When Video meets Inertial Sensors: Zero-shot Domain Adaptation for Finger Motion Analytics with Inertial Sensors

Yilin Liu
The Pennsylvania State University
yzl470@psu.edu

Shijia Zhang
The Pennsylvania State University
sbz5188@psu.edu

Mahanth Gowda
The Pennsylvania State University
mahanth.gowda@psu.edu

ABSTRACT

Ubiquitous finger motion tracking enables a number of exciting applications in augmented reality, sports analytics, rehabilitation-healthcare etc. While finger motion tracking with cameras is very mature, largely due to availability of massive training datasets, there is a dearth of training data for developing robust machine learning (ML) models for wearable IoT devices with Inertial Measurement Unit (IMU) sensors. Towards addressing this problem, this paper presents *ZeroNet*, a system that shows the feasibility of developing ML models for IMU sensors with zero training overhead. *ZeroNet* harvests training data from publicly available videos for performing inferences on IMU. The difference in data among video and IMU domains introduces a number of challenges due to differences in sensor-camera coordinate systems, body sizes of users, speed/orientation changes during gesturing, sensor position variations etc. *ZeroNet* addresses these challenges by systematically extracting motion data from videos and transforming them into acceleration and orientation information measured by IMU sensors. Furthermore, data-augmentation techniques are exploited that create synthetic variations in the harvested training data to enhance the generalizability and robustness of the ML models to user diversity. Evaluation with 10 users demonstrates a top-1 accuracy of 82.4% and a top-3 accuracy of 94.8% for recognition of 50 finger gestures thus indicating promise. While we have only scratched the surface, we outline a number of interesting possibilities for extending this work in the cross-disciplinary areas of computer vision, machine learning, and wearable IoT for enabling novel applications in finger motion tracking.

CCS CONCEPTS

• **Human-centered computing** → **Mobile devices**; **Ubiquitous and mobile computing design and evaluation methods**; • **Computing methodologies** → *Neural networks.*

KEYWORDS

IoT; Wearable; IMU; Data argumentation; Finger gesture

ACM Reference Format:
Yilin Liu, Shijia Zhang, and Mahanth Gowda. 2021. When Video meets Inertial Sensors: Zero-shot Domain Adaptation for Finger Motion Analytics with Inertial Sensors. In *International Conference on Internet-of-Things Design and Implementation (IoTDI '21), May 18–21, 2021, Charlottesvle, VA, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3450268.3453537

IoTDI '21, May 18–21, 2021, Charlottesvle, VA, USA

ACM ISBN 978-1-4503-8354-7/21/05…$15.00
https://doi.org/10.1145/3450268.3453537

1 INTRODUCTION

Finger motion tracking enables exciting IoT applications in sports analytics [2], healthcare and rehabilitation [28, 65], sign languages [13], augmented reality (AR), virtual reality (VR), etc. Analysis of finger motion of aspiring players can be compared to experts to provide automated coaching support. In the context of healthcare, finger motion stability patterns are known to be bio-markers for predicting motor neuron diseases [12, 22]. AR/VR gaming as well as precise control of robotic prosthetic devices are some of the other applications that benefit from finger gesture tracking [11, 43].

Motivated by the above applications and coupled by recent innovations in machine learning (ML) and the availability of large scale training data, there is a surge of recent research [18, 31, 42] in computer vision that track accurate finger poses from monocular videos. Given that they do not require depth cameras, the range of applications enabled is far reaching. However, such vision based techniques are affected by issues such as occlusions and the need for good lighting conditions to capture intricate finger motions.

In contrast to vision, the main advantage of wearable IoT devices lies in enabling ubiquitous tracking without external infrastructure while being robust to lighting and occlusions. However, unlike vision, there is a dearth of large scale training data to develop robust ML models for wearable devices. Towards overcoming this gap, this paper presents a system called *ZeroNet*. This system requires zero training overhead for developing robust ML models for finger motion analytics using smart-ring based Inertial Measurement Unit (IMU) sensors. In particular, *ZeroNet* harvests training data from public videos of finger motions and develops ML models that can be used for inferences on smart-rings with IMU sensors.

Such a method of learning from one domain for performing inferences on a different domain has been explored before. Unsupervised domain adaptation [64, 67] can adapt distributions between source (video) and target (IMU) domains such that the model learnt on source domain is used for inference on target domain. However, such techniques are hard to apply to our problem domain since this still requires enough real training data (atleast in unlabelled form) from IMU to achieve sufficient convergence of the domain adaptation process. Furthermore, each user's finger motion pattern as well as natural variations in sensor wearing positions could lead to different distributions in the sensor data [15, 20] thus entailing

more training data for each setting. On the other hand, *ZeroNet* performs comparable to models developed with semi-supervised domain adaptation [27, 83] which need partial labelled real IMU data and even outperforms models fully trained on our own real IMU dataset (details in Sec. 6). Given lack of large training datasets under diverse conditions for smart-rings, we believe *ZeroNet*'s ability to provide promising accuracy without any training cost is an important first step to bootstrap applications. With enough applications, more data can potentially be generated via crowd-sourcing approaches to further push the accuracy of domain adaptation.

Fig. 1 illustrates the architecture of *ZeroNet* with the following sequence of actions. (i) Appropriate sources of publicly available videos (YouTube, ViMeo, Flickr etc.) are first identified as candidates for training data. (ii) Finger locations are then extracted from these videos using computer vision techniques [19, 73]. (iii) Appropriate motion metrics that can be captured from IMU (acceleration, orientation etc) are then derived from these finger locations. (iv) The training data thus extracted from videos is further enlarged using data augmentation techniques (introducing variants of rotations, speed of gestures, temporal clipping etc) to create a large and high quality training dataset. (v) Such synthetic datasets are used for training ML models (vi) Finally, the trained models can be deployed directly for inferences on wearable devices with zero training overhead. Inspired by favorable usability reviews of smart-rings in monitoring activity in gym, sleep etc.,[46–48] we place a sensor on the finger for gesture inferences (details in Sec. 3).

Although in a similar spirit to recent works [38, 55, 66, 74] showing the feasibility of harvesting training data from videos for identifying upto ten classes of human activities, *ZeroNet* differs from the above works in following ways: (i) Shows the feasibility of harvesting training data from videos for a gesture recognition problem involving intricate finger motions. (ii) The harvested training data from videos is combined with data augmentation techniques to enable better generalizability of ML models. (iii) Shows the ability of recognition over 50 classes – a five fold higher number of classes than prior work extracting training data from public videos.

Harvesting training data from videos for performing inferences on IMU is challenging because: (i) The IMU and camera have differences in sensing modalities, coordinate systems etc., thus requiring careful pre-processing to transfer knowledge between the two domains. (ii) The speed/orientation of gesturing, and body sizes can differ across users. Similarly, the sensor wearing position and orientation can vary due to natural errors in sensor placement. (iii) The distribution of training data and test data will not match since they come from different sources. Appropriate techniques are needed to generalize the model developed from video-based training data to perform accurate inferences on wearable devices.

In solving the above challenges, *ZeroNet* exploits a number of opportunities. (i) The sensor and camera coordinates can be appropriately aligned by measuring the orientation of the wearable device to perform coordinate transformations. (ii) *ZeroNet* approximates IMU-like sensor data from location estimates extracted from videos by performing systematic finite differences of locations to derive accelerometer data. Similarly, the angle between finger joints and the vertical plane is extracted from videos to approximate a dimension of the orientation data. (iii) Towards handling body size diversity, the location data from each video is normalized to a measurement corresponding to a uniform body size (for example, by scaling the data by the ratio of the shoulder length of the person in video to a standard shoulder length). (iv) Towards enhancing the robustness and generalizability of ML models, we augment the training data by creating synthetic variants of the data with varying speeds and magnitudes of acceleration. In addition, variants of data with minor shifts in rotations is also added to provide robustness to varying finger orientations or sensor positioning/displacement.

We implement *ZeroNet* on a wearable platform of a button shaped off-the-shelf Mbient Sensor[9] worn as a ring on fingers. We extract training data for 50 gestures of finger motion from a popular public video source of American Sign Language (ASL) tutorial [8]. We develop a CNN based model using this data by exploiting the above enumerated opportunities. Testing results on 10 users achieves a top-1 accuracy of 82.4% and a top-3 accuracy if 94.8% which demonstrates the feasibility of our system. An implementation on Samsung Galaxy S20 smartphones using TensorflowLite validates the low latency and energy efficiency of the system.

A summary of our contributions in the paper include:
(i) Showing the feasibility of harvesting training data from videos for performing inferences on IMU for finger gesture tracking.
(ii) A systematic pipeline that fuses data processing and data augmentation techniques for better generalizability of ML models
(iii) Evaluation with 10 users that shows a top-1 accuracy of 82.4% *and a top-3 accuracy of* 94.8% *over 50 gestures.*

The rest of the paper will expand upon this idea. Sec. 2 provides a background on the nature of data in the domains of videos and IMU, while Sec. 3 introduces the IMU platform . Sec. 4 will design a signal processing pipeline for systematically transforming video-based training data into IMU-based data. Sec. 5 will discuss data augmentation techniques to handle the domain difference between training and test data as well as for creating robust model that is generalizable to any new user. Sec. 6 will provide results from our experiments. Sec. 7 will survey related research and finally we conclude with limitations and future directions in Sec. 8.

2 BACKGROUND

The success of human activity recognition in machine learning depends on the availability of large scale annotated datasets. For example *Human 3.6m* [30] has 3.6 million images of various activities such as eating, walking, discussing, sitting, providing-directions, talking on phone, making purchases etc. Similarly, the popular *ImageNet* database consists of 14 million images. In contrast, for wearables, *Daphnet* [14] gait dataset has 5 hours of walking data from 10 subjects and *PAMP2* dataset[54] has 7.5 hours of sensor data from 9 subjects. Such datasets are very small in comparison to vision datasets. Moreover to the best of our knowledge, such datasets do not exist for finger motion tracking that use the recently

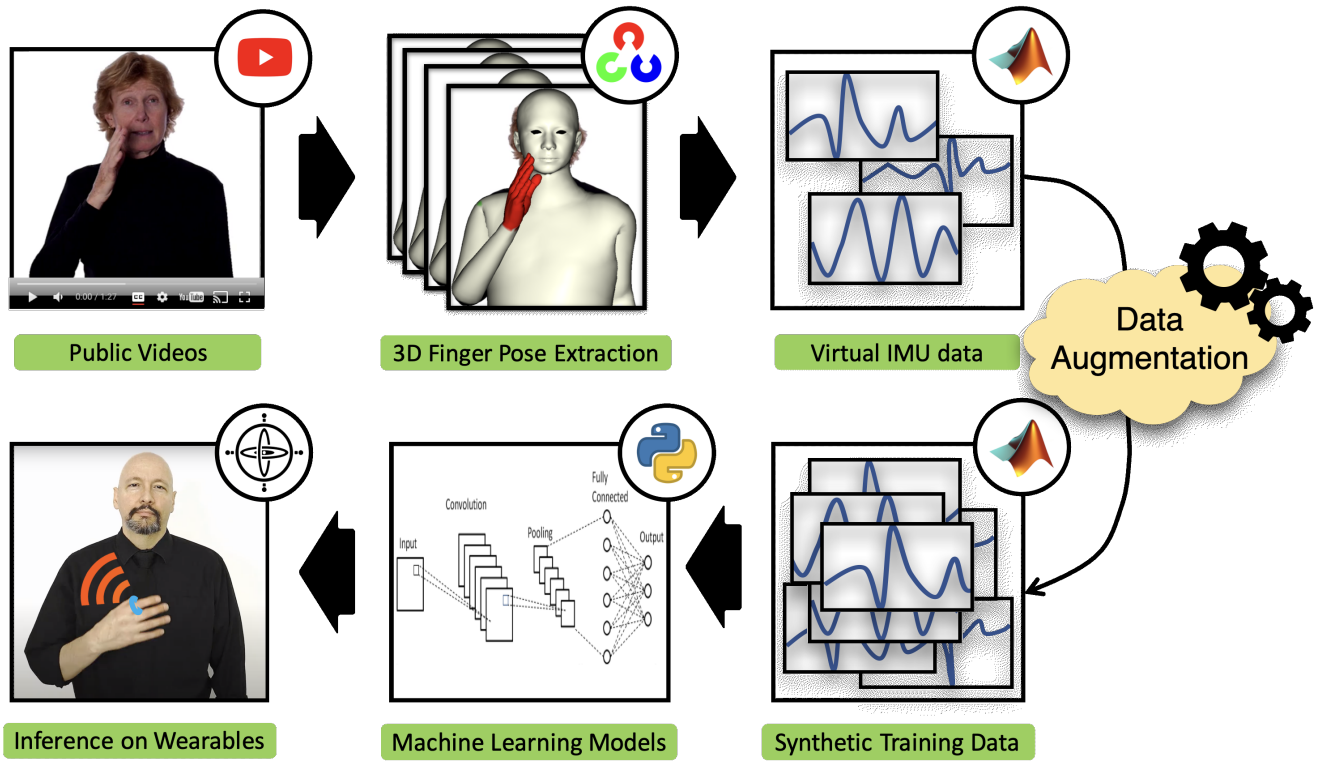

Figure 1: The flow of operations in *ZeroNet*. 3D finger pose and locations are first extracted from videos. The location and pose information is transformed into acceleration and orientation that can be captured by inertial sensors. Data augmentation techniques are then introduced to create robust synthetic training datasets. The ML models developed on such datasets are generalizable and directly used for inferences on wearable devices (smart-ring worn on finger) without any training overhead.

emerging platform of smart-rings. Towards overcoming this gap, this section briefly discusses extracting finger locations from video data for harvesting training data. We also discuss the nature of IMU data to be approximated with video data.

2.1 Video Data

Large amounts of video datasets are publicly available. For example, there are several YouTube videos of sports activities, movie clips of human activities, sign language news etc. Exploiting such datasets for harvesting training data can significantly reduce the overhead of training data generation on wearable devices. In this paper, we harvest training data from a popular public tutorial of sign language gestures [8]. We show the feasibility of recognition of 50 most popular finger gestures without any training data from IMU.

We exploit state-of-the art computer vision techniques for extracting motion data from the videos for training ML models. Fig. 1 shows an example of a frame from our video dataset. We exploit techniques in [73] that can extract finger joint locations from simple RGB images, also shown in Fig. 1. In particular, Xiang et al [73] use an efficient representation called 3D part orientation fields (POF) to encode 3D orientation of all body parts in a 2D image space. The POFs are learnt by a CNN trained over a large dataset thus learning to predict 3D deformable mesh model of the whole body, face, and fingers. While RGB images do not contain depth information, the CNN model exploits the known priors of shape and pose models of human body in addition to applying constraints of temporal smoothness for extracting 3D motion information. As shown in Fig. 1, the whole body shape is extracted from which we only identify the finger locations from the red highlighted region. The extracted finger locations is used by *ZeroNet* to develop ML models for IMU data as elaborated in further sections.

2.2 IMU Sensor Data

Inertial Measurement Units (IMU) consists of accelerometer, gyroscope, and magnetometer sensors widely embedded in wearable IoT devices for enabling a number of applications in gesture recognition, augmented reality, smart health etc. We provide a brief overview of the 3D orientation of an object since it plays a critical role in modeling the data captured by these sensors.

Consider a *global frame* of reference pointed towards "Up", "East", and "North" directions (Fig. 2). Consider an object (e.g., IMU sensor) whose *local frame* of reference is also shown in the figure. While the two frames are perfectly aligned in Fig. 2(a), there is a misalignment between the two frames as shown in Fig. 2(b). The *3D orientation* of an object captures this misalignment between the *local* and *global frames* of reference. Consider a vector V whose representation in

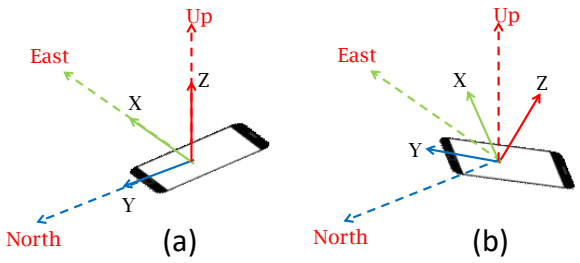

Figure 2: (a) Perfectly aligned *local* and *global frames* (b) Misalignment between *local* and *global frames*. Orientation captures the misalignment between *local* and *global frames*

Figure 3: Button sized IMU worn as a ring

the *local* and *global frames* are $V_l = [X_l\ Y_l\ Z_l]$, and $V_g = [X_g\ Y_g\ Z_g]$ respectively. The 3D orientation of the object can be mathematically quantified using a 3×3 rotation matrix R which rotates the vector between the two frames of reference as indicated below.

$$\begin{bmatrix} X_l & Y_l & Z_l \end{bmatrix} R = \begin{bmatrix} X_g & Y_g & Z_g \end{bmatrix}$$

When an accelerometer sensor is under rest, it measures the projection of the gravity vector on its three axes [59]. Similarly, the magnetometer sensor measures the projection of the earth's magnetic field on its three axes. Since the acceleration due to gravity and the geomagnetic field are globally known vectors, the local measurements of these values using the sensors can ideally be used for computing the rotation matrix R described above to quantify the orientation of an object. However, in reality, the mobility of the sensor can corrupt the measurements of gravity by the accelerometer, as well as the electromagnetic interference can interfere with the magnetometer. Therefore, the gyroscope sensor data which measures the change in orientation (angular velocity) can be fused with estimates of orientation from accelerometer and magnetometer to compute accurate 3D orientation estimates of an object [82].

An accelerometer sensor measures the superposition of the gravity and acceleration due to the linear motion of the wearable device. The measurement is relative to the sensor's *local frame* of reference. Therefore, the orientation estimates as discussed above is useful not only in converting the accelerometer measurements to the *global frame* of reference but also in subtracting the component of gravity from the acceleration measurements.

3 PLATFORM DESCRIPTION

We begin with a simple platform with a ring-like sensor worn on the index finger as shown in Fig. 3. Note that all fingers are involved in gesturing, but we place the sensor only on the index finger. While we believe this is sufficient to show the feasibility of harvesting training data from videos, this will cause miss-classifications among gestures with similar motion of the index finger and different motion of other fingers. However, surprisingly, the accuracy with just index-finger data is significant with very few miss-classifications due to the specific reason noted here (details in Sec. 6). While the miss-classification rate might increase with number of classes, we discuss opportunities with additional techniques and sensors in Sec. 8. The majority of the study places the sensor on the index finger since it is more frequently involved in gestures in our video dataset. However, we also conduct experiments to understand the best placement option among other fingers (Sec. 6).

Smart rings that can pair with phones wirelessly to stream information as well a monitor activity are already available on the market [5, 6]. For example, the Oura ring [6] is popular as a sleep tracking device and weighs between 4 and 6 grams, which is even lighter than conventional rings, and packaged in a stylish design. It is low intrusive with users finding it comfortable for wearing day and night, gym, pool etc [46], thus receiving favorable online reviews for usability [46–48]. However, most of these platforms are closed and do not provide access to raw sensor data. Therefore we use a button-shaped sensor from MbientLabs [9] snugly fit on the finger like a ring as shown in Fig. 3. The sensor streams data wirelessly to a smartphone which runs the ML models for gesture recognition. The ring generates 9 axis IMU data - 3 axes each for Accelerometer, Magnetometer, and Gyroscope. This forms the input to *ZeroNet*.

4 SYNTHETIC TRAINING DATA FROM VIDEOS

The 3D locations captured from the video will be transformed into synthetic accelerometer and orientation data for training the ML models. A natural first step would be to simply double differentiate the index finger location as extracted from the video to obtain the acceleration of the index finger. However, such a simple differentiation will not emulate the accelerometer data because of a number of differences between IMU and video data. In this section, we elaborate these differences together with approaches in *ZeroNet* to address these differences. We begin by discussing the basic pre-processing steps.

4.1 Pre-processing

A number of simple but critical pre-processing steps are needed to match the distribution of the video and IMU dataset. We enumerate the main steps here: (i) A low pass filtering with a cutoff frequency of 10Hz was applied on both video derived acceleration and IMU acceleration. (ii) The orientation data extracted from videos posses a characteristic shape mainly because of the noise in the camera data. Simply using these orientation estimates made the CNN model

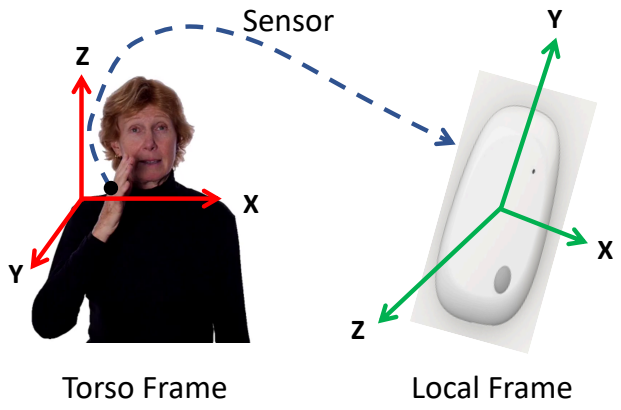

Figure 4: The camera's motion data in *Torso Coordinate Frame* can be aligned with the sensor measure data relative to *Local Frame* using orientation estimates of sensor

memorize the shape and overfit. Thus, we regularized the orientation data using a smooth, low parametric function so as to prevent the CNN model from memorizing the noise in the data.

4.2 Extraction of Acceleration

Coordinate differences: The location data captured by cameras is relative to the camera's frame of reference. However, the locations can be transformed into torso coordinate frame (TCF) as shown in Fig. 4. We chose our x-axis as the line joining the two ends of the shoulder when the user is in a *stable pose*. Similarly, we chose the z-axis to be in the plane of the torso but perpendicular to x-axis. The y-axis is perpendicular to these two axes. Since we extract entire shape of the human body using the work in [73], we identify the appropriate shoulder and torso joints corresponding to the TCF. We then project the extracted locations from the camera into TCF.

On the other hand, the acceleration measured by the sensors will be in the *local frame* of reference which depends on the instantaneous orientation of the sensor as depicted in Fig. 4. Therefore, *ZeroNet* first converts the acceleration into the *global frame* of reference. The difference between the *global frame* and the user's facing direction can be roughly computed when the sensor is in vertical free-fall position or if the user is walking a few steps [56, 60]. We adopt this approach in this paper for computing the difference between TCF and *global frame*. Thus, the acceleration is first converted to *global frame*, and then to TCF by using the orientation estimates of the sensor. After this transformation, the acceleration due to gravity is subtracted from the result since the accelerometer measurement includes the sum of gravity and linear acceleration. The video and IMU data will now be comparable with each other.

Fig. 5(a) compares the z-axis accelerometer data with double differentiated data of video locations before such coordinate transformations for a hand gesture. Evidently, the two data look dissimilar. On the other hand, after performing appropriate transformations,

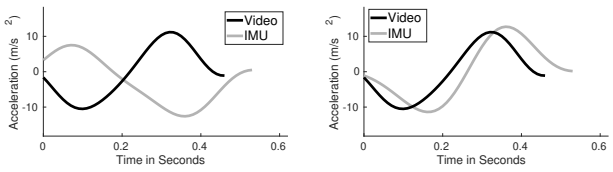

Figure 5: (a) The acceleration data from camera and video do not match before coordinate alignment between *TCF* and *LF* (b) The data from the two domains match well after coordinate alignment between *TCF* and *LF*

the two sources of data look similar as depicted in Fig. 5(b), which indicates the z-axis acceleration along TCF.

Double differences: While we considered tools like IMUSim [75] to convert location data from videos to IMU data (e.g. acceleration), there is no support for simulating finger joints. Therefore, we perform finite double differences as indicated by the equation below, as also explored in prior work [66, 74].

$$a_t = \frac{p_{t-\Delta t} + p_{t+\Delta t} - 2 \cdot p_t}{\Delta t^2}$$

This extracts accelerometer data a_t from locations p_t extracted from videos. While the IMU provides instantaneous acceleration, the finite time double differences is only an approximation. Choosing a smaller Δt reduces the error in approximation due to finite differences. However, smaller Δt also decreases the signal to noise ratio (SNR) of the generated acceleration signal because the change in location may be too small over a small time interval whereas the noise in the data is independent of time. We choose a value of Δt as 0.1s which provides a tradeoff that works well in practice. An example is depicted in Fig. 5(b) where finite differences are performed after the preprocessing steps such as low pass filtering.

Body size normalization and camera parameters: Difference in body sizes of users can create differences in the recorded sensor data even for the same gesture. In addition, the primary unit of estimate of locations from images is in pixels. Extraction of location in units of cms from public videos will need information or estimates about the camera parameters [42]. Towards handling body size differences as well as to eliminate the need for camera parameters, we normalize all location estimates from camera to the size of a standard human. In particular, we measure the shoulder length in pixels and scale it with factor such that the shoulder length is 27 cm. Such scaled locations are used for deriving synthetic accelerometer data. During testing, the accelerometer measurements from a human are similarly scaled depending on how their shoulder length compares with the standard length (27 cm). Fig. 6 shows an example of comparison between video and IMU data before and after normalization. Experimentally validated, the normalization step enables better similarity in sensor data despite the difference in body sizes of users and not having the camera parameters.

4.3 Extraction of Finger Orientation

Fig. 7 shows the metacarpophalangeal (MCP) and proximal interphalangeal (PIP) joints of the index finger. The angle made by the

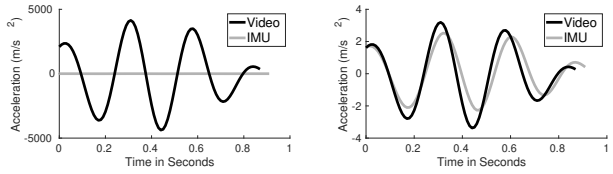

Figure 6: (a) The data from video and IMU domains can vary widely in magnitude because of differences in body sizes and units of measurements (b) Normalization techniques in *ZeroNet* renders the data from the two domains comparable

line joining these two joints with the vertical plane can be extracted from these videos. The same piece of information can be extracted from the orientation estimates of the IMU as indicated in the below equations.

$$y_{proj,xz} = R_f * R \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} - \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} R_f * R \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$angle = \arccos \frac{y^T_{proj,xz} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}{|y_{proj,xz}|}$$

Here, $y_{proj,xz}$ denotes the projection of the direction of the finger (line joining MCP and PIP joints) on the XZ plane. The sensor is roughly aligned such that its local y-axis is along the direction of the finger, but no careful calibration is needed. R is the 3×3 rotation matrix, R_f indicates the misalignmentment between the user's facing direction and the magnetic north. We compute this by adopting ideas from past work [56, 60]. Thus, the *angle* between MCP-PIP joints and the vertical axis as indicated above will be used as a virtual orientation data for training the ML models in *ZeroNet*. While the orientation estimates of a IMU sensor is 3 dimensional, we restrict ourselves to extracting the 1 dimensional *angle* information as discussed above mainly because: (i) We can extract it reliably and compares well with the same information extracted from IMU. (ii) We found that estimating rotation along the axis of the index finger although possible in theory from the information extracted from videos, proved to be unreliable and erroneous in practice.

5 GESTURE RECOGNITION MODELS WITH SYNTHETIC TRAINING DATA

We explore two methods for exploiting the training data extracted from videos for performing gesture recognition on IMU: (i) A simple DTW based model (ii) A Convolutional Neural Network based machine learning model

5.1 Dynamic Time Warping

We begin by using dynamic time warping (DTW) [16] to compare the IMU data from an unknown user gesture to video-based training dataset for gesture recognition. Briefly, DTW is a pattern matching technique that inspects the overall shape of two signals to determine their similarity. For example, Fig. 8(a) shows the z-axis

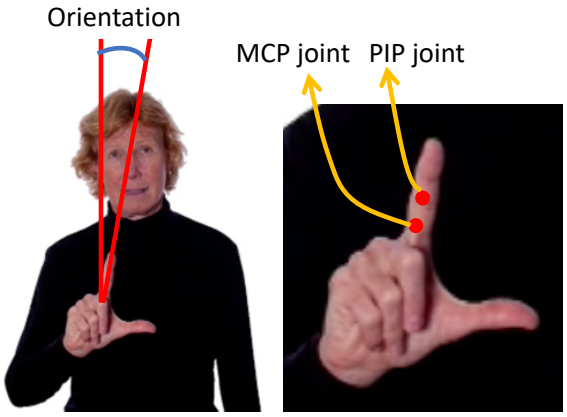

Figure 7: The angle depicted here can be extracted from videos and used as a training data for inferences on IMU

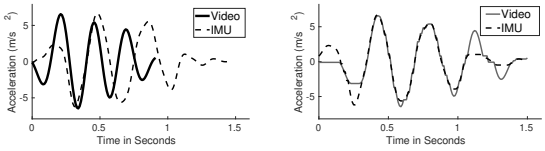

Figure 8: (a) Accelerometer data for "More" extracted from video of one user in comparison with IMU data of another user (b) Data from IMU is compressed and stretched to match with video by DTW

accelerometer data from IMU and the synthetic accelerometer data extracted from a video of the same gesture. Although the overall shape is similar, parts of the motion traces happen at a faster rate or earlier for IMU while other parts happen slower. DTW uses a dynamic programming optimization to minimally compress and stretch the two sequences relative to each other such that they provide the best overlap. Fig. 8(b) shows the two sequences after DTW optimization. DTW is known to do a good job of matching such series with similar overall shape. The residual differences between the two series determines the similarity score among them. The similarity score of an unknown gesture is compared with all gestures in the training data. The gesture with the best match would correspond to the correct gesture with high probability. The 3-axis accelerometer data and the orientation of the index finger is used for performing the DTW matching as described above.

5.2 Convolutional Neural Networks

Towards increasing the robustness of recognition, we take a data-driven ML approach in addition to DTW. The architecture of the model is depicted in Fig. 9. The success of ML models depend on availability of large scale high quality training datasets. In addition to extracting the training data from videos, we exploit the following data augmentation techniques to ensure stability, robustness, and convergence of the above ML model.

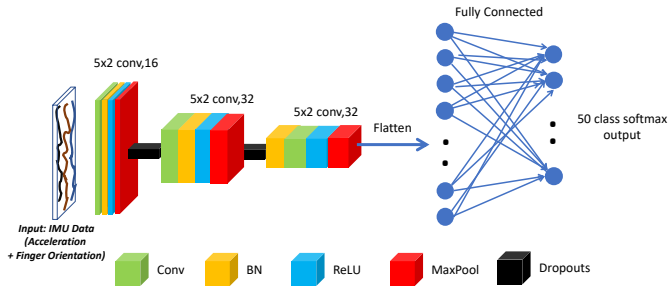

Figure 9: The CNN based machine learning model in *ZeroNet*

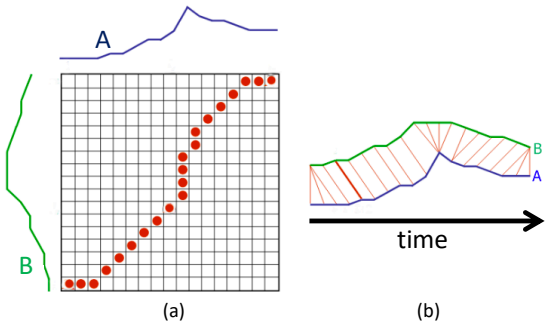

Figure 10: DTW alignment matrix between two sequences A and B. Pictures adopted from [1]

DTW based augmentation: The performance of gestures will vary widely across users. The speed of hand motion is one of the metrics that can vary across users. Various parts of the gesture might be performed at a faster or slower pace by different users. Towards making the ML models robust to such variations, we augment the training data by injecting such variations into existing training data. In particular, we stretch and compress different parts of the training data with different factors to create new training data from existing samples.

Fig. 10 shows an example where two sequences A and B are aligned using DTW. Fig. 10 (b) shows the correspondence between samples in the two sequences, whereas Fig. 10 (a) depicts the same in matrix form. Given a training data sample A, we generate random matrices similar to Fig. 10 (a) to create dynamically stretched and compressed versions of the training data sample. In creating these matrices, we resample the original time series of the training data with a stochastic non-uniform sampling such that compression/expansion ratio varies between 0.25 to 2. Appropriate interpolation strategies are used since the resampling positions may not coincide exactly with the positions in the original time series. Fig. 11 shows an example where two variants of new training data has been created from an existing training data.

Orientation Variation: Similar to variations in gesturing where users perform at different speeds, the orientation of the hand can vary during motion. Such variations can also happen because of

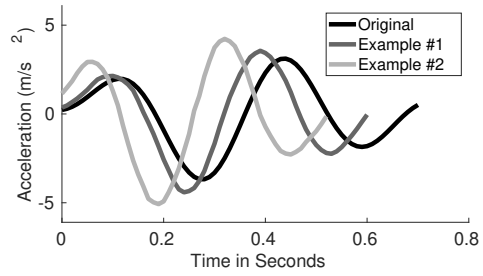

Figure 11: DTW synthetic training data

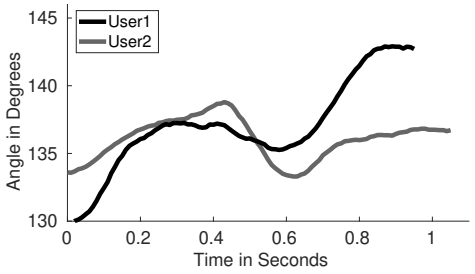

Figure 12: Variation in orientation across users

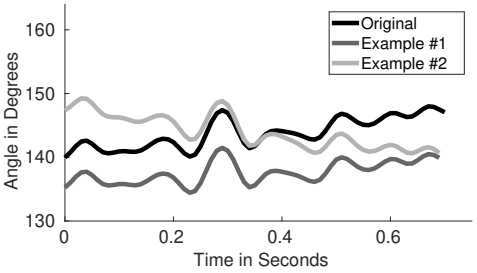

Figure 13: Examples of synthetic orientation data

minor changes in the sensor wearing position or orientation across users. Fig. 12 shows an example where the same gesture is performed by two users with a minor shift in the hand orientation. The ML model must be robust for adapting to such natural variations. Therefore we augment training datasets emulating variations in hand orientation while gesturing. The injected variations range from 0 to 10 degrees, but they are not random, rather they ensure smoothness and continuity thus emulating a realistic gesture with small changes in orientation. Fig. 13 shows examples of augmented data with varying orientations for a given gesture.

Temporal Clipping: We also hypothesize that the start and end periods of performance of gestures by several users will vary. Different users might start the gesture from slightly different positions as well as end the gesture prematurely or continue with extra motions beyond the gesture. To help the model generalize under such diversity, we augment training data by introducing versions of the training data with minor extrapolations or trimming of samples at the begin and end of the gestures. Fig. 14 shows an example where

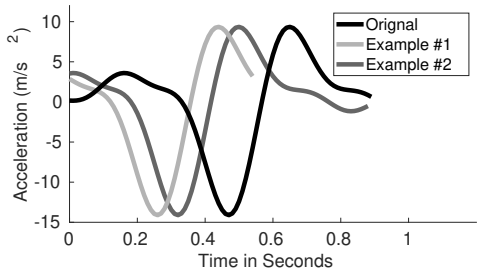

Figure 14: Examples of temporal clipping

two variants of synthetic training data are added with random clipping at the beginning and end of an original training data.

6 IMPLEMENTATION AND EVALUATION

Implementation: The sensor frontend includes an Mbient sensor [9] as described in Sec. 3 which is worn on the index finger as a ring. The 9-axis IMU data including accelerometer, gyroscope, and magnetometer data is streamed to a smartphone. *ZeroNet* is implemented on a combination of desktop and smartphone devices. The machine learning architecture is implemented using TensorFlow [10] packages and the training is performed on a desktop with Intel i7-8700K CPU, 16GB RAM memory, and Nvidia GTX 1080 GPU. We use the Adam optimizer[36] with a learning rate of 1e-3, β_1 of 0.9 and β_2 of 0.999. To avoid over-fitting issues that may happen in the training process, we apply the L2 regularization[17] on each CONV layer with a parameter of 0.01 and also add dropouts[70] with a parameter of 0.1 following each RELU activations. Once a model is generated from training, the inference is done entirely on a smartphone device using TensorFlowLite [26] on a Samsung Galaxy S20 smartphone with Android operating system.

User Study: All reported results in the paper are generated from a systematic user study campaign. The study evaluates the classification accuracy of 50 gestures that represent the top 50 ASL words. The training data is extracted from the following video source [8]. We recruit 10 users aged 21-32 and weighs between 47 to 96kgs. It includes 7 males and 3 females. During the data collection process, the user is first shown the video of a gesture. The user practices performing the gesture several times. When the user feels comfortable performing the gesture correctly, we let the user perform the gesture 5 more times and we record the sensor data during this period. After this process, we repeat the procedure for the next gesture until we finish collection of the data for all 50 gestures. The entire recorded dataset during the study is solely used as a 'test data' since the training data is extracted entirely from videos.

We specifically aim to answer the following questions.

- What is the overall gesture recognition accuracy? *(Figs. 15(a), Figs. 20)*
- Is the accuracy consistent across diverse gestures? *(Figs. 15(b), Figs. 15(c))*
- How does the accuracy vary across users? *(Figs. 15)*
- In cases of errors in recognition, what is the rank of the correct gesture among all the 50 gestures? *(Figs. 16)*
- How does the accuracy vary with the speed of gesturing? *(Figs. 17)*
- How does the accuracy vary with sensor placement on the hand? *(Figs. 18)*
- What is the accuracy of the model transferred to the left hand? *(Figs. 19)*
- What is the role of various techniques of data augmentation in the final accuracy metric? *(Figs. 21)*
- How does the accuracy vary with the size of the synthetic dataset? *(Figs. 22)*
- How does *ZeroNet* compare with models fine-tuned with real-data or models fully trained on real data? *(Figs. 22, Figs. 23)*
- What is energy, latency, and compute profile of executing the ML models on embedded devices? *(Figs. 24)*

Robustness to sensor wearing positions and diverse gesture patterns: Fig. 15(a) depicts the overall accuracy as a function of users. Evidently, the accuracy is stable across users, body sizes, motion patterns etc. In addition, the sensors were mounted naturally on the fingers with y-axis roughly along the direction of the index finger. There was no special calibration and hence the position/orientation across users would naturally vary. However, the accuracy is robust to such variations. This is because of the inbuilt robustness to such natural variations through the data augmentation techniques incorporated in the design of *ZeroNet*. While the top-1 accuracy is 82.4%, the top-3 accuracy is around 94.8% which indicates promise for future improvements.

Accuracy over gestures: Fig. 15(b) shows the confusion matrix over all 50 gestures in our dictionary. The performance is consistent across all gestures. However, in certain special cases, such as the gesture for "mother", and "father", there can be miss-classifications because the index finger motion for these two gestures are very similar. Fig. 15(c) shows the confusion matrix for top-3 accuracy which shows a higher accuracy because in many cases of miss-classifications the correct word is occupies the second or third place in the rank of softmax probabilities.

Rank of incorrect gestures: We provide further breakup of cases where the top identified gesture is incorrect. Fig. 16 shows the rank of the correct gesture in case of erroneous detections. Evidently, majority of the cases are rank-2 and 70.5%, 83.0% of cases are in top-3 and top-5 ranks respectively. This indicates that appropriate application specific prior information or context can be exploited to further imporve the accuracy of *ZeroNet*.

Accuracy over speed: Fig. 17 provides a breakup of accuracy of gestures executed at varying speeds. Note that in addition to some gestures being inherently slow or fast paced, variations in pace can also occur because of user diversity. Regardless of the reason of variation, the accuracy is robust at various possible speeds.

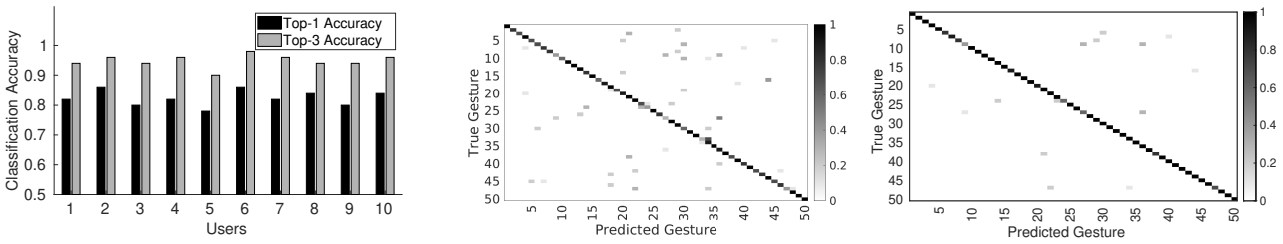

Figure 15: (a) Overall Accuracy vs Users (b) Top-1 Accuracy vs Gestures (c) Top-3 Accuracy vs Gestures

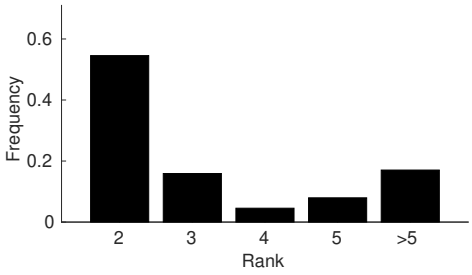

Figure 16: Rank of correct gestures for erroneous cases.

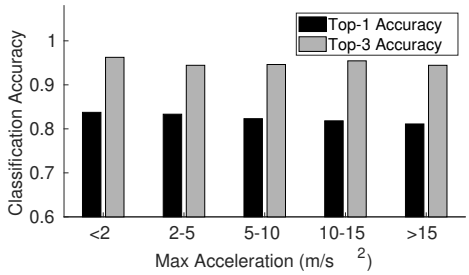

Figure 17: Accuracy over speed of gesturing

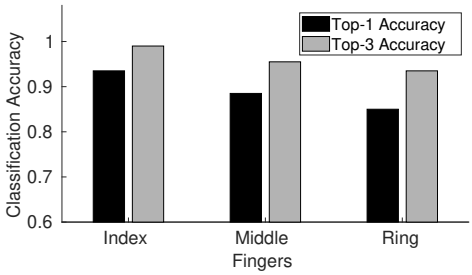

Figure 18: *ZeroNet* can generate training data for any finger position, thus facilitating optimal sensor positioning

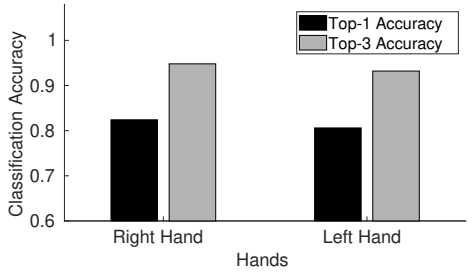

Figure 19: Model transfer from right to left hand

Accuracy vs Finger Position: An advantage of harvesting training data from videos is that optimal sensor placement can be determined for any given application where there is a tradeoff in number of sensors that can be used due to reasons including usability, accuracy, power consumption etc. We conduct a small study to determine the optimal sensor placement among index, middle, and ring fingers for the top 20 gestures from our video dataset [8]. The little finger and thumb were excluded in the study since it is not comfortable to wear sensors on those fingers. Fig. 18 shows that the top-1 accuracy values are 93.4%, 88.3%, and 85.1% for index, middle, and ring fingers respectively. This indicates that the optimal sensor placement among the three fingers is the index finger for the set of gestures considered in this application.

Model transfer between right and left hands: Fig. 19 shows the accuracy when the left hand was used in gesturing. This is useful when the training data from videos of right-handed users is used for performing inferences on left-handed users. The training data captured from the right hand was appropriately mirrored to emulate a training data for the left hand. This includes making the x-axis in Fig. 4 negative and projecting the acceleration and orientation to the new TCF relative to the left hand. The transformed training data was used to train the ML model in Fig. 9 to perform inferences when the sensor is worn on the left hand. Evidently, the accuracy for such inferences is same as the right hand.

Performance comparison across techniques: Fig. 20 provides a breakup of accuracy across techniques. Basic DTW already achieves a reasonable accuracy of 59.4%. On the otherhand, the accuracy of the basic CNN model is slightly lower than DTW because of the inability to generalize to diversity in user motion patterns. However, data augmentation techniques in *ZeroNet* can make the CNN model robust to speed of gesturing, sensor positions, orientation variation, noise etc, thus boosting the accuracy to 82.4%.

Breakup of performance gain from data augmentation: DTW based augmentation, rotation based augmentation and time clipping individually achieve accuracies of 70.7%, 53.4%, and 60.8% respectively as shown in Fig. 21. DTW-based augmentation performs the best while the other techniques also offer non-trivial gain

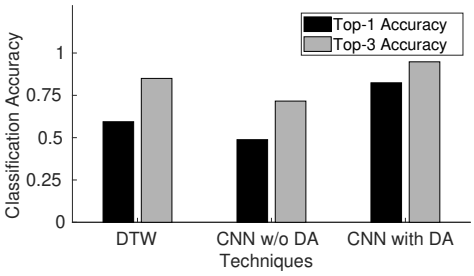

Figure 20: Performance comparison across: (i) DTW (ii) CNN (iii) CNN with data augmentation (DA)

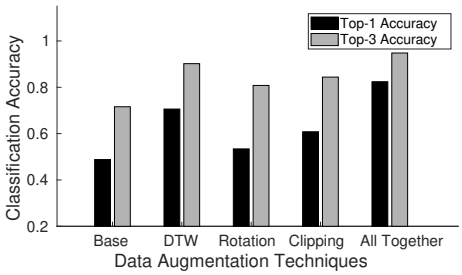

Figure 21: Role of individual data augmentation techniques

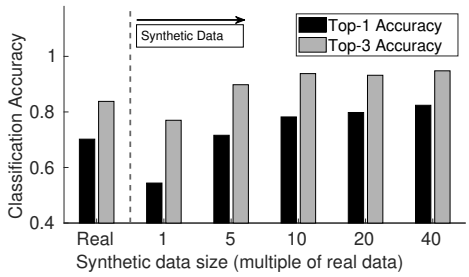

Figure 22: Diversity in synthetic data leads to better generalization of the CNN model

in performance relative to a baseline without data augmentation. However, combining all of them yields the best performance.

Training with Synthetic vs Real Data The first bar in Fig. 22 shows the performance accuracy of training with real data alone. Evidently, the small size of the data leads to overfitting and poor generalization thus leading to overall low accuracy. On the other hand, the last bar depicts the effect of training with synthetic data which together with data augmentation techniques leads to better generalization of the ML models and higher accuracy.

Effect of the size of synthetic data: Fig. 22 depicts the performance of the CNN model as a function of the size of the synthetic data. The x-axis label denotes the size of the synthetic data in multiples of the size of the real data. Evidently, higher size of synthetic data creates more robustness in the training examples that the ML model sees during training. Thus, the overall accuracy improves

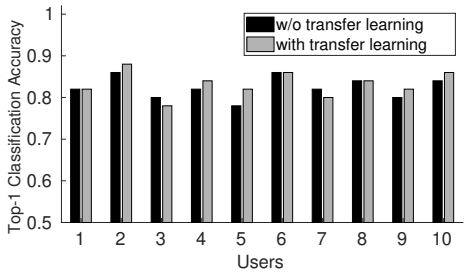

Figure 23: Model fine tuning with real IMU data

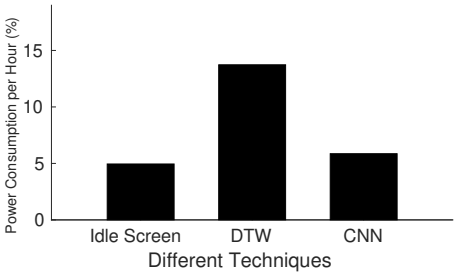

Figure 24: The power consumption profile the CNN model is better than simple DTW because of builtin optimizations in TensorFlowLite [26]

with size, ultimately saturating when the size of the synthetic data is 40 times of the real data.

ML models fine tuned with real IMU data: The CNN model that was trained in *ZeroNet* using synthetic IMU data from videos was fine-tuned [52, 76] with real IMU data. Fig. 23 depicts the performance over users. Leave-one-out cross-validation was adopted across users. The fine tuning improves the performance only marginally. We believe this is because the data augmentation techniques sufficiently cover the space of variations thus generalizing the CNN model to the maximal extent.

Energy, latency, and compute: we use Batterystats and Battery Historian[7] toolkits for profiling the energy of the TensorflowLite model for inference using CNN and the DTW-based classification model. We compare the difference between energy consumption in two states (i) When the device is idle with screen on. (ii) The device is making inferences at a rate of 2 gestures per second. The idle display-screen on discharge rate 4.95% per hour while the discharge rates for various techniques is depicted in Fig. 24. Evidently, the power consumption profile of the CNN model is very low. We believe the CNN model is more efficient than the simple DTW because of inbuilt optimizations within the TensorFlowLite library [26]. The latency results are very much correlated with power consumption results. In particular, each classification takes 2.2ms on average with CNN whereas it takes 266.4ms with the DTW model. We believe the overall power consumption and latency profiles of the CNN model enables energy efficient real time performance.

7 RELATED WORK

Inertial Sensors: Inertial sensors have been used in many localization and gesture tracking applications. UnLoc [72] fuses information from smartphone sensors for extracting characteristic fingerprints in indoor environments for localization. RisQ [49] recognizes smoking gestures for appropriate intervention measures using smartwatches. Similarly, smartwatches are used for eating activity recognition [58] and measuring calorie intakes. Smart rings are also being used for ASL gesture recognition in recent times [39, 40]. DUI [41] detects blood alcohol level based on user performance on smartphone activities. Other applications have been explored in the areas of augmented and virtual reality, sports analytics, smart-health, and security [35, 45, 50, 77]. In contrast to these works that create training datasets with user studies, crowdsensing etc, *ZeroNet* exploits harvesting training data from publicly available videos.

Vision: Depth cameras including kinect[4] and leap motion [3] sensors have revolutionized the gaming industry by gesture interfaces. Use of depth camera is one way to capture finger motion. However, advances in machine learning, availability of large training datasets as well as techniques for creation of synthetic datasets have enabled precise tracking of finger motion even from monocular videos that do not contain depth information [18, 31, 42]. While such works are truly transformative in nature, we believe wearable based solutions have benefits over vision based approaches which are susceptible to occlusions, lighting, and resolution. In addition, wearable devices offer ubiquitous solution with continuous tracking without the need of an externally mounted camera.

Radio Frequency (RF): RF including WiFi, RFID, and mmWave hardware have been used for a number of human activity recognition applications. WiSee [51] can detect hand gestures by measuring doppler shifts from WiFi reflections. 3D pose of the human body has been detected even behind occlusions such as Walls using wireless body reflections [33, 81]. Heart rate, breathing, and physiological signals of interest to healthcare applications have been detected using RF signals [79, 80]. Google project Soli [63] can detect fine grained finger gestures using mmWave reflections. While RF based tracking, like vision, is completely passive, we believe the advantage of wearable device is being completely ubiquitous without the need for any external infrastructure.

Transfer Learning from Videos: Deep Inertial Poser [29] uses synthetic data from motion capture videos (from cameras like ViCON[69]) instead of public videos for training human pose tracking algorithms with 6 on-body IMUs. Such motion capture cameras can provide high quality training data with mm level accuracy. However, creating such datasets requires 6-8 costly ViCON cameras. We believe using publicly available videos is an easier alternative. More recently, several innovative works [38, 55, 66, 74, 78] have explored the use of YouTube-like videos for training human activity recognition (HAR) on wearable sensors. In contrast to such works that classify tens of large scale motion activities (running, sitting, eating etc.), *ZeroNet* performs recognition of fine grained finger motions over a larger class of gestures with potential to applications in augmented and virtual reality, sign language recognition etc. In addition, *ZeroNet* fuses the harvested training data with data-augmentation techniques for better robustness of ML models.

Data Augmentation: Data augmentation enriches the quality of datasets to help ML models generalize well and exhibit higher accuracy and robustness with limited quantity of training data. Transformation such as rotation, scaling, translation and elastic distortions on images have been explored to create more training data from existing datasets. [21, 57, 61, 71]. Similarly, image cropping, flipping, color shifting, and whitening are other techniques to create new training data from existing datasets [37]. In the area of automatic speech recognition (ASR), data augmentation techniques such as frequency axis distortions[34], speech rate variations, vocal tract normalization[32] etc have been explored to improve the accuracy. In a similar spirit, *ZeroNet* incorporates ideas in data augmentation for IMU datasets for better accuracy, robustness, and generalizability of ML models. This is particularly important in the context of IMU data since there is no large scale public datasets like computer vision or speech. Data augmentation techniques have been explored in the context of wearable sensing for parkinson disease gait monitoring [68] and construction activity monitoring [53]. More recently, data augmentation for human activity recognition has been extensively studied in [20] for several benefits including robustness to sensor wearing positions. In contrast to these works, *ZeroNet* performs a fusion of data extraction from videos and combines it with data augmentation techniques to enable inferences on IMU devices without any training overhead.

Transfer Learning, Domain Adaptation, and Zero-shot Learning: Transfer-learning based domain adaptation is popular in vision and speech. For example, AlexNet model [37] pretrained on ImageNet database [24] has been fine-tuned for classifying images in medical domain[83], remote-sensing [27] and breast-cancer [44]. Similarly, a pre-trained BERT language model [25] has been fine-tuned for tasks in text-summarizing [76], question answering [52] etc. Adversarial domain adaptation [67] using generative adversarial networks (GAN) is popular. Here, an unsupervised game theoretic strategy is used to transform the distribution of the feature representations from the target-domain into the distribution of the source-domain on which the model was trained. If successful, the model trained on the source domain is directly useful for performing inferences on a target domain. Similarly, other architectures for learning feature transformations to adapt the feature representations from a source domain to a target domain have been proposed [64]. However, such techniques are hard to apply to our problem domain since this still requires enough real training data (atleast in unlabelled form) from IMU to achieve sufficient convergence of the domain adaptation process. Furthermore, each user's finger motion pattern as well as natural variations in sensor wearing positions could lead to different distributions in the sensor data [15, 20] thus entailing more real training data under each setting. On the other hand, *ZeroNet* performs comparable to semi-supervised domain adaptation techniques [27, 83] which need partial labelled real IMU data and even outperforms models fully trained on our own real

IMU dataset. We believe *ZeroNet*'s ability to provide promising accuracy without any training overhead is a first step towards generating data for unsupervised domain adaptation. Our approach is related to zero-shot learning [62], where a ML model is trained to predict classes for which no training examples has been observed. Appropriate representations are learnt for both training examples and class labels. By learning the mapping between representations of known training examples and their classes, the mapping between representations of a new example is made even if it belongs to an unseen class. One difference between *ZeroNet* and classical zero-shot learning is that zero-shot learning needs training data from the target domain for some classes, whereas *ZeroNet* does not need training data for any classes.

8 DISCUSSION AND FUTURE WORK

Exploiting large scale video datasets: *ZeroNet* only scratches the surface in harvesting training data from videos. 300 hours of videos are uploaded to YouTube every minute for human activities ranging from sports, tutorials, physical exercises, speech, daily activities (cooking, eating, jogging) etc. Exploiting more videos for building ML models can enhance the robustness.

Automated data augmentation: In *ZeroNet*, the amount of perturbations introduced in the data for augmentation is fixed. Automated data augmentation [23] is an active area of research where the parameters for data augmentation can be modeled as a learning problem. We plan to incorporate the innovations from this area into *ZeroNet* as a part of the future work.

Augmented and Virtual Reality applications: AR and VR applications benefit from fine grained tracking of hand and finger locations. Towards pushing the limits of accuracy, *ZeroNet* will exploit video-based training data for free form tracking of 3D finger joint locations. Similar to our analysis on finding the optimal finger to place the sensor, enough training data can be generated from videos for analyzing the tradeoff between number and position of placement of sensors and the expected accuracy.

9 CONCLUSION

Application of ML models for finger gesture recognition can enable a number of exciting applications. However, unlike computer vision and speech, there is a dearth of large scale training data for developing robust and sophisticated ML models. Towards addressing this problem, this paper presents *ZeroNet* that extracts training data from publicly available videos of annotated finger gestures. Appropriate data augmentation techniques are exploited to increase the robustness and generalizability of ML models to natural patterns in user gesturing. A systematic user study with 10 users over 50 gestures demonstrates a top-1% accuracy of 82.4% and a top-3% accuracy of 94.8% with zero training overhead. While the results are promising, we believe we have only scratched the surface. Exploiting the availability of large scale video datasets that are publicly available can enhance the start of the art in a number of applications including augmented reality, virtual reality, healthcare and rehabilitation etc.

REFERENCES

[1] Dynamic time warping (dtw) algorithm for time series analysis. https://medium.com/datadriveninvestor/dynamic-time-warping-dtw-d51d1a1e4afc.
[2] Knuckleball Grip, Part 3: Depth of the Baseball. "https://knuckleballnation.com/how-to/knuckleballgrip3/".
[3] Leap motion developer. https://developer.leapmotion.com/.
[4] Microsoft kinect2.0. https://developer.microsoft.com/en-us/windows/kinect.
[5] Motiv ring | 24/7 smart ring | fitness + sleep tracking | online security. https://mymotiv.com/.
[6] Oura ring: The most accurate sleep and activity tracker. https://ouraring.com/.
[7] Profile battery usage with batterystats and battery historian. https://developer.android.com/topic/performance/power/setup-battery-historian.
[8] Signing savvy – asl sign language video dictionary. https://www.signingsavvy.com/.
[9] Wearables for motion tracking + wireless environment monitoring. https://mbientlab.com/store/adhesive-sensor-research-kit/.
[10] Abadi, M., et al. Tensorflow: A system for large-scale machine learning. In *OSDI* (2016), pp. 265–283.
[11] Acharya, S., et al. Towards a brain-computer interface for dexterous control of a multi-fingered prosthetic hand. In *2007 3rd International IEEE/EMBS Conference on Neural Engineering* (2007), IEEE, pp. 200–203.
[12] Agostino, R., et al. Impairment of individual finger movements in parkinson's disease. *Movement disorders 18*, 5 (2003), 560–565.
[13] Ahmed, M. A., et al. A review on systems-based sensory gloves for sign language recognition state of the art between 2007 and 2017. *Sensors 18*, 7 (2018), 2208.
[14] Bachlin, M., et al. Wearable assistant for parkinson's disease patients with the freezing of gait symptom. *IEEE Transactions on Information Technology in Biomedicine 14*, 2 (2009), 436–446.
[15] Barbosa, P., et al. Unsupervised domain adaptation for human activity recognition. In *International Conference on Intelligent Data Engineering and Automated Learning* (2018), Springer, pp. 623–630.
[16] Berndt, D. J., et al. Using dynamic time warping to find patterns in time series. In *KDD workshop* (1994), vol. 10, Seattle, WA, pp. 359–370.
[17] Bertero, M., et al. The stability of inverse problems. In *Inverse scattering problems in optics*. Springer, 1980, pp. 161–214.
[18] Cai, Y., et al. Weakly-supervised 3d hand pose estimation from monocular rgb images. In *ECCV* (2018), pp. 666–682.
[19] Cao, Z., et al. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1812.08008* (2018).
[20] Chang, Y., et al. A systematic study of unsupervised domain adaptation for robust human-activity recognition. *ACM IMWUT* (2020).
[21] Ciregan, D., et al. Multi-column deep neural networks for image classification. In *CVPR* (2012), IEEE, pp. 3642–3649.
[22] Cordella, F., et al. Patient performance evaluation using kinect and monte carlo-based finger tracking. In *IEEE BioRob* (2012), IEEE, pp. 1967–1972.
[23] Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., and Le, Q. V. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2019), pp. 113–123.
[24] Deng, J., et al. Imagenet: A large-scale hierarchical image database. In *IEEE CVPR* (2009), Ieee, pp. 248–255.
[25] Devlin, J., et al. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).
[26] Google. Deploy machine learning models on mobile and IoT devices. "https://www.tensorflow.org/lite", 2019.
[27] Han, X., et al. Pre-trained alexnet architecture with pyramid pooling and supervision for high spatial resolution remote sensing image scene classification. *Remote Sensing 9*, 8 (2017), 848.
[28] Hesse, S., et al. A new electromechanical trainer for sensorimotor rehabilitation of paralysed fingers: a case series in chronic and acute stroke patients. *Journal of neuroengineering and rehabilitation 5*, 1 (2008), 21.
[29] Huang, Y., et al. Deep inertial poser: learning to reconstruct human pose from sparse inertial measurements in real time. *ACM TOG 37*, 6 (2018), 1–15.
[30] Ionescu, C., et al. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence 36*, 7 (2013), 1325–1339.
[31] Iqbal, U., et al. Hand pose estimation via latent 2.5 d heatmap regression. In *ECCV* (2018), pp. 118–134.
[32] Jaitly, N., et al. Vocal tract length perturbation improves speech recognition. In *Proc. ICML Workshop on Deep Learning for Audio, Speech and Language* (2013).
[33] Jiang, W., et al. Towards 3d human pose construction using wifi. In *ACM MobiCom* (2020), pp. 1–14.
[34] Kanda, N., et al. Elastic spectral distortion for low resource speech recognition with deep neural networks. In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding* (2013).
[35] Kim, M., et al. Golf swing segmentation from a single imu using machine learning. *Sensors* (2020).

[36] Kingma, D. P., et al. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[37] Krizhevsky, A., et al. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (2012).

[38] Kwon, H., et al. Imutube: Automatic extraction of virtual on-body accelerometry from video for human activity recognition. *arXiv preprint arXiv:2006.05675* (2020).

[39] Liu, Y., et al. Application informed motion signal processing for finger motion tracking using wearable sensors. In *IEEE ICASSP 2020*.

[40] Liu, Y., et al. Finger gesture tracking for interactive applications: A pilot study with sign languages. *ACM IMWUT 2020*.

[41] Mariakakis, A., et al. Drunk user interfaces: Determining blood alcohol level through everyday smartphone tasks. In *CHI Conference on Human Factors in Computing Systems* (2018).

[42] Mueller, F., et al. Ganerated hands for real-time 3d hand tracking from monocular rgb. In *IEEE CVPR* (2018), pp. 49–59.

[43] Murguialday, A. R., et al. Brain-computer interface for a prosthetic hand using local machine control and haptic feedback. In *2007 IEEE 10th International Conference on Rehabilitation Robotics* (2007), IEEE, pp. 609–613.

[44] Nawaz, W., et al. Classification of breast cancer histology images using alexnet. In *International conference image analysis and recognition* (2018), Springer.

[45] Nirjon, S., et al. Typingring: A wearable ring platform for text input. In *ACM MobiSys* (2015).

[46] Oura ring review. https://www.wareable.com/health-and-wellbeing/oura-ring-2018-review-6628, 2018.

[47] Oura ring - what we learned about the sleep tracking ring. https://www.cnbc.com/2019/12/20/oura-ring-review---what-we-learned-about-the-sleep-tracking-ring.html, 2019.

[48] Oura ring review - the early adopter catches the worm. https://www.androidauthority.com/oura-ring-2-review-933935/, 2019.

[49] Parate, A., et al. Risq: Recognizing smoking gestures with inertial sensors on a wristband. In *ACM MobiSys* (2014), pp. 149–161.

[50] Park, K.-B., et al. Deep learning-based smart task assistance in wearable augmented reality. *Robotics and Computer-Integrated Manufacturing* (2020).

[51] Pu, Q., Gupta, S., Gollakota, S., and Patel, S. Whole-home gesture recognition using wireless signals. In *Proceedings of the 19th annual international conference on Mobile computing & networking* (2013), ACM, pp. 27–38.

[52] Qu, C., et al. Bert with history answer embedding for conversational question answering. In *ACM SIGIR* (2019), pp. 1133–1136.

[53] Rashid, K. M., et al. Times-series data augmentation and deep learning for construction equipment activity recognition. *Advanced Engineering Informatics 42* (2019), 100944.

[54] Reiss, A., et al. Introducing a new benchmarked dataset for activity monitoring. In *IEEE ISWC* (2012), pp. 108–109.

[55] Rey, V. F., et al. Let there be imu data: generating training data for wearable, motion sensor based activity recognition from monocular rgb videos. In *ACM UbiComp/ISWC* (2019), pp. 699–708.

[56] Roy, N., et al. I am a smartphone and i can tell my user's walking direction. In *ACM MobiSys* (2014).

[57] Sato, I., et al. Apac: Augmented pattern classification with neural networks. *arXiv preprint arXiv:1505.03229* (2015).

[58] Sen, S., et al. Annapurna: An automated smartwatch-based eating detection and food journaling system. *Pervasive and Mobile Computing* (2020), 101259.

[59] Shaeffer, D. K. Mems inertial sensors: A tutorial overview. *IEEE Communications Magazine 51*, 4 (2013), 100–109.

[60] Shen, S., Wang, H., and Roy Choudhury, R. I am a smartwatch and i can track my user's arm. In *Proceedings of the 14th annual international conference on Mobile systems, applications, and services* (2016), ACM, pp. 85–96.

[61] Simard, P. Y., et al. Best practices for convolutional neural networks applied to visual document analysis. In *Icdar* (2003), vol. 3.

[62] Socher, R., et al. Zero-shot learning through cross-modal transfer. In *NeurIPS* (2013), pp. 935–943.

[63] Google project soli. https://atap.google.com/soli/, 2020.

[64] Sun, B., et al. Deep coral: Correlation alignment for deep domain adaptation. In *ECCV* (2016), Springer, pp. 443–450.

[65] Susanto, E. A., et al. Efficacy of robot-assisted fingers training in chronic stroke survivors: a pilot randomized-controlled trial. *Journal of neuroengineering and rehabilitation 12*, 1 (2015), 42.

[66] Takeda, S., et al. A multi-sensor setting activity recognition simulation tool. In *ACM UbiComp* (2018), pp. 1444–1448.

[67] Tzeng, E., et al. Adversarial discriminative domain adaptation. In *IEEE CVPR* (2017), pp. 7167–7176.

[68] Um, T. T., et al. Data augmentation of wearable sensor data for parkinson's disease monitoring using convolutional neural networks. In *ACM ICMI* (2017).

[69] Vicon - award winning motion capture system. https://www.vicon.com/, 2020.

[70] Wager, S., et al. Dropout training as adaptive regularization. In *Advances in neural information processing systems* (2013), pp. 351–359.

[71] Wan, L., et al. Regularization of neural networks using dropconnect. In *ICML* (2013), pp. 1058–1066.

[72] Wang, H., et al. No need to war-drive: Unsupervised indoor localization. In *ACM MobiSys* (2012), pp. 197–210.

[73] Xiang, D., et al. Monocular total capture: Posing face, body, and hands in the wild. In *IEEE CVPR* (2019).

[74] Xiao, F., et al. A deep learning method for complex human activity recognition using virtual wearable sensors. *arXiv preprint arXiv:2003.01874* (2020).

[75] Young, A. D., et al. Imusim: A simulation environment for inertial sensing algorithm design and evaluation. In *ACM/IEEE IPSN* (2011).

[76] Zhang, H., et al. Pretraining-based natural language generation for text summarization. *arXiv preprint arXiv:1902.09243* (2019).

[77] Zhang, H., et al. Pdlens: smartphone knows drug effectiveness among parkinson's via daily-life activity fusion. In *ACM MobiCom* (2020).

[78] Zhang, S., et al. Deep generative cross-modal on-body accelerometer data synthesis from videos. In *ACM UbiComp/ISWC* (2020), pp. 223–227.

[79] Zhao, M., et al. Emotion recognition using wireless signals. In *ACM MobiCom* (2016), pp. 95–108.

[80] Zhao, M., et al. Learning sleep stages from radio signals: A conditional adversarial architecture. In *ICML* (2017), pp. 4100–4109.

[81] Zhao, M., et al. Through-wall human mesh recovery using radio signals. In *IEEE ICCV* (2019), pp. 10113–10122.

[82] Zhou, P., et al. Use it free: Instantly knowing your phone attitude. In *ACM MobiCom* (2014), pp. 605–616.

[83] Zhou, Z., et al. Fine-tuning convolutional neural networks for biomedical image analysis: actively and incrementally. In *IEEE CVPR* (2017), pp. 7340–7351.