM model outperforms all other baselines for both train-test splits. SSLAM achieves an accuracy of 71.65% on the 85-15% split, while on the 80-20% split, it achieves an accuracy of 70.74%. The wheelchair dataset has unbalanced classes; thus, we also report the weighted F1 score. From the above two tables, it is clear that the F1 score represents a similar trend where SSLAM outperforms the other two baselines.

In the case of the wheelchair dataset, all the models have performed poorly due to the limited size of the dataset. For all our models we use the same amount of labeled and unlabeled data. Results from the CASE dataset demonstrate that as we increase the number of unlabeled samples, the encoder’s representations improve thus resulting in optimal classification accuracy. The wheelchair dataset is relatively small in size, which means that the number of unlabeled samples ( $\leq 100,000$ ) is not enough to help the encoder generalize well and learn good representations of the inputs. As a result, the performance of the models is not significantly improved. Additionally, the dataset’s class imbalance problem further hinders the model’s performance.

## 5 Discussion

We demonstrated that the SSLAM framework outperformed other baselines on both datasets. Our proposed methodology is best suited when dealing with large amounts of unlabeled data where annotating the data is tedious and expensive. This is often the case in real-world scenarios such as annotating surface-induced vibration data for wheelchair users and emotional data from physiological sensors. Thus, SSLAM can efficiently be employed to generate meaningful representations from the unlabeled samples and to generate labels reducing large annotation overhead.

The **Role of  $\log - \cosh$  in SSLAM:** is evident from the performance of the framework in comparison to the standard MSE in the encoder set-up. The use of  $\log - \cosh$  delivers significant improvements across both datasets - CASE and Wheelchair. The largest increase in performance has been observed in the CASE dataset. This dataset has an outlier problem and  $\log - \cosh$  being robust to outliers overcomes this considerably.

## 6 Relevant Literature

We discuss our research literature in three parts: continuous emotion annotation techniques and their drawbacks, limited data annotation emotion recognition methods, and the effectiveness of different self-supervised approaches on tabular data.

**Continuous Emotion Annotation:** In the existing literature, the most widely adopted approach of emotion annotation using self-report is the post-interaction or post-stimuli one, where the participants after watching the video provide emotion self-reports based on a standard scale (e.g., Self-assessment Manikin (SAM) [2]). However, in the post-stimuli approach capturing intra-video subtle nuances and time-aligning all the emotions is challenging. To address these issues, researchers use continuous emotion annotation strategies, where participants continuously provide emotion annotations as they watch the videos using a mouse, a joystick or another similar device [5, 8, 28]. Similarly, the CASE [19] dataset involved participants who used a joystick to provide continuous annotations of their emotions, specifically *valence* and *arousal*, based on the Circumplex Model of emotion [16]. Yet, the challenges with these approaches are the following - (a) for emotion annotations they require the users to utilize an auxiliary device, (b) due to the continuous nature of emotion annotation and video consumption in parallel, the cognitive load increases and the viewing experience degrades.

**Recognizing Emotions with Limited Data:** Numerous studies in affective computing have attempted to tackle the issue of the restricted availability of labeled data. Chen et al. (2021) [4] proposed a CNN method to tackle the problem of limited samples and imbalanced datasets for emotion recognition on the DEAP dataset through a data augmentation algorithm called the Borderline-SMOTE. They achieved a performance of 97.47% and 97.76% on valence and

arousal prediction tasks. Zhang et al. (2022) [29] address the issue of data scarcity in EEG data by proposing a data augmentation method called generative adversarial network-based self-supervised data augmentation (GANSER) to perform emotion recognition. Their model synthesizes simulated EEG signals that do not skew from the underlying data distribution, which helps to perform well on emotion classification tasks. SigRep [6] produces performances for arousal (76.3%) and valence (74.1%) accuracy through a contrastive learning-based self-supervised technique using the data obtained from wearable devices. Tianyi et al. (2020) [27] propose a correlation-based emotion recognition algorithm (CorrNet) that employs an autoencoder to perform automatic feature extraction of signals generated by wearables. The model proposed by Tang et al. (2017) [21] for valence and arousal emotion classification on SEED and DEAP datasets used a denoising autoencoder. Subramanian et al. (2018) [20] learn features from electrocardiogram (ECG) data using a Naive Bayes classifier and Support Vector Machine (SVM). Sarkar et al. (2022) [17] propose a self-supervised multi-task CNN framework to learn ECG representations using pretext tasks.

**Self-supervised Learning on Tabular Data:** Some recent approaches propose using self-supervised learning techniques that utilize existing unlabeled data to discover broad feature representations specific to the data. In computer vision [10, 26] and language modeling tasks [14], these approaches have proven to be fairly successful due to the underlying spatial, syntactic or semantic structure of the image or language data. Regardless, these approaches are not very effective for tabular data and sparse literature exists on handling tabular data using these methods. Recent studies focus on solving pretext tasks. Yoon et al. [24] proposed a self-supervised framework called Value Imputation and Mask Estimation (VIME) which employs two pretext tasks to train an encoder. The pretext generator is fed a random binary mask and unlabeled tabular data samples. This setup results in unlabeled samples that are corrupted by the mask. Given the corrupted heterogeneous inputs to the encoder, it is trained to generate informative homogeneous representations. In this architecture, the encoder representation of the data is fed into the mask and feature estimators, which predict both the binary mask and the original uncorrupted input. These learned transformed representations are further provided to the predictive model to perform the main downstream task.

## 7 Conclusion and Future Works

We presented a framework SSLAM for self-supervised label generation for annotation overhead reduction. The framework trains an encoder in a self-supervised manner by implementing two pretext tasks using a contrastive sampling method. The structure of VIME inspires our approach, but we distinguish ourselves by employing a novel loss function ( $\log - \cosh$ ) compared to the denoising autoencoder loss used in VIME in the pre-training phase. Also, in the pre-training phase, we employ the parameterized Elliot activation function in the encoder to generate better representations to ensure more accurate predictions. Since we

present our model as an improvement over the VIME, we have employed the same baselines used in VIME. Also, we are comparing against VIME because it is *the* state-of-the-art method. The other SOTA methods are applicable on vision data (such as MixMatch and ReMixMatch). Therefore the efficacy of the proposed method is best compared with VIME.

We evaluated the framework to determine its effectiveness in reducing the continuous annotation overhead on two datasets: wheelchair and CASE. The framework showed better results compared to the state-of-the-art self-supervised approach and the supervised approach. We also observed that the framework can generalize across different use cases, as demonstrated in a large-scale surface classification dataset for wheelchair users. Additional experiments on KEemoCon, MNIST and Fashion-MNIST datasets produce similar SOTA results. Along with further theoretical considerations, we defer the additional details on the generalizability of SSLAM to future work.

In summary, our SSLAM provides superior performance over existing baselines in label generation, particularly when more unlabeled data is available. We attribute this improved performance to our novel reconstruction  $\log - \cosh$  loss that is employed by the encoder. The study results demonstrate the approach's potential to reduce annotation overhead in scenarios with imbalanced labeled and unlabeled data.

**Acknowledgments.** Snehanshu Saha, Surjya Ghosh and Sougata Sen would like to thank the Anuradha and Prashanth Palakurthi Center for Artificial Intelligence Research (APPCAIR), SERB-DST (SUR/2022/001965) and SERB CRG-DST (CRG/2023/003210), Govt. of India for partially supporting the work. Swarnali Banik gratefully acknowledges the Chanakya Fellowship from AI4CPS Innovation Hub, IIT Kharagpur.

## References

1. Abdel Hakim, A.E., Deabes, W.: Can people really do nothing? handling annotation gaps in adl sensor data. *Algorithms* **12**(10), 217 (2019)
2. Bradley, M.M., Lang, P.J.: Measuring emotion: the self-assessment manikin and the semantic differential. *J. Behav. Ther. Exp. Psychiatry* **25**(1), 49–59 (1994)
3. Chatterjee, S., Chakma, A., Gangopadhyay, A., Roy, N., Mitra, B., Chakraborty, S.: Laso: exploiting locomotive and acoustic signatures over the edge to annotate imu data for human activity recognition. In: *Proceedings of the 2020 International Conference on Multimodal Interaction*, pp. 333–342 (2020)
4. Chen, Y., Chang, R., Guo, J.: Effects of data augmentation method borderline-smote on emotion recognition of eeg signals based on convolutional neural network. *IEEE Access* **9** (2021)
5. Cowie, R., Douglas-Cowie, E., Savvidou, S., McMahon, E., Sawey, M., Schröder, M.: ‘FEELTRACE’: an instrument for recording perceived emotion in real time. In: *ITRW Speech-Emotion* (2000)
6. Dissanayake, V., Seneviratne, S., Rana, R., Wen, E., Kaluarachchi, T., Nanayakkara, S.: Sigrep: toward robust wearable emotion recognition with contrastive representation learning. *IEEE Access* **10**, 18105–18120 (2022)



7. Garcia-Ceja, E., Riegler, M., Nordgreen, T., Jakobsen, P., Oedegaard, K.J., Tørresen, J.: Mental health monitoring with multimodal sensing and machine learning: a survey. *Pervasive Mob. Comput.* **51**, 1–26 (2018)
8. Girard, J.M., Wright, A.G.: Darma: software for dual axis rating and media annotation. *Behav. Res. Methods* **50**(3), 902–909 (2018)
9. Hossain, H.S., Khan, M.A.A.H., Roy, N.: Active learning enabled activity recognition. *Pervasive Mob. Comput.* **38**, 312–330 (2017)
10. Jing, L., Tian, Y.: Self-supervised visual feature learning with deep neural networks: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **43**(11) (2020)
11. Mediratta, I., Saha, S., Mathur, S.: Liparelu: arelu networks aided by lipschitz acceleration. In: 2021 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE (2021)
12. Muralidharan, A., Gyongyi, Z., Chi, E.: Social annotations in web search. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 1085–1094 (2012)
13. Nowak, S., Rüger, S.: How reliable are annotations via crowdsourcing: a study about inter-annotator agreement for multi-label image annotation. In: Proceedings of the International Conference on Multimedia Information Retrieval, pp. 557–566 (2010)
14. Qiu, X., Sun, T., Xu, Y., Shao, Y., Dai, N., Huang, X.: Pre-trained models for natural language processing: a survey. *SCIENCE CHINA Technol. Sci.* **63**(10), 1872–1897 (2020)
15. Ronao, C.A., Cho, S.B.: Human activity recognition with smartphone sensors using deep learning neural networks. *Expert Syst. Appl.* **59**, 235–244 (2016)
16. Russell, J.A.: A circumplex model of affect. *J. Pers. Soc. Psychol.* **39**(6), 1161 (1980)
17. Sarkar, P., Etemad, A.: Self-supervised ecg representation learning for emotion recognition. *IEEE Trans. Affective Comput.* (2020)
18. Settles, B.: Active learning literature survey. Technical Report, University of Wisconsin-Madison Department of Computer Sciences (2009)
19. Sharma, K., Castellini, C., van den Broek, E.L., Albu-Schaeffer, A., Schwenker, F.: A dataset of continuous affect annotations and physiological signals for emotion analysis. *Sci. Data* **6**(1), 1–13 (2019)
20. Subramanian, R., Wache, J., Abadi, M.K., Vieriu, R.L., Winkler, S., Sebe, N.: Ascertain: emotion and personality recognition using commercial sensors. *IEEE Trans. Affect. Comput.* **9**(2), 147–160 (2016)
21. Tang, H., Liu, W., Zheng, W.L., Lu, B.L.: Multimodal emotion recognition using deep neural networks. In: International Conference on Neural Information Processing, pp. 811–819. Springer (2017)
22. Wang, Y., Nazir, S., Shafiq, M.: An overview on analyzing deep learning and transfer learning approaches for health monitoring. *Computational and Mathematical Methods in Medicine* **2021** (2021)
23. Yang, J., Fan, J., Wei, Z., Li, G., Liu, T., Du, X.: Cost-effective data annotation using game-based crowdsourcing. *Proc. VLDB Endowment* **12**(1), 57–70 (2018)
24. Yoon, J., Zhang, Y., Jordon, J., van der Schaar, M.: Vime: extending the success of self-and semi-supervised learning to tabular domain. *Adv. Neural. Inf. Process. Syst.* **33**, 11033–11043 (2020)
25. Yu, H., Raychoudhury, V., Saha, S., Edinger, J., Smith, R.O., Gani, M.O.: Automated surface classification system using vibration patterns—a case study with wheelchairs. *IEEE Trans. Artif. Intell.* **4**(4), 884–895 (2023). <https://doi.org/10.1109/TAI.2022.3190828>

26. Zhai, X., Oliver, A., Kolesnikov, A., Beyer, L.: S4l: self-supervised semi-supervised learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2019)
27. Zhang, T., El Ali, A., Wang, C., Hanjalic, A., Cesar, P.: Corrnet: fine-grained emotion recognition for video watching using wearable physiological sensors. *Sensors* **21**(1), 52 (2020)
28. Zhang, T., El Ali, A., Wang, C., Hanjalic, A., Cesar, P.: Rcea: real-time, continuous emotion annotation for collecting precise mobile video ground truth labels. In: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, pp. 1–15 (2020)
29. Zhang, Z., Zhong, S.h., Liu, Y.: Ganser: a self-supervised data augmentation framework for eeg-based emotion recognition. *IEEE Trans. Affective Comput.* (2022)