From Vision to Motion: Translating Large-Scale Knowledge for Data-Scarce IMU Applications

Hyungjun Yoon, Hyeongheon Cha, Hoang C. Nguyen, Taesik Gong, and Sung-Ju Lee, *Fellow, IEEE*

***Abstract*—Pre-training representations acquired via self-supervised learning could achieve high accuracy on even tasks with small training data. Unlike in vision and natural language processing domains, pre-training for IMU-based applications is challenging, as there are few public datasets with sufficient size and diversity to learn generalizable representations. To overcome this problem, we propose IMG2IMU that adapts pre-trained representation from large-scale images to diverse IMU sensing tasks. We convert the sensor data into visually interpretable spectrograms for the model to utilize the knowledge gained from vision. We further present a sensor-aware pre-training method for images that enables models to acquire particularly impactful knowledge for IMU sensing applications. This involves using contrastive learning on our augmentation set customized for the properties of sensor data. Our evaluation with four different IMU sensing tasks shows that IMG2IMU outperforms the baselines pre-trained on sensor data by an average of 9.6%p F1-score, illustrating that vision knowledge can be usefully incorporated into IMU sensing applications where only limited training data is available.**

***Index Terms*—Mobile sensing, deep learning, self-supervised learning, contrastive learning.**

I. Introduction

MOBILE sensing powered by deep learning has enabled various ubiquitous applications in everyday life. Motion sensing with inertial measurement units (IMUs), such as accelerometers, is particularly promising due to its broad applicability, including activity recognition [1], transportation [2], agriculture [3], and healthcare [4]. However, deep learning models for IMU sensing rely heavily on task-specific datasets, where the amount and diversity of labeled training data limit model performance. Collecting large-scale IMU data is challenging due to cost, device and user heterogeneity, and privacy concerns.

Recent research has focused on addressing label scarcity in deep learning by leveraging representation learning. A common strategy is self-supervised learning (SSL), which pre-trains models using large amounts of unlabeled data to capture general data characteristics through predefined tasks [5]. SSL has shown remarkable performance in domains with large public datasets. For example, in natural language processing, models such as Llama [6] and the GPT series [7] are pre-trained on massive Internet text corpora and serve as foundation models for various tasks. Similarly, in computer vision, pre-training on large-scale datasets such as ImageNet [8], JFT-3B [9], and LAION-5B [10] has led to state-of-the-art performance across a range of tasks [11].

In IMU sensing, pre-training with unlabeled sensor data can enhance downstream performance [12]. Nevertheless, unlike images and text that benefit from large-scale, diverse public datasets, existing IMU datasets [13], [14], [15] primarily focus on Human Activity Recognition (HAR) and lack diversity. For instance, Capture-24 [13] dataset collects data exclusively from wrist-worn devices at a single sampling rate, lacking diverse sensor types, placements, and signal processing methods. As a result, models pre-trained on such datasets face generalizability challenges, unable to adapt to tasks with varying targets, sensor positions, subjects, or sampling frequencies (see Section V-B1).

Motivated by this challenge, we leverage *external knowledge beyond sensor data* to tackle IMU sensing tasks. By transforming IMU data into 2D visual representations such as spectrograms [16], [17], patterns emerge that are visually interpretable through attributes such as brightness, shapes, and spatial structures. These attributes align naturally with the capabilities of vision models pre-trained on large-scale image datasets (as detailed in Section II).

Building on this intuition, we present IMG2IMU that translates the knowledge from pre-trained vision models to IMU sensing tasks. IMG2IMU transforms IMU data into spectrograms, mapping the three sensor axes to RGB channels, following established practices for visualizing sensor data [16], [17]. By fine-tuning pre-trained vision models with the spectrograms, IMG2IMU effectively handles IMU sensing tasks with scarce labeled data.

However, directly applying vision models introduces a domain gap. Unlike images, which are often invariant to transformations like rotations or flips, spectrograms encode spatiotemporal information and depend on the precise

Received 7 January 2025; revised 20 March 2025; accepted 26 March 2025. Date of publication 2 April 2025; date of current version 6 August 2025. This work was supported by the National Research Foundation of Korea (NRF), in part by the Korea government (MSIT) under Grant RS-2024-00337007, in part by the Institute of Information & communications Technology Planning & Evaluation (IITP), in part by the Korea Government (MSIT) under Grant 2024-00444862, in part by the Non-invasive near-infrared based AI technology for the diagnosis and treatment of brain diseases), in part by the the National Research Foundation of Korea(NRF), and in part by the Korea government (MSIT) under Grant RS-2025-00553241. Recommended for acceptance by D. Xu. *(Corresponding author: Sung-Ju Lee.)*

Hyungjun Yoon, Hyeongheon Cha, and Sung-Ju Lee are with the School of Electrical Engineering, KAIST, Daejeon 34141, South Korea (e-mail: hyungjun.yoon@kaist.ac.kr; hyeongheon@kaist.ac.kr; profsj@kaist.ac.kr).

Hoang C. Nguyen is with the Department of Computer Science, Stony Brook University, New York 11794 USA (e-mail: hcnguyen@cs.stonybrook.edu).

Taesik Gong is with the Department of Computer Science and Engineering and Artificial Intelligence Graduate School, UNIST, Ulsan 44919, South Korea (e-mail: taesik.gong@unist.ac.kr).

Digital Object Identifier 10.1109/TMC.2025.3556998

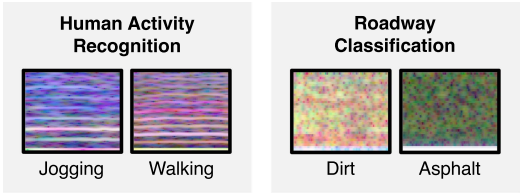


Fig. 1. Spectrogram images converted from triaxial IMU sensor data of human activity recognition and roadway classification tasks.

orientation of their axes—time and frequency—making such transformations disruptive. To bridge this gap, IMG2IMU incorporates a tailored pre-training method with *sensor-aware augmentations* that account for IMU-specific properties. We define four augmentations—*TranslateX*, *PermuteX*, *Hue*, and *Jitter*—to generate positive views for contrastive learning, ensuring the model learns robust, spectrogram-specific features during pre-training. We evaluate IMG2IMU on four diverse IMU sensing tasks and demonstrate its effectiveness in scenarios with limited training data. IMG2IMU consistently outperforms existing self-supervised methods, achieving a 9.6%p improvement in mean F1-score. These results underscore the potential of transferring knowledge from large-scale image datasets to enhance IMU sensing performance. The key contributions of this work are as follows:

- We propose IMG2IMU that leverages vision knowledge pre-trained on large-scale image datasets and translates it into IMU sensing applications on limited data.
- We design contrastive learning using four sensor-aware image augmentations that bridge the domain gap between image-based pre-training and IMU sensing tasks, enabling effective pre-training with images.
- We analyze the contribution of each augmentation to improving robustness against IMU-specific variations.
- We demonstrate through experiments that IMG2IMU enhances performance across diverse IMU sensing tasks in data-scarce scenarios.

II. Background and Motivation

A. Why Visualization Works: Interpretable Features

Data scientists often transform IMU data into visual representations (e.g., spectrograms) to improve interpretability [16], [18]. Visual representations are effective as they make latent features (e.g., frequency, amplitude, and temporal variation) perceptible through generally recognizable attributes, such as brightness, patterns, or colors. This approach minimizes the need for extensive domain knowledge of raw sensor data, enabling both human analysts [19] and machine learning models [17] to extract meaningful insights.

For example, in Fig. 1, the spectrograms of a human activity recognition (HAR) [1] task distinguish jogging and walking based on distinct patterns. Jogging exhibits a wider spacing between horizontal stripes, reflecting a higher motion frequency. Similarly, spectrograms from a roadway classification task [2] highlight differences in brightness, where darker plots correspond to dirt roads with irregular vibrations, in contrast to smoother asphalt surfaces.

This insight motivates our approach: complex sensing tasks can be solved by exploiting fundamental visual interpretation abilities, such as distinguishing colors or patterns, even without deep knowledge about the specific sensing task. In light of this intuition, we explore the potential to utilize the knowledge learned from vision to enhance IMU sensing tasks.

B. From Scarce Sensor Data to Abundant Image Data

Publicly available datasets for IMU sensing typically focus on specific tasks, such as daily activities [1], gait detection [4], or sports [20]. Although these datasets are valuable, they are often limited in scale and diversity. Larger-scale efforts [15], such as Capture-24 [13] and U.K.-Biobank [14], are restricted to specific conditions, such as wrist-worn devices measuring general daily activities. These datasets lack variety in sensor locations and tasks, leading to suboptimal generalization when applied to other sensing scenarios, as we demonstrate in Section V-B1.

In contrast, the field of computer vision has dramatically benefited from the availability of abundant data. Starting with ImageNet [8], which contains 1.2 million images in 1,000 classes, vision researchers have significantly scaled the size of the dataset. Examples include JFT-3B [9], which contains billions of images, and LAION-5B [10], a dataset of 5.85 billion images. These large-scale datasets provide a rich source of pre-trained knowledge, enabling vision models to generalize across diverse applications [21].

Vision models pre-trained on large-scale datasets excel at extracting foundational features, such as brightness, texture, and patterns [22], and have been successfully applied to domains beyond natural images (see Section III-C). Motivated by their success, we explore leveraging vision models trained on image datasets to address IMU sensing tasks.

C. Challenges in Bridging Vision and Sensing

While pre-trained vision models offer significant opportunities for sensor data analysis, applying them directly to sensor spectrograms introduces unique challenges. Unlike standard images, spectrograms encode critical information along the time and frequency axes, where orientation reflects the scale of values. Transformations such as rotation and flipping, which preserve labels for standard images, disrupt this information in spectrograms: rotation swaps the axes while flipping reverses axis values, as shown in Fig. 2. These distortions lead to misinterpretation and degraded performance when vision models are naïvely transferred to IMU sensing tasks.

To address this, we propose IMG2IMU, which adapts vision models to sensor data through task-specific augmentations and fine-tuning. This approach ensures that vision models effectively align with spectrogram-specific properties, allowing accurate interpretation of sensor data.

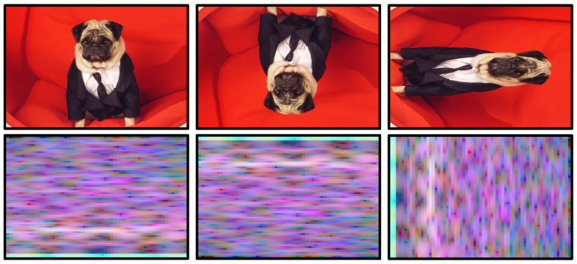


Fig. 2. Flipping and rotating an image from ImageNet (top) and a spectrogram image (bottom). Deformations misinterpret the spectrograms by swapping the time-frequency axes and inverting the values along an axis.

III. Related Work

A. Self-Supervised Learning for Sensing

Prior works [23], [24] applied self-supervised learning using a Multi-task transformation prediction for human activity recognition (HAR). Using Multi-task learning, the original data is augmented with a random augmentation, and the network is trained to predict the type of augmentation applied. SelfHAR [25] integrated the ideas of multi-task learning and teacher-student self-learning to create an effective semi-supervised learning framework.

Contrastive learning is another effective method where MoCo [26] and SimCLR [11], [27] are representative frameworks. They have been redesigned for HAR as MoCoHAR [28], SimCLR for HAR [29], and CSSHAR [30]. Another study [31], [32] adopted Contrastive Predictive Coding (CPC), which trains an encoder to predict the next sequence chunk based on previous sequences.

Masked region reconstruction [15], [33] is also adopted as a self-supervised learning strategy for sensory data. Haresaumudram, et al. [12] conducted an assessment of seven state-of-the-art self-supervised learning methods applied to HAR (e.g. SimSiam [34]) in addition to previously discussed methods.

While these studies showed their effectiveness for HAR tasks, IMU sensing applications include diverse target tasks [2] and subjects [3]. As publically available large-scale sensor datasets [13], [24] are centered on HAR, the pre-trained model for sensing has poor generalizability. IMG2IMU resolve this challenge by interpreting IMU sensor data as images and utilizing models pre-trained from a larger scale of vision data.

B. Use of Cross-Modal Data for Sensing

Prior studies have explored cross-modal data to enhance self-supervised learning for IMU sensing. ColloSSL [35] and COCOA [36] used cross-modal sensor data as positive view pairs for contrastive learning, while Vision2Sensor [37] employed vision-based activity recognition to generate labels for IMU data. However, these methods depend on synchronization between modalities, limiting their applicability to asynchronous settings. In contrast, IMG2IMU eliminates the need for synchronization by independently performing pre-training and fine-tuning.

IMU2Doppler [38] employed IMU data to train models for mmWave radar sensing. While this approach highlights the potential of leveraging IMU data for cross-modal applications, IMG2IMU addresses the scarcity of IMU sensor data by utilizing images for pre-training. Similarly, Tong et al. [39] used videos to construct semantic spaces for IMU-based activity recognition, focusing on zero-shot learning. Unlike IMG2IMU, their method specifically targets semantic embedding construction for HAR and does not explore generalizable pre-training strategies. IMUTube [40] and IMUGPT 2.0 [41] tackled data scarcity by generating virtual IMU data, using videos and textual descriptions, respectively. However, both approaches struggle to fully replicate real-world sensor noise and differ from IMG2IMU's focus on leveraging pre-trained vision models and addressing data scarcity through tailored pre-training strategies.

C. Using Pre-Trained Models From Images

Pre-trained models on large-scale image datasets, such as ImageNet [8] and JFT-3B [9], are highly effective for transfer learning [42]. These models have demonstrated exceptional performance across diverse tasks [11], including object detection [43] and semantic segmentation [44].

The versatility of image-based pre-trained models extends to diverse domains. For instance, Azizi et al. [45] employed ImageNet-pre-trained models for medical image analysis, including dermatology classification [46] and chest X-ray diagnosis [47]. Additionally, pre-trained vision models have been applied to sound classification [48] by converting audio data into mel-spectrograms. Building upon these, we designed pre-training methods tailored to IMU spectrograms.

Recent advancements in Vision-Language Models (VLMs) (e.g., using BLIP [49] and SAM [50]) further highlight the power of vision models. By mapping visual representations on semantic space, VLMs have achieved state-of-the-art performance in data-scarce scenarios [51]. For instance, VLMs have been used to classify sensor data by associating visual graphs with textual descriptions [52]. However, the large size of VLMs limits their applicability in resource-constrained settings. Our approach leverages lightweight networks and optimized pre-training strategies to effectively adapt vision-based models for sensor data.

IV. IMG2IMU

To enhance the performance of IMU sensing tasks when a fair amount of training data is difficult to obtain, we propose to utilize large-scale public image datasets to pre-train a model. Fig. 3 overviews our IMG2IMU that consists of two main stages: (i) pre-training a model using large-scale image datasets to learn sensor-aware knowledge through self-supervised contrastive learning, and (ii) transferring the learned knowledge from the vision model to downstream IMU sensing tasks that use 2D-transformed sensor data.

A. Converting Triaxial IMU Sensing Data to Images

Spectrograms display the intensity of frequency features along the time axis. Existing works [16], [17], [18] showed that the frequency-based visualization effectively represents features

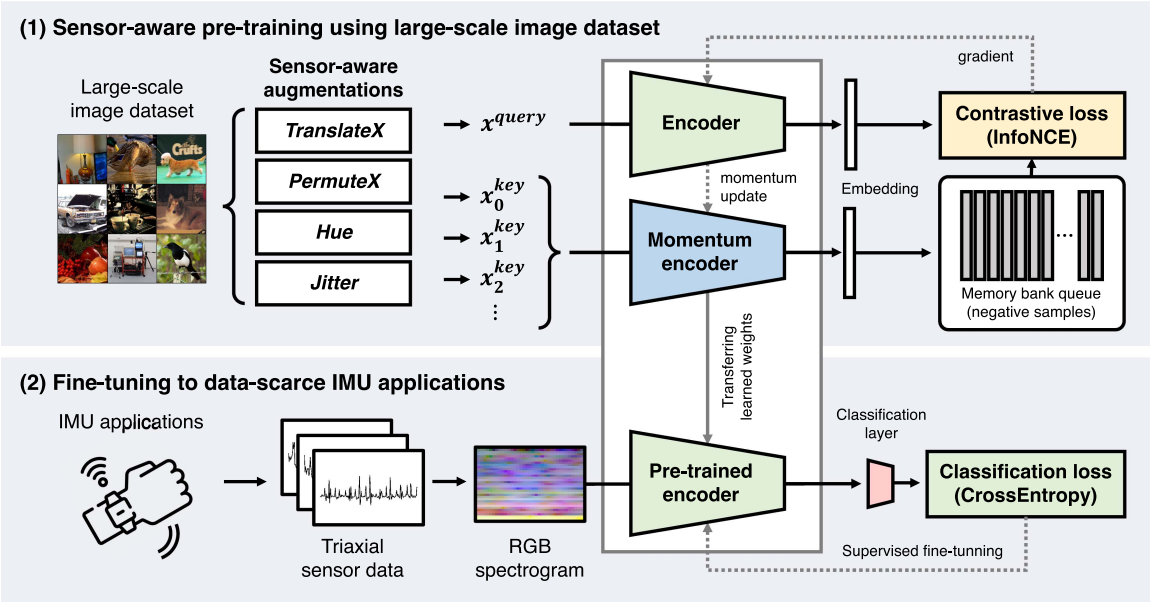


Fig. 3. Overview of IMG2IMU. (1) Pre-training is performed with the large-scale image dataset collected from the public domain, using contrastive learning with sensor-aware augmentations. (2) The pre-trained model is transferred to IMU sensing tasks using 2D-transformed triaxial IMU sensor data as input.

for various IMU sensing tasks. Building on this foundation, we set spectrograms as our primary visualization method, expecting that the ability to interpret visual features from images can also be applied to spectrograms.

Our research scope is on applications that utilize triaxial IMU data, reflecting the common practice of measuring motion across the x, y, and z axes. To harness data in all axes for no information loss, we map the x, y, and z axes to the RGB channels to generate a single image, which was shown to be effective in IMU sensing tasks [16], [18]. This method ensures that the intensity of motion, measured as the root mean square of the triaxial values, is reflected in the brightness, derived from the aggregation of RGB values. It also differentiates each axis's contribution through the prevalence of RGB hues.

We acknowledge several issues in the mapping strategy. For instance, an effective augmentation method for sensor data is *rotation*, i.e., switching the x, y, and z axes, which is the same as changing the image's RGB color tones (i.e., Hue). However, these RBG color tone changes would not be an ideal image augmentation method; for example, replacing the blue sky with a green sky does not make sense. This indicates that following the standard augmentation rules in the vision domain might fail to transfer knowledge to IMU sensing tasks effectively. To handle this mismatch between sensor and image data, we propose a *sensor-aware augmentation* strategy that effectively accounts for such variations, detailed in the subsequent sections.

Fig. 4 illustrates the generation process of 3-channel spectrograms. Spectrograms are created for each axis and mapped to the corresponding RGB color channels. To standardize inputs, we resize all spectrograms to match the image size used for pre-training. This resizing preserves the integrity of spectrogram characteristics, provided the time window and frequency range are maintained. Next, we normalize the spectrograms using

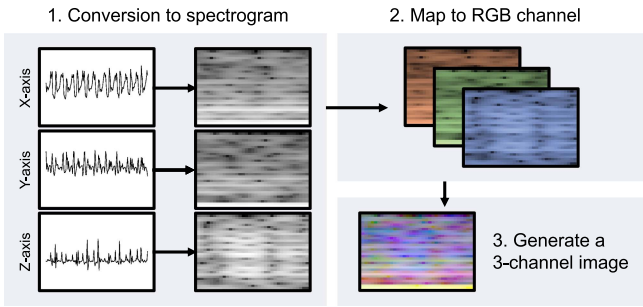


Fig. 4. Generation of a 3-channel image from triaxial IMU sensing data.

the mean and standard deviation of the pre-training images to maintain consistency between the input distributions. Key spectrogram parameters, such as the number of points in the Fast Fourier Transform (*nfft*), are treated as hyperparameters and tuned to optimize performance.

B. Sensor-Aware Pre-Training Using Image Dataset

1) Contrastive Self-Supervised Learning: To address the unique challenges presented by the distinct characteristics of spectrograms compared with conventional images (Fig. 2), IMG2IMU employs contrastive learning [26], [27] for pre-training. We use contrastive learning for its exceptional performance in training vast unlabeled data [11]. More importantly, it has the capability to *selectively train knowledge that is valuable for IMU sensing* while avoiding incompatible information from public image datasets.

Contrastive learning generates a pair of augmented views from a single source, ensuring that these views retain essential mutual information about their inherent characteristics. The goal

during training is to enhance the model's ability to identify and align these augmented pairs while distinguishing them from unrelated examples. The model is trained to capture the intrinsic features maintained across augmentations. We focus on the strategic use of augmentations in contrastive learning; by selecting appropriate augmentations, we can direct the model to learn particular feature insights. For example, scaling augmentation teaches the model to recognize an object with different sizes as similar entities. In contrast, color augmentation trains it to understand that objects are similar with varying colors. In IMG2IMU, we define tailored augmentations designed for IMU sensing tasks, empowering IMG2IMU to acquire useful knowledge, detailed in Section IV-B2.

IMG2IMU implements contrastive learning based on MoCo [26] as it uses a much smaller batch size while achieving comparable performance compared with other baselines such as SimCLR [27]. This efficiency allows operating in resource-constrained environments, resulting in greater scalability. MoCo maintains two encoders; the query encoder and the key encoder. The query encoder generates an embedding named q from a data sample. It generates embedding named positive key, k_+, from the positive pair of the sample, and negative keys $k_i (i = 0, 1, 2, \ldots, K)$ that are encoded from the other data points. The training objective is to make the query q distinguish the positive key (k_+) from the other negative keys (k_i). The query encoder is trained with InfoNCE loss [53] during learning. We calculate the InfoNCE loss as follows:

$$L_q = -\log \frac{\exp(q \cdot k_+/\tau)}{\sum_{i=0}^{K} \exp\left(q \cdot k_i/\tau\right)}, \tag{1}$$

where τ indicates the temperature parameter for controlling the concentration level. MoCo maintains a large set of negative keys by constructing a dictionary that stores data of multiple encoded keys. A moving average is used to update the key encoder based on the weights trained from the query encoder, which enables the dictionary to be dynamic. After contrastive learning is performed on the training image data, the parameters of the query encoder network are used as pre-trained weights for the downstream IMU sensing task.

2) IMU Sensor-Aware Augmentations: Data augmentation preserves the key property of data and generates a different view of the same data. For instance, images are often rotated, flipped, and scaled to change their viewpoint while maintaining color and relative shapes. Using augmentations in contrastive learning, the model learns what mutual information to use to cognize the original and augmented data as the same. Augmentation types should be carefully selected based on what knowledge the model aims to acquire. The usefulness of different augmentations varies in different downstream tasks.

Our downstream tasks take spectrograms derived from triaxial IMU sensing data as the input. Compared with the images from public datasets used for pre-training, spectrograms show unique properties. Spectrograms have *directional properties along the axes*; thus, augmentations such as flipping images would damage the downstream performance as they reverse the time or frequency values. Similarly, rotating images would distort nature as *each axis has fixed values of time and frequency*. Further, the RGB channels in our spectrograms indicate the triaxial axes of x, y, and z, thus we must be aware of *the difference in the channel information*. These are the important domain gap between public image datasets and sensor data, and we thoughtfully select the augmentations for IMG2IMU to bridge this gap.

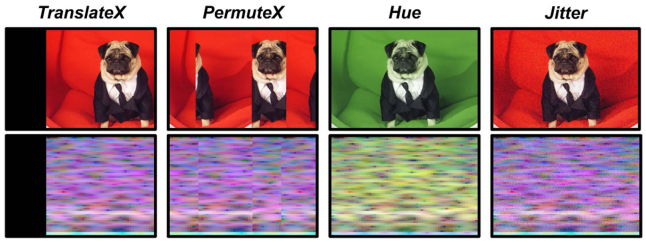


Fig. 5. Sensor-aware augmentations in IMG2IMU.

We identify the important properties of sensor data that must be preserved and define augmentations to assist the model in learning useful knowledge for downstream IMU sensing tasks. Fig. 5 visualizes the selected image augmentations.

- ***TranslateX*** randomly shifts image data on the x-axis. Sensor data are segmented into fixed-size time windows for processing. During this stage, the window can be started at any temporal point from the same context. As the key features of data are within the time window, the classification remains the same *regardless of whether a window is shifted left or right over the time axis*. Based on this property, we expect that *TranslateX* benefits sensing tasks as the x-axis represents time in the spectrogram.
- ***PermuteX*** splits data over the x-axis into multiple chunks and randomly perturbs the chunks. For time-series data, permutation is known to *preserve local temporal features while distorting the global structure* to produce a different view for the same label [54]. We apply *PermuteX* exclusively to the x-axis to introduce variability in the temporal dimension while preserving the frequency-domain features along the y-axis. By keeping the y-axis unchanged, *PermuteX* maintains the frequency-specific patterns.
- ***Hue*** alters the color tone of image data while preserving the overall brightness and contrast. The values between RGB channels are often interchanged with *Hue*. In IMU sensing, *x, y, and z channels are interchangeable based on the rotation* of the sensor. Reflecting the property, rotation is commonly used as an augmentation for triaxial sensors [54]. Our approach maps the sensor data's x, y, and z channels to the RGB channel of an image. By applying *Hue*, we replicate the effect of interchangeability between the three channels in the triaxial IMU sensing data.
- ***Jitter*** adjusts the color by adding random noise for each pixel in the image. We implemented the augmentation by injecting uniform noise centered on zero to preserve the average color information of the image. *Jitter* mimics the augmentation method of *adding random noise to sensor data*. Sensors can be affected by random noise, which in turn can affect the spectrogram by making some regions brighter or darker. We adopt *Jitter* to make the model robust

TABLE I
THE IMPACT OF EACH SENSOR-AWARE IMAGE AUGMENTATION ON IMPROVING ROBUSTNESS AGAINST SENSORY PERTURBATIONS APPLIED TO THE WISDM DATASET [1]

Augmentations				original	time-shifted		masked		rotated		noised	
T	*P*	*H*	*J*	F1	F1	drop	F1	drop	F1	drop	F1	drop
✓	✓	✓	✓	0.754	0.545	−27.75%	0.534	−29.16%	0.695	−7.83%	0.580	−23.10%
✗	✓	✓	✓	0.686	0.434	**−36.78%**	0.468	−31.83%	0.684	−0.34%	0.627	−8.68%
✓	✗	✓	✓	0.687	0.435	**−36.66%**	0.387	**−43.76%**	0.661	−3.90%	0.533	−22.48%
✓	✓	✗	✓	0.704	0.559	−20.68%	0.548	−22.24%	0.539	**−23.45%**	0.622	−11.59%
✓	✓	✓	✗	0.749	0.540	−27.87%	0.502	−33.02%	0.695	−7.19%	0.562	**−24.91%**

T, *P*, *H*, and *J* denotes TranslateX, PermuteX, Hue, and Jitter respectively. We report the drop of the F1-score in each sensory augmentation compared with the original data. The largest drop shown in F1-score (±1%) is in bold.

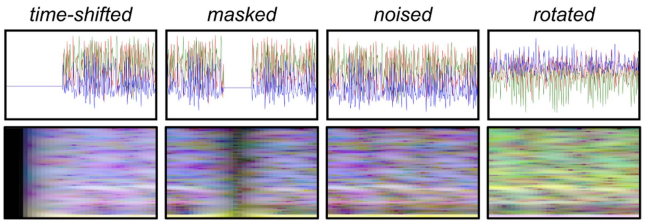


Fig. 6. Four versions of synthetic data from the WISDM [1] dataset to replicate sensor data augmentations: (i) time-shifted, (ii) masked, (iii) rotated, and (iv) noised. Both the sensor data and the resulting spectrograms are shown.

to the noise that could be included in sensor data from uncontrolled environments.

3) Effect of Sensor-Aware Augmentations: We propose four sensor-aware image augmentations: *TranslateX*, *PermuteX*, *Hue*, and *Jitter*. To assess their impact on improving the interpretation of visualized IMU data when used in pre-training with images, we conducted an ablation study.

First, we prepared five pre-trained vision models. The baseline model utilized all four sensor-aware *image augmentations* to generate positive views for contrastive learning. Additionally, we created four ablation settings, each omitting one specific augmentation while using the remaining three. Thus, we examine the impact of excluding a particular augmentation on model performance. All models were pre-trained on the ImageNet dataset [8] under identical conditions.

To assess robustness, we curated four test datasets with distinct variations, reflecting natural sensor data variability. Using the WISDM dataset [1], a widely used benchmark for human activity recognition, we generated synthetic datasets by applying *sensory augmentations* [54]: (i) *time-shifted* data, created by shifting sensor readings left or right; (ii) *masked* data, simulating sensor disconnections by distorting global structures while preserving local temporal features; (iii) *rotated* data, generated through linear transformations that interchange axis values, and (iv) *noised* data, augmented with uniform random noise. Fig. 6 illustrates these augmented datasets and their resulting spectrograms.

Finally, we compared the performance of the models across these datasets. If excluding a particular *image augmentation* leads to a significant performance drop on a dataset with a corresponding *sensory augmentation*, this suggests that the excluded *image augmentation* plays a crucial role in improving the model's robustness to the *sensory augmentation*.

Table I presents the results for each pre-trained model across different augmented sensor datasets. We evaluated the F1-score for each model applied to each dataset. Note the performance drops when different sensory augmentations are applied compared to the original dataset. The pre-trained model with our four sensor-aware augmentations performed the best for all datasets. When we excluded each augmentation, performance dropped to different degrees.

We examined how the absence of each sensor-aware image augmentation influenced robustness. On time-shifted data, the models trained without *TranslateX* and *PermuteX* showed the largest drops, indicating that these augmentations help preserve local temporal structures. Similarly, removing *PermuteX* significantly impacted the model's performance on masked data, which is designed to distort the global features. It verifies that *PermuteX* enhances local feature extraction. For rotated data, eliminating *Hue* weakened robustness to rotation, resulting in a substantial performance drop. Lastly, on noisy data, the pre-trained model without *Jitter* performed the worst, confirming its role in mitigating noise.

While using the four augmentations generally improves performance, applying all of them is not always optimal. In synthetic datasets except *rotated*, eliminating *Hue* resulted in better performance than utilizing all. This occurs because our framework randomly selects an augmentation, and adding more augmentations reduces the likelihood of applying those that are more critical for a given dataset. The effectiveness of an augmentation depends on the key features that must be preserved or learned. Therefore, while IMG2IMU provides a general pool of four augmentations, selecting relevant ones or incorporating additional augmentations when the dataset has distinct characteristics (e.g., extensive missing data or frequent rotations) is critical.

To summarize, we validated that *TranslateX*, *PermuteX*, *Hue*, and *Jitter* serve as sensor-aware augmentations that align with general sensory properties [54]. More importantly, our findings highlight that augmentation selection should be dataset-specific. By understanding the correlation between augmentation types and sensor variations, developers can fine-tune augmentation strategies to maximize performance.

C. Fine-Tuning to IMU Sensing Tasks

Reflecting the scarcity of sensor data, our problem setting assumes only a few samples are available for fine-tuning. We follow a typical fine-tuning setup; the model trained on the public image dataset is fine-tuned on a small subset of data from each

downstream sensing task. As shown in Fig. 4, the data from downstream tasks, which are from IMU sensing applications, are represented as spectrograms. We adopt a popular linear evaluation protocol, freezing the backbone networks and training a fully connected layer as the linear classifier at the end of the backbone network.

V. Evaluation

A. Experimental Setup

1) Datasets: IMG2IMU utilizes image datasets to pre-train representations for downstream sensing tasks. We employed ImageNet [8], a widely known image dataset with 1.28 M samples. For comparison, we used the Capture-24 [13] dataset for pre-training for sensor-based baselines. Capture-24 comprises accelerometer data collected from wrist-worn devices of 151 participants. It comprehensively tracks daily activities, encompassing 4,000 hours of data sampled at 100 Hz.

We assessed the effectiveness of the pre-trained models through their application to four different IMU sensing tasks, all utilizing triaxial accelerometer data for classification. To thoroughly investigate the generalizability, we chose datasets based on the diversity of subjects, sensor position, and tasks.

WISDM [1] covers human activity recognition tasks. Six activities of sitting, standing, walking, jogging, walking downstairs, and walking upstairs were performed by 36 participants. Participants carried smartphones in their pockets during the experiment, where accelerometer data was collected.

Goat Movement [3] contains activity recognition for goats on farms. Data was collected by six accelerometers attached to the collar-shaped device worn by five goats. Activities include stationary, walking, eating, running, and trotting. We omitted eating as it did not have enough samples.

PVS [2] is designed for roadway classification. Accelerometers were placed on the vehicles, and the data was measured from three drivers driving three different types of cars. We use the label information indicating the type of roadway for our main classification task: asphalt, dirt, and cobblestone.

Daphnet [4] is used to detect the freeze of gait for Parkinson's disease patients. A wearable was attached to the ten users (ankle, leg, and waist), and the acceleration was measured. We use the data measured from the ankle to differentiate the positional property from WISDM.

2) Data Preprocessing: All of the images were resized to 128×96 pixels. The images were then normalized using ImageNet statistics. The Capture-24 dataset was downsampled to 50Hz. Given the variety of downstream tasks, data from Capture-24 was segmented into windows of 2, 5, and 10 seconds, each with a 50% overlap. Separate models were pre-trained for each window size, and corresponding models were utilized for downstream tasks requiring different window sizes. The Capture-24 data was normalized using its statistics.

All downstream sensing data were resampled to 50 Hz. Data was windowed into 2, 5, or 10 seconds, using sliding windows with a 50% overlap. The chosen window size matches the description in the respective dataset's original publication [1], [2], [3], [4]. All sensory data were normalized based on the statistics of the pre-training source dataset, following a prior work [12].

Spectrograms were generated from the sensory data. Spectrogram generation parameters *nfft* and *noverlap*, were treated as hyperparameters. A grid search was conducted to determine the optimal hyperparameters, with *nfft* values $\{32, 64, 128, 256\}$ and *noverlap* set at *nfft* minus 2, 4, 8, and 16 for each *nfft* value. As described in Section IV-A, each spectrogram was concatenated into a single RGB image. These images were resized to $128\times$ 96pixels and normalized using the ImageNet data statistics.

Each dataset was randomly divided into training, validation, and testing sets in a 6:2:2 ratio. The splits were based on distinct subjects, ensuring that data from the same subject did not appear in multiple splits. For fine-tuning, we selected very few samples per class (e.g., 10) to simulate data scarcity, randomly sampling from the training split. We repeated the experiments using five different random seeds, creating five independent train-validation-test splits.

3) Baselines: We compared IMG2IMU against nine baselines: four models taking raw (1D) sensory data as input (i.e., sensor-based) and five models utilizing 2D-transformed spectrograms (i.e., image-based).

For the sensor-based baselines, we selected self-supervised learning methods designed for human activity recognition (HAR) [12], [33]. They were pre-trained on the Capture-24 dataset [13], and the pre-trained weights were used for the downstream tasks that use waveform data as input. The following are the sensor-based baselines.

Randomly-initialized (1D) model serves as a baseline for testing weights on 1D waveform data without pre-training.

LIMU-BERT [33] applies BERT-like masked reconstruction, designed for HAR using 1D sensor data.

SimCLR (HAR) [29] applies contrastive learning, redesigned for HAR with 1D inputs and sensory augmentations. Unlike IMG2IMU, it applies sensory augmentations to the raw sensor data.

Multi-task learning (HAR) [23] is a prevalent self-supervised learning technique tailored to HAR. It applies different sensory augmentations to create unique prediction tasks, all processed through a single encoder. By training mutual information between tasks, the encoder learns generalizable representation.

Contrastive Predictive Coding (CPC) (HAR) [31] is a self-supervised learning method that trains models to forecast embeddings by aggregating past embeddings. This enables the model to capture the temporal dynamics and adapt to sensory tasks. We used the latest version, designed for HAR, achieving the state-of-the-art benchmark performance.

For the image-based baselines, we compared models pre-trained on the ImageNet [8] dataset, each utilizing unique pre-training strategies. These were used for downstream tasks with 2D-transformed spectrograms as input.

Randomly-initialized (2D) model serves as a baseline for testing weights without pre-training, where only the spectrograms of the downstream tasks are used for fine-tuning.

ImageNet-supervised model is pre-trained on ImageNet using supervised learning and its labels, with the weights transferred for downstream tasks using spectrograms.

SimSiam [34] represents contrastive learning that bypasses the need for negative samples with stop-gradient. It showcases the application of different approaches in contrastive learning. We used the augmentations provided by the authors.

MoCo [26] is used as a baseline in contrast to the model using sensor-aware augmentations. This incorporates the default augmentations provided in MoCo v2: crop and resize, jittering, horizontal flipping, and Gaussian blurring.

MoCo + All augmentations (2D) [55] uses a wider set of image augmentations: rotating, sharpening, shearing, adjusting contrast, brightness, and color, inverting RGB values, polarizing, posterizing, equalizing, and applying automatic contrast. They were applied in the MoCo-based pre-training.

As upper bounds, we also set **Fully-supervised (1D and 2D)** models by training both sensor- and image-based models using each dataset's fully available training data.

4) Training Configurations: We used ResNet18 [56] backbone and Adam optimizer. IMG2IMU was implemented upon MoCo [26] by replacing the augmentations to *TranslateX*, *PermuteX*, *Hue*, and *Jitter*, without cascading.

Pre-training was conducted over 40 epochs, using a learning rate of $1e^{-6}$ and a batch size of 256. We used a reduced MoCo feature dimension of 64 and a queue size of 4,096 to decrease the computational load. The learning rate started from $1e^{-8}$ and increased up to $1e^{-5}$ for the initial 10 epochs and dropped to $1e^{-6}$ by the last epoch. During fine-tuning, we loaded the pre-trained weights and replaced the last layer of ResNet18 with a randomly initialized layer. We leveraged a linear evaluation protocol, aiming to assess the effectiveness of the pre-trained weights as a feature extractor. Fine-tuning involved only a few samples (e.g., 10) from each class and was conducted over 50 epochs. A batch size of 4 was used for fine-tuning. We conducted a grid search for optimal spectrogram generation parameters for each downstream task (described in Section V-A2).

For sensor-based baselines, except for LIMU-BERT, we implemented 1D CNNs followed by a fully connected layer, strictly replicating the network architecture from the prior assessment [12]. For LIMU-BERT, we adopted the transformer-based structure and training settings from the original paper. For CPC, we replicated the updated version [31], known for its enhanced performance. All models were pre-trained on the Capture-24 [13] dataset for 50 epochs. All image-based baselines were built upon ResNet18. We maintained the pre-training configuration of IMG2IMU for MoCo-based baselines. With SimSiam, we strictly followed the settings in its official implementation [34]. Pre-training hyperparameters were optimized via grid search: learning rates from $1e^{-1}$, $1e^{-2}$, $1e^{-3}$, $1e^{-4}$, $1e^{-5}$, batch sizes from 64, 128, 256 (and 1024, 2048, 4096 for SimCLR, which requires larger batches), and weight decays from 0, $1e^{-3}$, $1e^{-4}$. The fine-tuning for all image-based baselines was conducted in the same setting as IMG2IMU. Fine-tuning mirrored the IMG2IMU protocol, training the only last layer for 50 epochs and maintaining a consistent batch size of 4. Fine-tuning hyperparameters were optimized, exploring the same range of values as for pre-training hyperparameters.

Experiments were repeated using five different random seeds for robustness. All implementations were conducted using PyTorch and eight NVIDIA TITAN Xp GPUs.

5) Metric: The evaluation datasets contain extreme class imbalances. We use macro-averaged F1-score which is robust under class imbalance.

B. Performance Analysis

1) Overall Results: We conducted experiments to investigate the performance of IMG2IMU against the baselines when only a few labeled data were available. For all pre-trained models, we used 10 samples per class for fine-tuning. We examined the performance of the fine-tuned models on the test data of the same downstream task.

Table II shows the result, where IMG2IMU demonstrates superior performance over all baselines. When compared to sensor-based baselines, IMG2IMU achieves a significant improvement, surpassing the highest F1-score by 9.8%p. This performance of IMG2IMU is not simply attributed to the adoption of 2D-transformed inputs, as evidenced by the poor average F1-score (0.407) of randomly initialized models with 2D inputs compared with the F1-score of those with 1D sensory inputs (0.456). This highlights the efficacy of IMG2IMU's pre-training, which yielded a substantial F1-score increase from 0.407 to 0.675. This is a marked contrast to the modest gain of the sensor-based pre-training, which increased at most from 0.456 to 0.579. This result indicates that pre-training using Capture-24 is limited in being applied across downstream tasks involving heterogeneous sensor positions, subjects, or task types. In contrast, IMG2IMU shows that pre-training with the ImageNet dataset—despite its lack of spectrogram images—enables the model to interpret visual features within spectrograms, illustrating better applicability of IMG2IMU in various sensory tasks.

Comparison with image-based baselines shows the effectiveness of IMG2IMU pre-training, as they all use the same ImageNet dataset. IMG2IMU surpasses ImageNet-supervised and SimSiam by a margin greater than 7%p. Comparison with two MoCo-based baselines underscores the impact of augmentations. Despite the default MoCo augmentations achieving the highest performance for typical vision benchmarks, our findings indicate that our sensor-aware augmentations are more appropriate for IMU sensing tasks (0.640→0.675). Furthermore, comparison with MoCo + All augmentations [55] (0.584→0.675) suggests that merely increasing the augmentations does not guarantee enhanced performance.

Additional experiments were conducted by varying the number of training samples ({1, 2, 5, 10, 20, 50}) per class. Fig. 7 shows that generally IMG2IMU performs better than the baselines, especially when training data is limited. Note that we do not limit the potential of IMG2IMU to be trained solely with ImageNet. We anticipate using larger datasets such as LAION-5B would result in greater benefits.

2) Visualizing Semantic Class-Discriminative Heatmaps: To evaluate whether IMG2IMU effectively captures sensory

TABLE II
F1-SCORES OF IMG2IMU AND THE FINE-TUNED BASELINES USING 10 SAMPLES PER CLASS

	Pre-Training Method	WISDM	Goat Movement	PVS	Daphnet	Average
Sensor-based methods (Pre-trained on Capture-24 [13])	Fully-supervised (1D)	0.738 ± 0.100	0.864 ± 0.020	0.722 ± 0.044	0.602 ± 0.041	0.731 ± 0.110
	Randomly-init. (1D)	0.550 ± 0.141	0.270 ± 0.123	0.585 ± 0.065	0.420 ± 0.058	0.456 ± 0.184
	LIMU-BERT [33]	0.516 ± 0.141	0.450 ± 0.123	0.526 ± 0.065	0.408 ± 0.058	0.475 ± 0.123
	SimCLR (HAR) [29]	0.645 ± 0.050	0.585 ± 0.061	0.560 ± 0.113	0.438 ± 0.053	0.557 ± 0.124
	Multi-task (HAR) [23]	0.550 ± 0.170	0.662 ± 0.029	0.583 ± 0.051	0.520 ± 0.073	0.579 ± 0.126
	CPC (HAR) [31]	0.552 ± 0.151	0.650 ± 0.112	0.578 ± 0.084	0.517 ± 0.083	0.574 ± 0.165
Image-based methods (Pre-trained on ImageNet [8])	Fully-supervised (2D)	0.808 ± 0.097	0.855 ± 0.024	0.716 ± 0.066	0.609 ± 0.067	0.747 ± 0.116
	Randomly-init. (2D)	0.374 ± 0.105	0.314 ± 0.055	0.483 ± 0.118	0.456 ± 0.090	0.407 ± 0.115
	ImageNet-supervised	0.620 ± 0.043	0.756 ± 0.051	0.535 ± 0.069	0.499 ± 0.101	0.603 ± 0.111
	SimSiam [34]	0.613 ± 0.099	0.798 ± 0.093	0.518 ± 0.045	0.465 ± 0.058	0.598 ± 0.143
	MoCo [26]	0.689 ± 0.023	0.801 ± 0.057	0.569 ± 0.062	0.502 ± 0.097	0.640 ± 0.119
	MoCo + All aug. [55]	0.627 ± 0.035	0.756 ± 0.061	0.470 ± 0.071	0.484 ± 0.093	0.584 ± 0.123
	IMG2IMU (ours)	**0.739** ± 0.038	**0.821** ± 0.024	**0.594** ± 0.053	**0.547** ± 0.085	**0.675** ± 0.114

Sensor-based baselines were pre-trained on capture-24 [13], while image-based baselines were pre-trained on imagenet [8]. Encoders were frozen during fine-tuning, with only the last layer trained. Highest F1-scores are in bold fonts except for the fully-supervised baselines.

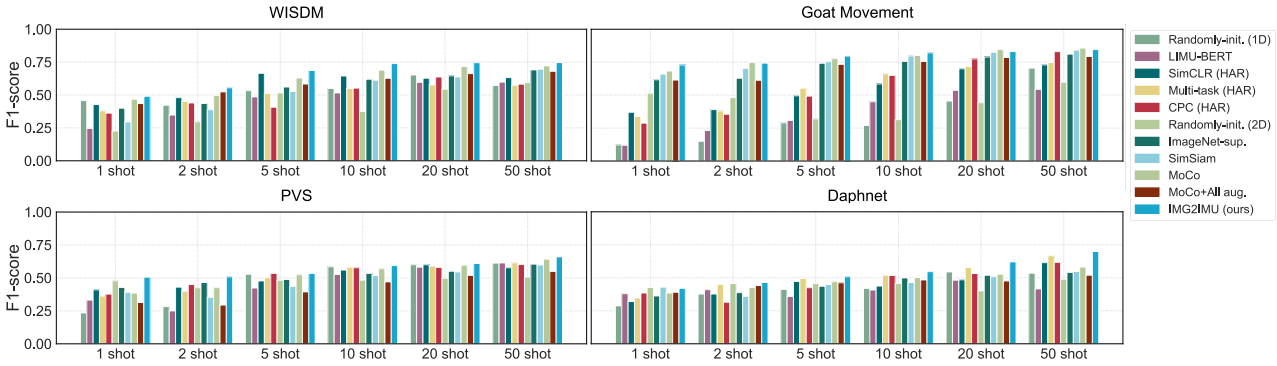


Fig. 7. Performance of the baselines and IMG2IMU using n training samples where the number of training samples is $n \in \{1, 2, 5, 10, 20, 50\}$.

information, we examined the similarity between representations learned from images and those learned under full supervision with sufficient sensor data. We used Grad-CAM [57] to visualize feature interpretations of the pre-trained models. By tracking gradient flows in convolutional layers, Grad-CAM generates a class-discriminative localization map that highlights influential regions in images contributing to the target concept prediction. We compared IMG2IMU with the **Fully-supervised (2D)** baseline, which is trained on the full sensory dataset and outperforms other few-shot baselines. Additionally, we set a **Randomly-initialized (2D)** model as a baseline to show the default heatmap from an image-based model without any pre-trained information. We kept the convolutional layers of IMG2IMU frozen to preserve the pre-trained weights.

Fig. 8 depicts the Grad-CAM heatmaps using the WISDM [1] dataset. A random sample was selected from each class. IMG2IMU and Fully-supervised (2D) models highlight similar regions in the spectrograms across all activity classes. Overall, the low-frequency band is emphasized in the spectrograms. Activities with longer durations, such as walking and jogging, exhibit a broad range of highlighted temporal features, while shorter-duration activities, like going upstairs and downstairs, show a narrower range of emphasized features. Although IMG2IMU is trained solely on a public image dataset, the Grad-CAM results suggest that it correctly interprets sensor data, closely aligning with the fully supervised model.

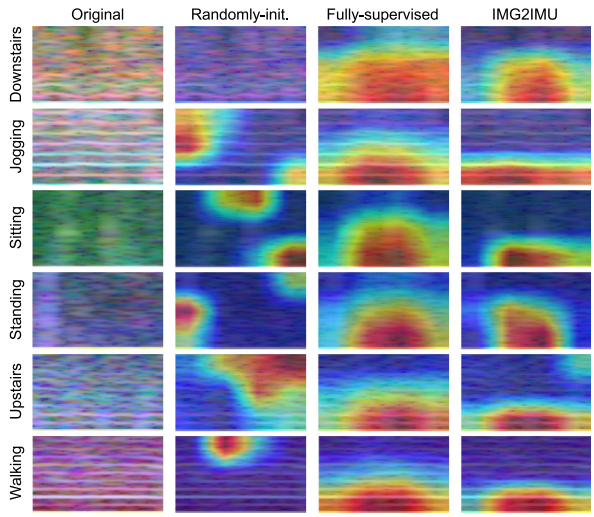


Fig. 8. Grad-CAM comparison on WISDM [1] dataset among Randomly-initialized (2D), Fully-supervised (2D), and IMG2IMU models. The highlighted areas in red indicate the part on which the model focused.

C. Performance on Vision Transformers

To further investigate the impact of scaling the encoder backbone, we conducted experiments using Vision Transformers (ViT) [58]. We examined ViT-S (22 M) and ViT-B (84 M), extending beyond ResNet-18's 11 M parameters. For baselines, we

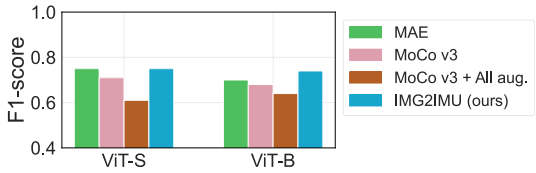


Fig. 9. Performance comparison of IMG2IMU and baseline models using ViT-S and ViT-B as backbone encoders.

TABLE III
ON-DEVICE COMPUTATIONAL OVERHEAD OF IMG2IMU ON THREE COMMODITY SMARTPHONES

	Visualize			Inference		
Device	Time	CPU	Mem	Time	CPU	Mem
Galaxy S20 Ultra	55.33	116	11	26.47	143	189
Galaxy S22 Ultra	48.72	120	11	16.5	105	174
Pixel 2 XL	88.86	108	13	40.67	133	156

Execution time (ms), CPU usage (%), and memory utilization (MB) are measured for spectrogram generation (visualize) and model inference (inference).

adopted MoCo v3 [59], the latest version of MoCo optimized for ViT architectures. We evaluated two MoCo v3 variants: **MoCo v3 (default)** with the augmentation set from the original paper and **MoCo v3 + All augmentations**, which leverages a broader range of augmentations [55]. Additionally, we included **Masked Autoencoder (MAE)** [60], a widely adopted self-supervised learning approach designed for ViT-based models. Finally, we applied IMG2IMU augmentations to MoCo v3 and evaluated all models on the WISDM [1] dataset.

Fig. 9 presents the results. On both ViT-S and ViT-B, IMG2IMU achieved the best performance across all baselines. On ViT-S, MAE exhibited comparable results to IMG2IMU, while on ViT-B, IMG2IMU outperformed MAE. Both ViT variants benefited from IMG2IMU augmentations, consistently improving over the baseline augmentations. These results align with our findings on ResNet-18, demonstrating the effectiveness of IMG2IMU across different architectures.

D. On-Device Computational Overhead

We consider an on-device deployment scenario where we evaluate IMG2IMU's real-time operation capabilities. We assume that pre-training and fine-tuning are completed with a powerful server, after which the model is deployed to a device. Consequently, our focus is on evaluating the overhead associated with on-device inference.

Our framework incurs overhead from the transformation into spectrograms and the use of 2D network architecture. To quantify the overhead, we implemented the IMG2IMU inference framework on smartphones using the PyTorch Android library. We evaluated three commodity smartphones running the fine-tuned IMG2IMU on the WISDM dataset: Galaxy S20 Ultra (8-core CPU, 12 GB RAM), Galaxy S22 Ultra (8-core CPU, 12 GB RAM), and Pixel 2 XL (8-core CPU, 4 GB RAM). Overhead was measured in average execution time (ms), CPU usage (%), and memory utilization (MB) over ten experiments.

Table III presents the computational overhead measured on-device. Overall, the framework's end-to-end computation time was under 0.15 seconds, demonstrating that IMG2IMU incurs negligible overhead for real-time on-device inference. Across all smartphones, spectrogram generation required less than 13 MB of memory and 120% CPU, while 2D inference used up to 189 MB of memory and 143% CPU, confirming its feasibility for on-device deployment on smartphones.

VI. DISCUSSION AND LIMITATIONS

A. Adaptability to Non-Triaxial IMU Setups

IMG2IMU is designed for triaxial IMU data. Our design aligns with the common nature of motion sensors that generate data to represent physical movement in three dimensions [16], [17], [18]. By focusing on triaxial data, we take advantage of its correspondence to the structure of images, where the three axes map to the RGB channels. This alignment allows IMG2IMU to leverage pre-trained vision models optimized for capturing relationships between the color channels. To fully capitalize on the color relationships inherent in images, we introduced the *Hue* augmentation.

However, there are non-triaxial IMU setups, such as single-axis sensors or setups with more than three channels (e.g., combined accelerometer and gyroscope data). In these cases, IMG2IMU can be adapted through the following strategies:

Single-Axis: for setups that provide single axis or aggregated motion, we can leverage IMG2IMU by removing the channel-wise augmentation (*Hue*), while retaining spatial augmentations (*TranslateX*, *PermuteX*, and *Jitter*). These augmentations remain effective for capturing the temporal and positional patterns even in uniaxial data.

Multi-Axis: for setups with more than three axes (e.g., accelerometer-gyroscope), we propose a modality-specific embedding fusion approach. Separate encoders are trained for each triaxial modality (e.g., one for accelerometer and another for gyroscope), enabling the model to capture intra-modality features. The embeddings are then concatenated and passed through shared projection layers, allowing the system to learn inter-modality relationships. This approach enables IMG2IMU to adapt to multi-axis configurations while preserving both intra- and inter-modality features. Note that our millisecond-level computation ensures minimal overhead for the fusion process, making it feasible for multi-modal tasks.

B. Potential for Exploring Sensor-Aware Augmentations

The selection of augmentation types in contrastive learning strongly impacts the performance of downstream tasks. IMG2IMU defines four augmentations that benefit contrastive learning for IMU sensing tasks. This augmentation design was derived from the key invariants in sensing applications, referring to the widely accepted sensor data augmentations [54]. While we also attempted other types of image augmentation, such as *Brightness* and *Contrast*, they did not show clear correlations. Nevertheless, as there are numerous invariants in sensor data, there could be other augmentations useful for sensing applications. More augmentations could be built upon and

potentially further improve the pre-trained model's performance with IMG2IMU.

C. Optimizing 2D Transformation Process of Sensor Data

To apply the knowledge learned from images, we transform the IMU sensor data into spectrograms. While our results show that spectrogram conversion benefits diverse sensing tasks when combined with IMG2IMU, its effectiveness relies on parameters used in its generation. For instance, spectrograms may fail to capture key features if the *nfft* parameter is set inappropriately. We found that using *nfft*=128 on the WISDM dataset achieved the highest 0.739 F1-score, but reducing *nfft* to 64 slightly decreased performance to 0.734, and further reducing it to 32 led to a drop to 0.660. These results indicate that IMG2IMU's performance is sensitive to the visualization, showing the importance of hyperparameter selection.

To mitigate this sensitivity, future work could explore adaptive spectrogram configurations, dynamically performing optimal visualization based on data characteristics. Additionally, previous research has demonstrated that alternative 2D representations [17] can be highly effective for sensor-based classification. These representations could be integrated into IMG2IMU by designing augmentation strategies tailored to their specific properties, further enhancing robustness.

VII. Conclusion

We presented IMG2IMU that utilizes the learned representation from images to IMU sensing tasks. We proposed a new contrastive learning method that employs image augmentations explicitly designed for sensing applications and correlates each augmentation type with sensory properties. Our evaluations demonstrated that IMG2IMU improves performance on a variety of IMU sensing applications when fine-tuned to the learned representations. IMG2IMU showcased how vision knowledge can be effectively translated to IMU sensing tasks and is beneficial for IMU sensing applications that lack large-scale training data.

References

[1] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," *ACM SigKDD Explorations Newslett.*, vol. 12, no. 2, pp. 74–82, 2011.

[2] J. Menegazzo and A. von Wangenheim, "Multi-contextual and multi-aspect analysis for road surface type classification through inertial sensors and deep learning," in *Proc. 2020 X Braz. Symp. Comput. Syst. Eng.*, 2020, pp. 1–8.

[3] J. W. Kamminga, D. V. Le, J. P. Meijers, H. Bisby, N. Meratnia, and P. J. Havinga, "Robust sensor-orientation-independent feature selection for animal activity recognition on collar tags," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 2, no. 1, pp. 1–27, 2018.

[4] M. Bachlin et al., "Wearable assistant for Parkinson's disease patients with the freezing of gait symptom," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 2, pp. 436–446, Mar. 2010.

[5] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[6] A. Dubey et al., "The llama 3 herd of models," 2024, *arXiv:2407.21783*.

[7] J. Achiam et al., "GPT-4 technical report," 2023, *arXiv:2303.08774*.

[8] O. Russakovsky et al., "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[9] X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer, "Scaling vision transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 12104–12113.

[10] C. Schuhmann et al., "Laion-5B: An open large-scale dataset for training next generation image-text models," 2022, *arXiv:2210.08402*.

[11] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. E. Hinton, "Big self-supervised models are strong semi-supervised learners," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 22243–22255.

[12] H. Haresamudram, I. Essa, and T. Plötz, "Assessing the state of self-supervised human activity recognition using wearables," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 6, no. 3, pp. 1–47, 2022.

[13] S. Chan et al., "Capture-24: A large dataset of wrist-worn activity tracker data collected in the wild for human activity recognition," *Sci. Data*, vol. 11, no. 1, 2024, Art. no. 1135.

[14] A. Doherty et al., "Large scale population assessment of physical activity using wrist worn accelerometers: The UK biobank study," *PLoS One*, vol. 12, no. 2, 2017, Art. no. e0169649.

[15] G. Narayanswamy et al., "Scaling wearable foundation models," 2024, *arXiv:2410.13638*.

[16] M. A. Alsheikh, A. Selim, D. Niyato, L. Doyle, S. Lin, and H.-P. Tan, "Deep activity recognition models with triaxial accelerometers," in *Proc. Workshops 30th AAAI Conf. Artif. Intell.*, 2016, pp 8–13.

[17] T. Hur, J. Bang, T. Huynh-The, J. Lee, J.-I. Kim, and S. Lee, "Iss2Image: A novel signal-encoding technique for CNN-based human activity recognition," *Sensors*, vol. 18, no. 11, 2018, Art. no. 3910.

[18] D. Ravi, C. Wong, B. Lo, and G.-Z. Yang, "Deep learning for human activity recognition: A resource efficient implementation on low-power devices," in *Proc. IEEE 13th Int. Conf. Wearable Implantable Body Sensor Netw.*, 2016, pp. 71–76.

[19] D. Keim, "Information visualization and visual data mining," *IEEE Trans. Vis. Comput. Graphics*, vol. 8, no. 1, pp. 1–8, First Quarter, 2002.

[20] G. Brunner, D. Melnyk, B. Sigfússon, and R. Wattenhofer, "Swimming style recognition and lap counting using a smartwatch and deep learning," in *Proc. ACM Int. Symp. Wearable Comput.*, 2019, pp. 23–31.

[21] A. Kolesnikov et al., "Big transfer (bit): General visual representation learning," in *Proc. 16th Eur. Conf. Comput. Vis.*, Glasgow, U.K., 2020, pp. 491–507.

[22] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. 38th Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763. [Online]. Available: https://proceedings.mlr.press/v139/radford21a.html

[23] A. Saeed, T. Ozcelebi, and J. Lukkien, "Multi-task self-supervised learning for human activity detection," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 3, no. 2, pp. 1–30, 2019.

[24] H. Yuan et al., "Self-supervised learning for human activity recognition using 700,000 person-days of wearable data," *NPJ Digit. Med.*, vol. 7, no. 1, 2024, Art. no. 91.

[25] C. I. Tang, I. Perez-Pozuelo, D. Spathis, S. Brage, N. Wareham, and C. Mascolo, "SelfHAR: Improving human activity recognition through self-training with unlabeled data," 2021, *arXiv:2102.06073*.

[26] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9729–9738.

[27] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.

[28] J. Wang, T. Zhu, J. Gan, H. Ning, and Y. Wan, "Sensor data augmentation with resampling for contrastive learning in human activity recognition," 2021, *arXiv:2109.02054*.

[29] C. I. Tang, I. Perez-Pozuelo, D. Spathis, and C. Mascolo, "Exploring contrastive learning in human activity recognition for healthcare," 2020, *arXiv:2011.11542*.

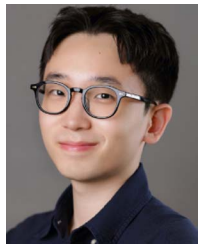
[30] B. Khaertdinov, E. Ghaleb, and S. Asteriadis, "Contrastive self-supervised learning for sensor-based human activity recognition," in *Proc. IEEE Int. Joint Conf. Biometrics*, 2021, pp. 1–8.

[31] H. Haresamudram, I. Essa, and T. Ploetz, "Investigating enhancements to contrastive predictive coding for human activity recognition," 2022, *arXiv:2211.06173*.

[32] H. Haresamudram, I. Essa, and T. Plötz, "Contrastive predictive coding for human activity recognition," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 5, no. 2, pp. 1–26, 2021.

[33] H. Xu, P. Zhou, R. Tan, M. Li, and G. Shen, "LIMU-BERT: Unleashing the potential of unlabeled data for IMU sensing applications," in *Proc. 19th ACM Conf. Embedded Netw. Sensor Syst.*, 2021, pp. 220–233.

[34] X. Chen and K. He, "Exploring simple siamese representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15750–15758.
[35] Y. Jain, C. I. Tang, C. Min, F. Kawsar, and A. Mathur, "ColloSSL: Collaborative self-supervised learning for human activity recognition," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 6, no. 1, pp. 1–28, 2022.
[36] S. Deldari, H. Xue, A. Saeed, D. V. Smith, and F. D. Salim, "COCOA: Cross modality contrastive learning for sensor data," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 6, no. 3, pp. 1–28, 2022.
[37] V. Radu and M. Henne, "Vision2sensor: Knowledge transfer across sensing modalities for human activity recognition," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 3, no. 3, pp. 1–21, 2019.
[38] S. Bhalla, M. Goel, and R. Khurana, "IMU2Dopple: Cross-modal domain adaptation for doppler-based activity recognition using IMU data," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 5, no. 4, pp. 1–20, 2021.
[39] C. Tong, J. Ge, and N. D. Lane, "Zero-shot learning for IMU-based activity recognition using video embeddings," in *Proc. ACM Interactive, Mobile Wearable Ubiquitous Technol.*, vol. 5, no. 4, pp. 1–23, 2021.
[40] H. Kwon et al., "IMUTube: Automatic extraction of virtual on-body accelerometry from video for human activity recognition," in *Proc.ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 4, no. 3, pp. 1–29, 2020.
[41] Z. Leng et al., "IMUGPT 2.0: Language-based cross modality transfer for sensor-based human activity recognition," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 8, no. 3, pp. 1–32, 2024.
[42] K. He, R. Girshick, and P. Dollár, "Rethinking ImageNet pre-training," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 4918–4927.
[43] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 580–587.
[44] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3150–3158.
[45] S. Azizi et al., "Big self-supervised models advance medical image classification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 3478–3488.
[46] Y. Liu et al., "A deep learning system for differential diagnosis of skin diseases," *Nature Med.*, vol. 26, no. 6, pp. 900–908, 2020.
[47] J. Irvin et al., "Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 590–597.
[48] S. Shin, J. Kim, Y. Yu, S. Lee, and K. Lee, "Self-supervised transfer learning from natural images for sound classification," *Appl. Sci.*, vol. 11, no. 7, 2021, Art. no. 3043.
[49] L. Xue et al., "xGen-MM (BLIP-3): A family of open large multimodal models," 2024, *arXiv:2408.08872*.
[50] A. Kirillov et al., "Segment anything," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 4015–4026.
[51] J. Zhang, J. Huang, S. Jin, and S. Lu, "Vision-language models for vision tasks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 8, pp. 5625–5644, Aug. 2024.
[52] H. Yoon, B. A. Tolera, T. Gong, K. Lee, and S.-J. Lee, "By my eyes: Grounding multimodal large language models with sensor data via visual prompting," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2024, pp. 2219–2241.
[53] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv: 1807.03748*.
[54] T. T. Um et al., "Data augmentation of wearable sensor data for parkinson's disease monitoring using convolutional neural networks," in *Proc. 19th ACM Int. Conf. Multimodal Interaction*, 2017, pp. 216–220.
[55] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, pp. 1–48, 2019.
[56] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, 2016, pp. 770–778.
[57] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 618–626.
[58] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy
[59] X. Chen, S. Xie, and K. He, "An empirical study of training self-supervised vision transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 9640–9649.
[60] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 16000–16009.



Hyungjun Yoon received the BS (cum laude) degree in computer science from KAIST. He is currently working toward the PhD degree in electrical engineering with KAIST. His research interests include mobile sensing, and applied machine learning, and foundation models.

Hyeongheon Cha received the BS (magna cum laude) degree in electrical engineering from KAIST. He is currently working toward the PhD degree in electrical engineering with KAIST. His research interests include on-device AI, mobile computing, ubiquitous sensing, and applied machine learning.

Hoang C. Nguyen received the BS degree in computer science and mathematics from KAIST. He is currently working toward the PhD degree in computer science with Stony Brook University. His research interests include computer vision, multi-modal learning, explainable AI, and smart health.

Taesik Gong received the PhD degree in computer science from KAIST, in 2023. He is an assistant professor with the Department of Computer Science and Engineering, UNIST. During his Ph.D., he interned with Google Research, Microsoft Research, and Nokia Bell Labs. Before joining UNIST, he was a research scientist with Nokia Bell Labs and a visiting scholar with the University of Cambridge. He is also a recipient of the Google Ph.D. Fellowship. His research interests include on-device AI, human-centered AI, and ubiquitous computing.

Sung-Ju Lee (Fellow, IEEE) received the PhD degree in computer science from the University of California, Los Angeles, California, in 2000. After spending 15 years in the industry in Silicon Valley, he joined KAIST, where he is a professor and KAIST endowed chair professor. His research interests include mobile computing, mobile ML/AI, wireless networks, and HCI. He won the HP CEO Innovation Award, the Best Paper Awards at ACM CSCW 2021 and IEEE ICDCS 2016, and the Test-of-Time Paper Award at ACM WINTECH 2016.