KU LEUVEN

imec

FACULTY OF ENGINEERING SCIENCE

A Gesture to Learn: Predicting Hand Positions with Deep Neural Network

Authors:

Simon Forceville[1]

Gilda Maria Ida Marsicano[1]

Andrea Obando Carmona[2]

Carlos Jair Cordoba Contreras[2]

[1]Master of Nanoscience, Nanotechnology and Nanoengineering

[2]Erasmus Mundus Master of Science in Nanoscience and Nanotechnology

P & O Nanoscience, Nanotechnology and Nanoengineering (H0P61A)

2025 - 2026

Supervisors:

Victor Chuman Alvarado

Andrea El Haddad

Content

List of Figures

Abstract

Wristband-based gesture recognition is an emerging interface technology which allows individuals to interact with assistive technologies. To do it, an array of sensors is integrated inside the wearable, and, in principle, they can read the signals that the forearm produces when executing a pose. This reading is based on muscle activity. The bottleneck for developing such a device is the requirement of large datasets that are expensive to collect and tied to specific applications that might not be useful for researchers.

This project addresses the scarcity of datasets by building an end-to-end simulation pipeline that produces synthetic ultrasound data from public forearm MRI scans and uses that data to recognize six forearm poses static of four subjects, based on ultrasound signals given by different transducer geometries. The pipeline operates at three stages. First, segmentation of 2D cross-sectional slices is produced by identifying different tissues of the forearm, and an augmentation process using MONAI increases the number of samples. Second, a simulation setup converts the tissue maps into acoustic properties (sound speed, density, attenuation) and pushes them through the *k-Wave* toolbox. Three transducer geometries are simulated at this stage, resembling wristband sensors: a linear, clustered, and smartwatch-like array. Third, the resulting ultrasound datasets are analyzed by a set of machine learning models (LogReg, MLP, 1D CNN) to evaluate the best performance under a LOSO protocol.

On the linear array dataset, the three-block, ~ 248 k-parameter 1D CNN reaches a LOSO mean accuracy of 0.717 on validation and 0.422 on test. On real data, chance baseline drops to 0.167, making a +49% relative improvement over the original two-block design. An optimization is present by restricting the label to the four most anatomically poses, raising the LOSO test accuracy to $\sim$0.80. Smartwatch geometry underperforms compared to the linear and clustered array. Across architectures and geometries a residual gap of ~ 0.30 between validation and test accuracy persists; under the same protocol applied to a real wristband dataset, the pipeline collapses to chance. The evidence reviewed in the discussion attributes the gap to how differently four forearms can place the same tissue boundary, rather than to classical overfitting, and identifies the four-subject cohort as the binding constraint.

The pipeline shows that anatomically distinct forearm poses produce separable acoustic patterns even on a fully synthetic dataset and provides a reusable pipeline for evaluating new sensor layouts before any physical transducer has been built.

1 Introduction

Wristband-based gesture recognition supports interactions with assistive technologies [1]. Assistive technologies, per the World Health Organization (WHO), are all assistive systems and products that help individuals with disabilities, hence allowing their inclusion, participation and a better quality of life [2]. In order for wristband-based systems to support these individuals in their everyday activities, pose recognition is a key factor. It ensures a more natural and intuitive human-computer interaction (HCI) [3]. Wearable devices can support recognition of gestures by decoding neural and muscle activity without the need of external equipment or the user to be in a specific location, as the signal received by their inner sensors influences the response of the interface, however, large amounts of data are required to confirm what pose is executed [4].

To ensure an accurate recognition, an understanding of the anatomy of the upper limb or forearm is required. Focusing on the forearm and close to the wrist, the cross-section of this region can be divided into three different compartments: mobile wad, volar and dorsal compartment. While the volar and dorsal compartments are composed of muscles and neurovascular structures, the mobile wad only contains muscles [5]. As shown, each compartment encompasses anatomical differences, which will affect how sensors receive information.

For analyzing these anatomical differences, imaging techniques are key tools, where magnetic resonance imaging (MRI) stands out. MRI is a technique used for obtaining and processing medical images [6]. From a medical point of view, this technique allows to view and identify

the location of pathologies or abnormalities that might cause a problem [7]. The nature of the technique lies in its non-invasive testing , and that with the obtained data, it is possible to develop smart assistive technologies that will translate in prosthetics, rehabilitation protocols and HCI. However, on the other hand, this technique is expensive, limiting data acquisition [8]. Additionally, forearm data has variability across users [9].

As a consequence of these disadvantages, public forearm MRI datasets are scarce [8]. As researched by Dishner *et al.*, a small number of forearm datasets are publicly available, and those that are, tend to present limitations in terms of quality [10]. In a study conducted by Oakden-Rayner, L., there is an emphasis for performing quality inspections before publishing a dataset, due to a disconnection between the dataset development and the intended application [11]. The scarcity of datasets represents a major problem for researchers, as it restricts simulations and testing that could result in the development and scalability of emerging assistive technologies [12].

To tackle these challenges produced by the scarcity of forearm datasets, we propose a methodology for the generation of synthetic datasets, which will help in the design and fabrication of wristbands that are capable of enabling better assistive technologies that rely on pose recognition. *k-Wave* toolbox is used to generate synthetic ultrasound data using three different transducer geometries (Linear Array, Clustered Array and Smartwatch-Like), simulating a wristband transducer configuration on MRI forearm data. This results in pressure data that resembles real anatomical forearm structures.

Machine learning provides a data-driven framework for learning the relation between input signals and target labels provided by samples, instead of relying on manually defined rules. Deep learning is a subset of machine learning based on neural networks, where layers of trainable connections learn increasingly flexible representations from the data [13]. In this project, the generated ultrasound pressure traces are used to train different machine-learning models, including Logistic Regression (LogReg), Multilayer Perceptron (MLP), and a 1D Convolutional Neural Network (1D CNN), to identify and classify forearm static poses. This approach is motivated by previous work showing that wearable ultrasound can provide information about anatomical changes during static gestures and can be combined with machine-learning models for hand poses prediction. Following this approach, the simulated data can be used to compare recognition layouts and support the design of future wristband-based assistive technologies [14].

The report first introduces a *k-Wave* toolbox background. Additionally, a pipeline for obtaining the MRI dataset, the application of the *k-Wave* toolbox, and the classification of static poses using the Machine Learning algorithm is presented. Finally, results from the *k-Wave* and Machine Learning code are showcased, exhibiting the accuracy in pose classification.

1.1 *k-Wave*

k-Wave is a MATLAB-based numerical toolbox designed to simulate the propagation of acoustic waves in heterogeneous media [15]. It solves the acoustic wave equation using a computational grid, allowing the evolution of pressure fields to be tracked over time. This makes it useful for modeling ultrasound propagation in biological tissues, where material properties vary spatially [15, 16].

In realistic biological media, the speed of sound is not constant; it depends on the local mechanical properties of each tissue. *k-Wave* takes into account these spatial variables to define the velocity of the medium $c(\mathbf{x})$, allowing it to simulate into a more realistic and heterogeneous medium such as the forearm [16].

In order to numerically solve the coupled acoustic equations (Appendix C), the physical continuum needs to be discretized using a computational mesh. One of the most important

parameters in discretization is the grid resolution. As was explained before, the spectral Fourier collocation method requires sampling the field at least twice within one spatial wavelength according to the Nyquist criterion [16], for proper numerical simulation of linear wave propagation in a homogeneous medium.

When it comes to simulating a heterogeneous biological medium, such as the human forearm, things become quite complicated. In the interfaces between tissues, the speed of sound as well as the density of ambient air vary drastically. As the *k-Wave* manual states, applying the Nyquist minimum to the sampling of a heterogeneous medium leads to significant numerical dispersion errors [16].

As a consequence of numerical dispersion errors at tissue boundaries, an increased number of spatial sampling points per wavelength is required to accurately simulate the model. Grid spacing Δx, which should be uniform throughout the computational domain, is chosen to be inversely proportional to the maximal frequency of the excitation source $f_{\max}$ and the minimal velocity in the medium $c_{\min}$, corresponding to the slowest tissue. Mathematically, this relation can be expressed as:

$$\Delta x = \frac{c_{\min}}{f_{\max} \cdot \mathrm{PPW}} \tag{1}$$

Using the above expression, it is possible to calculate the sufficient spatial resolution needed to resolve all wavelengths in the system.

2 Methods

The proposed methodology follows a workflow of three main stages. First, forearm cross sections obtained from sliced MRI arm images are converted into discrete tissue maps. These maps are then used to define the acoustic medium in *k-Wave*. Finally, a set of classifiers is trained to recover the hand pose from the simulated ultrasound signals. Each stage is defined and implemented independently and is therefore discussed in a dedicated subsection. Detailed parameter tables, mathematical derivations, and additional figures are provided to the reader across all sections of the Appendix. The codebase is publicly accessible on GitHub at the following repository: https://github.com/gmimarsicano-cyber/Group-1b---Code.git.

2.1 Anatomical Data Acquisition and Segmentation

2.1.1 Source MRI Dataset

The data used was obtained from the public hand and forearm MRI dataset of Wang *et al.* [17], distributed as NIfTI volumes. The dataset includes MRI scans from four subjects performing twelve distinct hand and forearm poses. The scans were acquired with sub-millimeter in-plane resolution. They capture the internal anatomy of the hand and forearm, including muscles, bones, tendons and joints. To keep the classification problem tractable, six of the twelve available poses were selected for the downstream pipeline. The selected poses, namely poses 2, 5, 7, 8, 10 and 11, were chosen because they are visually distinct and therefore likely to correspond to anatomically distinguishable configurations. An overview of the selected poses is provided in Appendix B.1.

2.1.2 Slicing of MRI scans and Manual Segmentation

An initial set of 2D slices was extracted from the selected poses, resulting in a total of 68 slices. The slices were generated by sampling the source MRI volumes at different positions along the forearm. To obtain cross-sections approximately perpendicular to the forearm axis and located as close as possible to the wrist, the volumes were reoriented before slicing. The

number of slices extracted for each subject and pose and their details is reported in Appendix B.1.

When segmenting the images, accurate tissue identification is the crucial aspect. For this purpose, the software *ITK-SNAP* was used to manually segment each of the extracted slices [18]. Five anatomical classes were identified within the forearm region: bone, muscle, tendon, fat and joint. A sixth class, corresponding to a water-like background medium, was assigned to all unlabeled pixels, resulting in six classes in total for the ultrasound simulator.

As shown in Figure 1, segmentation was performed using the brush tool in the software, which allows to manually paint the different tissue regions and to label them with distinct colors. In regions where tissue boundaries were unclear or image quality was insufficient for reliable identification, ambiguous soft tissue was assigned to the muscle class, since muscle represented the dominant tissue type in most slices.

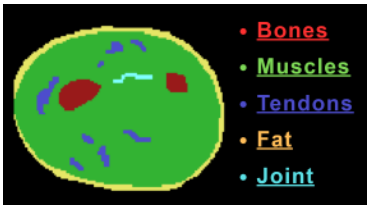

Figure 1: Example of a segmented image using *ITK-SNAP* software with the corresponding tissue labels.

In the final pipeline, the manually generated segmentations are used to define the simulator input. A *U-Net* model was also tested as an auxiliary segmentation approach. *U-Net* is able to automatically assign pixels to a specific anatomical class. Here, it was evaluated as a possible way to reduce manual segmentation effort. However, its predictions did not reach the anatomical accuracy required for reliable *k-Wave* simulations and were therefore excluded from the downstream pipeline. More information can be found in Appendix B.3.

To increase the dataset size without further manual annotation, the manually generated labels were instead expanded using MONAI-based data augmentation, as described in the following section.

2.1.3 Data Augmentation

To increase the segmentation dataset size without further manual annotation, the medical software MONAI was used [19]. The augmentation pipeline included random in-plane rotations of $\pm 17°$, random isotropic zooming, and 2D elastic deformation. Nearest-neighbor interpolation was used for all transformations to preserve the discrete integer label values and avoid the introduction of invalid or unlabeled pixels at tissue boundaries. The augmentation yields on the order of 100 augmented files per slice, a $\sim 20\times$ expansion that brings the simulator dataset to about 2000 samples per subject. Figure 2 shows an example of a manually segmented slice passed through the MONAI augmentation pipeline. Per-pixel class assignments are consistent as the same transformation happens to both the image and its label.

The final output of this stage was a structured dataset of integer-encoded 2D label maps, consisting of both the manual segmentations and their augmented variants. These label maps were then used as an input to the *k-Wave* pipeline after each tissue class was assigned acoustic physical properties.

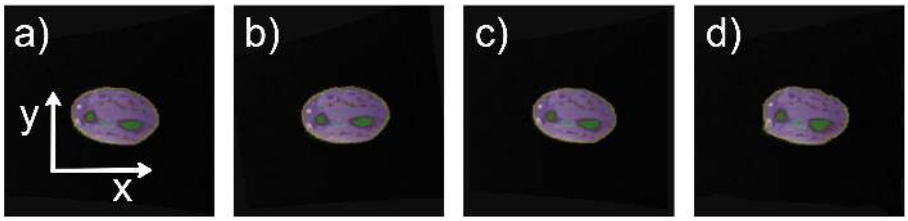

Figure 2: MONAI transformations applied to an MRI slice. a) Original slice, b) Random rotations of $\pm 17°$ on x and y axes, c) Random zoom, d) Random 2D elastic deformation.

2.2 Ultrasound Simulation with *k-Wave* Acoustic Toolbox

The following section presents and analyses the computational framework developed to generate synthetic ultrasound data (UD). Forearm labeled segmentations are preprocessed and converted into acoustic medium properties maps. Then, they are used as inputs to the *k-Wave* acoustic toolbox to simulate ultrasound wave propagation for three transducer array geometries: a linear (LA), a clustered (CA), and a smartwatch-like (SW). All configurations share the same anatomical preprocessing pipeline, acoustic medium definition, and computational grid design, while differing in transducer layout and number of transmit events.

2.2.1 Anatomical Preprocessing

Starting from the segmented forearm images obtained across different poses and subjects, the images are passed through a set of shared standardized preprocessing functions. This step reduces variability between inputs, so that differences in the simulated signals would only reflect changes in anatomical pose rather than inconsistencies in image orientation, size, or placement. The pipeline consists of four sequential steps:

- A canonical reorientation of the segmentation using PCA to align the principal anatomical axis parallel to the horizontal axis of the simulation grid [20].
- A bone-side correction to ensure that the bone regions are consistently positioned in the lower half of the simulation domain, furthest from the transducer.
- A rescaling of the forearm anatomy to a standardized width of 55 mm, representative dimension of an adult wrist [21].
- A placement of the resized anatomy within a fixed reference frame to maintain a constant spatial relationship between the wrist and the transducer across all segmentations.

A detailed description of each step is provided in Appendix C.2.

2.2.2 Tissue Parameter Mapping

Following preprocessing, each pixel within the forearm region was assigned acoustic properties according to its tissue class. Each of the six tissue classes was mapped to heterogeneous 2D maps of sound speed (c), density (ρ), and attenuation coefficient (α_0), which were used to construct the `kWaveMedium` object [16, 22].

Attenuation follows the power law $\alpha(f) = \alpha_0 f^y$ where $\alpha(f)$ is the attenuation in dB/cm, f is the frequency in MHz, α_0 is the attenuation coefficient at 1 MHz, and y is the power-law exponent. A common $y = 0.8$ was used for all tissues, as *k-Wave* defines `alpha_power` as a single scalar value [15]. A detailed discussion of the parameter assignment process and their values is provided in Appendix C [22].

The background outside the segmented anatomy was assigned water-like acoustic properties to represent the coupling medium and reduce artificial impedance mismatch at the segmentation

boundary [23, 24].

2.2.3 Grid Setup and Boundary Conditions

The grid spacing is set by the points-per-wavelength (PPW) criterion at PPW = 4 (Equation 1), giving $\Delta x \approx 73.75\,\mu$m based on a $c_{\min}$ equal to the fat-tissue sound speed of 1475 m/s [16]. A Perfectly Matched Layer (PML) is applied along all four edges of the simulation grid. The PML is an absorbing boundary layer that gradually attenuates outgoing pressure waves near the edges, reducing artificial reflections from the simulation boundaries. In addition, the simulation `kgrid` is sized dynamically to fit the forearm anatomy while satisfying PML and transducer clearance constraints. Specific calculations and details are given in Appendix C.4.

2.2.4 Time Array and Sampling Rate

The time step was determined from the CFL stability condition at CFL = 0.3, giving $\Delta t \approx 14.75$ ns and a corresponding sampling frequency of $f_s \approx 67.8$ MHz) [16]. This CFL value was chosen to ensure a time step sufficiently small for stable acoustic wave propagation. However, the resulting sampling frequency was well above the temporal Nyquist requirement limit for the 5 MHz ultrasound pulse used in the simulation. This produced oversampling and motivated the temporal downsampling applied before classification (Section 2.3).

2.2.5 Transducer Geometries and Pulse Design

Three transducers geometries are simulated, each backed by a dedicated top-level script (illustrated in Appendix F).

Original linear array: The original linear array consisted of 8 identical transducer elements arranged in a horizontal line above the forearm and distributed at equal intervals across the full wrist width. In between the forearm and the transducers a water-like coupling layer with the same acoustic properties as the rest of the background is enforced. [23, 24]. The array operated in full-matrix capture (FMC) mode: each element fired independently in sequence while all 8 elements recorded simultaneously [25]. The numerical parameters of this geometry, including the element size, transducer offset, and output transmit–receive tensor shape, are reported in Appendix C.

Clustered array: The clustered array was designed as an alternative to the uniform linear array. Instead of using single uniformly spaced elements, the array was divided into 8 clusters of 4 elements, each acting as a compact coherent sub-aperture. The inter-cluster spacing was adapted to the wrist width in each simulation in order to maximize the spatial coverage of the array, while still maintaining a minimum gap between neighboring clusters. With 32 elements in total operating in FMC mode, the clustered array produced 16 times more transmit–receive pairs than the linear array geometry. This denser sampling of the acoustic field was motivated by the hypothesis that, by yielding more data per simulation, an equal or higher performing model could be trained at a reduced overall computational cost. In practice, the per-fold cost of the clustered geometry did not drop below the linear-array baseline. Therefore, this geometry is included as a comparative configuration, rather than a computationally lighter substitute for the linear array.

Smartwatch geometry The smartwatch geometry was designed to mimic a wearable wristwatch-like configuration, where a larger transmitter (Tx) element insonifies a broader region of the forearm and four smaller receiving (Rx) elements are placed adjacent to it to trace the ultrasound echos. [26]. The numerical details of the Tx, Rx elements, elements offset, Tx/Rx

spacing, receiver pitch, jitter ranges, and output tensor shape are reported in Appendix C.5. As a data-augmentation strategy and to model small variations in the elements placement, random axial and lateral jitter were introduced. The jitter was applied to both the Tx and Rx elements together, preserving the internal geometry ratio. This geometry differs from the other simulated setups but is closer to a potential wearable configuration. Therefore, it was used to assess how well the machine-learning model could generalize to more application-oriented acquisition conditions. In this case, only one transmission event is performed, which substantially reduces the computational cost because only one solver call is required. However, this also removes synthetic-aperture information and angular diversity [27].

The source waveform was modified and improved during the development of the simulations. However, it is kept the same across the different simulation geometries.

The initial implementation used an idealized rectangular tone burst, which produced unrealistically sharp spectral edges that did not accurately represent the behavior of a real piezoelectric transducer. Therefore, the rectangular tone burst was replaced with a Gaussian-modulated pulse centred at $f_0 = 5\,\mathrm{MHz}$ with a 3-cycle envelope (temporal duration $\approx 0.6\,\mu\mathrm{s}$) [28]. A 3-cycle pulse was selected as a compromise between maintaining reasonable depth resolution and preserving adequate signal energy [28].

A Hann window apodization was applied spatially across each firing element before source injection [29, 30]. This gradually reduced the pressure amplitude from the center of the element towards its edges, avoiding abrupt aperture discontinuities. The apodization was introduced to reduce the artifacts produced by the abrupt spatial edges of the simulated transmitting element.

The source pressure amplitude was kept fixed at $S_{\mathrm{PA}} = 2 \times 10^5\,\mathrm{Pa}$ for all three geometries to ensure a consistent excitation level across simulations.

2.2.6 Simulation Execution

Wave propagation is solved on the GPU. Per-element pressure traces are reduced by row-averaging and an early-time cutoff. This removes the early direct Tx–Rx signal, so that the registered data mainly contain echoes back-scattered from the tissue.

Despite these optimizations, the clustered geometry remains the most computationally demanding configuration. In addition, the inherently larger number of simulated measurements generated does not result in a meaningful improvement in classifier performance.

For this reason, the report focuses primarily on the linear array configuration.

2.2.7 Signal Augmentation

Signal augmentation is applied at two stages. *Simulation-level* augmentation modifies the physical simulation parameters directly. The transducer placement is jittered laterally ($\pm 2\,\mathrm{px}$) and axially ($\pm 4\,\mathrm{px}$) per acquisition, and the global background sound speed is perturbed by $\pm 3\%$ to mimic acquisition variability. *Post-simulation* augmentation operates on the produced ultrasound traces. Additive Gaussian noise, small random time shifts, amplitude scaling ($\pm 15\%$), and bandpass filtering around the central frequency. As for the signal data augmentation case, this step is implemented to increase the dataset size without the need for time-consuming manual segmentations or additional simulation runs. The issues with post-simulation augmentation are further described in the 4 section.

The output of this stage consists of multi-channel ultrasound datasets generated separately for each transducer geometry and used as input to the machine-learning pipeline.

2.3 Machine Learning Classification Pipeline

After generating the synthetic ultrasound pressure time data with *k-Wave*, the next step was to test whether these signals contained enough pose-dependent information for classification. This is formulated as a supervised classification problem. During training, each model receives ultrasound traces with known pose labels and learns signal patterns that help separate the different poses. After training, the model is evaluated on unseen data to test whether these patterns generalize beyond the training examples.

The following subsections describe the preprocessing steps, the two baseline models used, the 1D CNN model, and the evaluation protocol used to perform the classification task.

2.3.1 Data Aggregation and Preprocessing

For each transducer geometry, the simulated ultrasound data were aggregated and processed. The Hilbert envelope was computed along the time axis, discarding the 5 MHz ultrasound data carrier so the classifier could see the envelope shape rather than the underlying oscillation. A short median filter suppressed impulsive numerical artifacts. The time axis was then downsampled by a factor of 50×–100× (axial sample spacing ≈ 0.55–1.1 mm), keeping the first-layer receptive field well below the forearm width and preserving CNN translation invariance. Finally, mean-variance normalization was applied either globally or per channel depending on the model.

2.3.2 Baseline Models: LogReg and MLP

Two baseline classifiers operated using flattened one-dimensional feature vectors, obtained by concatenating all channels and time samples:

- **LogReg**: a linear softmax classifier. This model does not explicitly account for spatial or temporal features, but provides a useful reference, meaning if classification of poses is already successful, a more complex model might not be needed.
- **MLP**: composed by two hidden layers of size $(256, 128)$ and ReLU activation functions. Compared with logistic regression, the MLP has greater representational capacity. However, because it treats the input as an unstructured vector, it is prone to memorizing subject-specific amplitudes.

The two baselines are useful precisely because they do *not* respect the time-series structure of the input. Comparing their performance with the convolutional model introduced below, therefore, helps assess whether pose information is mainly contained in global amplitude features or in the localized echo patterns.

2.3.3 1D CNN Architecture

The principal model is a custom 1D CNN designed to extract local temporal patterns from the multi-channel ultrasound signals. The convolutional filters operate along the time axis of each receiver channel. Each filter learns to respond to a specific local pattern in the signal trace. This is done so that a stack of convolutional layers progressively build up a description that the echoes features associates with each hand pose, independently of where exactly in time they occur. The architecture is a three-block design selected by an architecture search over seven candidates ranging from 12 k (the original two-block architecture) to 465 k parameters. The two-block architecture is later used as a comparison model to evaluate results. Details are provided in Appendix D.1. The final design has ~ 248 k trainable parameters.

The three convolution blocks use kernels of size 15, 9 and 5 respectively, with the channel count growing from 64 through 128 to 256, as illustrated in Figure 3. Wider kernels at the start let the first block see a broader segment of the trace and pick up the overall envelope shape.

Narrower kernels later refine more localized features, such as individual reflections. Between the blocks, a small pooling step halves the time resolution. At the end, an adaptive average pooling layer collapses the remaining time axis entirely. This last step is the most important one for cross-subject behavior: it forces the classifier to focus on the relative *shape* of the learned echo features rather than the absolute arrival time of each echo. The aforementioned is highly subject-dependent because different forearms have the same tissue boundary placed at different depths. Batch normalization stabilizes training across subjects with different tissue acoustics, and a dropout layer with rate 0.4 in front of the final linear classifier acts as the primary cross-subject regularizer.

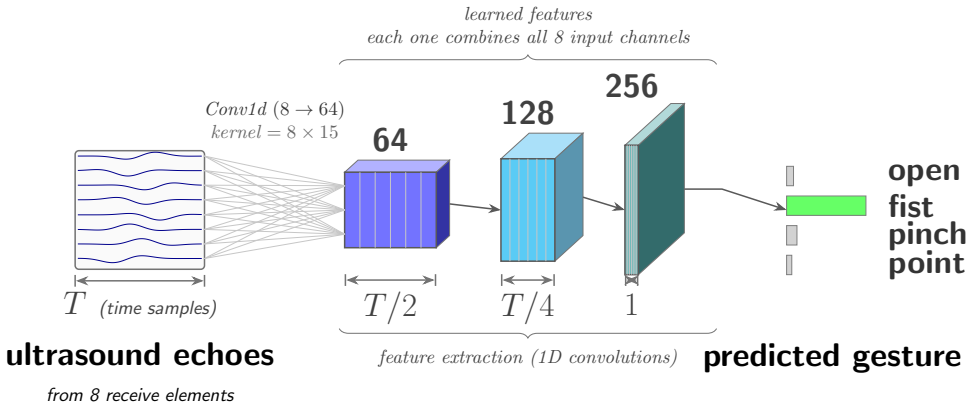

Figure 3: 1D CNN architecture overview. Three convolutional blocks ($64 \to 128 \to 256$ feature maps) followed by adaptive average pooling and a six-class softmax.

A recurrent alternative based on LSTMs was also implemented and evaluated, but ultimately discarded. It was an order of magnitude slower to train than the CNN at comparable or worse accuracy, exhibited the expected vanishing-gradient behavior in the long temporal range data, and offered no clear benefit on signals containing localized echo patterns, such as the one analyzed.

2.3.4 Evaluation Metrics

All models are evaluated with a Leave-One-Subject-Out (LOSO) testing method. At each fold, three subjects train (with an internal validation split for early stopping) and the fourth is held out as the true test set. Per-fold validation and test accuracy are reported alongside the LOSO mean. Validation accuracy is computed on samples from the *same* subjects as training, while test accuracy is computed on subjects unknown to the model. The gap between the two quantifies inter-subject domain shift directly. Where relevant, performance is compared against the chance baseline of $1/6 \approx 0.167$ for the six-pose problem. PyTorch models are trained with the Adam optimizer [31] (a standard adaptive-learning-rate gradient descent variant) at $\mathrm{lr} = 10^{-3}$, batch size 32, for up to 100 epochs with early stopping on the validation accuracy.

2.4 Use of AI in this project

Generative AI tools (*Claude Code* [32] and *ChatGPT* (GPT-5.3, Pro) [33]) were used for code collaboration, rewriting difficult-to-read phrases and improving the flow of the report text. All AI-generated code was reviewed, tested, and adapted by the authors before inclusion in the pipeline, and the authors take full responsibility for its correctness. GenAI was not used to design the experiments, interpret the results or write large sections of report text.

3 Results

Results are presented for each methodology stage: data augmentation, ultrasound simulations and machine learning classification. Their interpretation is deferred to the Discussion in Section 4.

3.1 Acoustic Simulations

The k-Wave simulations produced pressure–time data for each transmitter–receiver pair. After each transmit event, every receiver recorded a time-dependent pressure signal, which was then converted into a Hilbert envelope to visualize the received ultrasound energy over time. In 4, the envelope data are shown for one transmitter firing in the LA geometry, across all receive channels and all six gesture poses of Subject 1.

The simulated ultrasound data show structured, gesture-specific acoustic patterns. Each pose produced a characteristic distribution of signal energy across the receiver channels, visible as diagonal and localized high-amplitude regions in the time–receiver maps. These patterns arise because changes in wrist and hand pose modify the internal arrangement of tissues, changing the propagation, scattering, and reflection of the ultrasound waves. This receiver-dependent structure suggests that the transducer geometry captures spatial information about the underlying anatomy.

Across poses, differences are visible in both arrival time and signal amplitude. Some gestures show earlier or stronger envelope peaks, while others produce weaker or more spatially localized responses. This indicates that the simulated ultrasound data contain pose-dependent acoustic information rather than random noise. Similar shifts are also observed across subjects, reflecting anatomical variability in tissue geometry and transducer–forearm interaction.

These results show that the simulated ultrasound envelopes provide discriminative features that can be exploited by machine learning models for gesture-pose recognition.

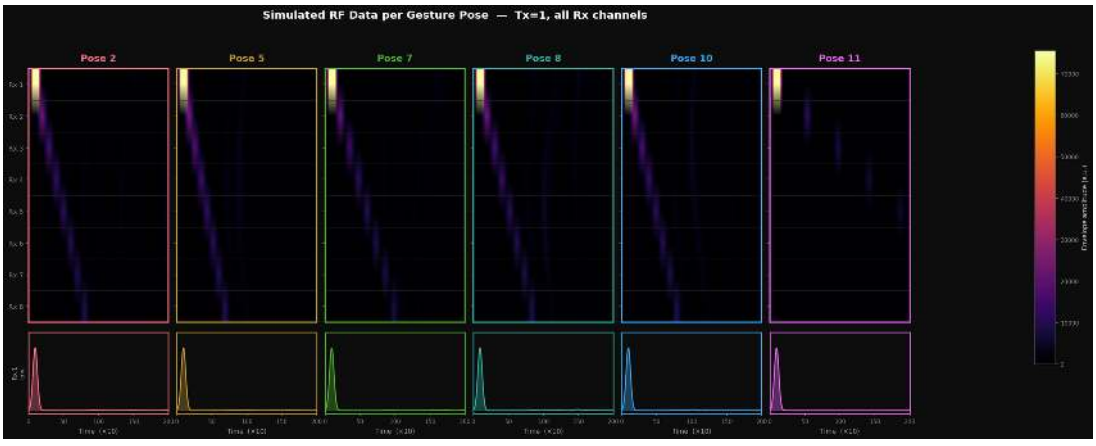

Figure 4: Hilbert envelopes per receive channel for a single transmitter firing on the linear array. All six poses of Subject 1.

3.2 Machine Learning Classification

All classifiers are evaluated under the four-fold LOSO protocol described in Section 2.3. The classification target is the six-pose problem, for which chance accuracy is $1/6 \approx 0.167$. The primary result of this work is the LOSO performance of the 1D CNN, where a three-block, $\sim$ 248 k-parameter network, reaches a mean validation accuracy of 0.717 and a mean test accuracy of 0.422. This shows a relative improvement of $\sim$ 49 % in test accuracy over the original two-block design (0.283).

1D CNN Per-subject LOSO. Figure 5 shows train, validation, test accuracy and mean test accuracy of 1D CNN for each held-out subject. Across the four folds, test accuracy ranges from 0.33 to 0.46, while validation stays within 0.65 to 0.75. The per-fold validation test gap therefore, sits at roughly 0.30 on average. Subject 3 is the lowest-performing fold across all simulated geometries and poses. In addition, Subject 3 Pose 2 shows a subject-dependent artefact in peak amplitude that does not appear in the other three subjects (Figure 6).

The per-fold confusion matrices (Appendix F, Figure 22) show that Poses 2, 10 and 11 are consistently recovered across all folds, while poses 5, 7 and 8 are frequently confused with one another.

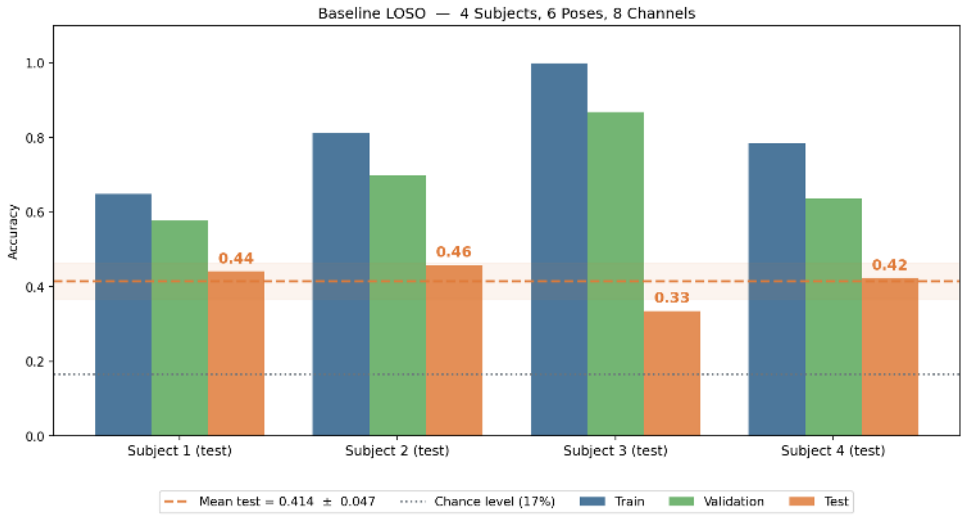

Figure 5: Train, validation, and test accuracy of 1D CNN per held-out subject.

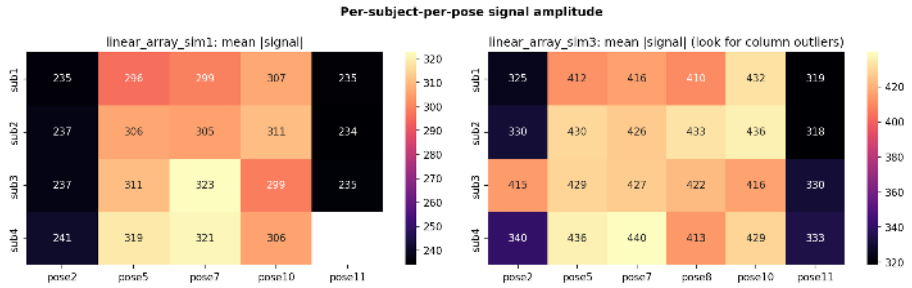

Figure 6: Peak-amplitude heatmaps per subject and pose on the linear array. Subject 3 shows a severe amplitude outlier in Pose 2 not present in the other subjects, which partly explains the lower test accuracy for that fold (Section 3.2).

The CNN is robust to aggressive temporal decimation (flat to 100×, dropping sharply beyond 200×), and the pose-specific accuracy structure is consistent with the confusion matrices. Both analyses are detailed in Appendix D.7.

Comparison against flat baselines. LogReg and MLP, on the same Hilbert-envelope features, fit the training set with a training accuracy of ≈ 1.0 on every fold. Under LOSO, however, their test accuracy drops into the 0.17 to 0.20 range, indistinguishable from chance and consistent with a complete failure to generalize across subjects. Figure 7 compares the LogReg and 1D CNN under three conditions: the six-pose baseline, the best individual simulation run, and the optimal channel/pose combination, using the results from the linear geometry. The three-way comparison, including the MLP, is shown in Appendix F, Figure 21.

Best four-pose subset. By restricting the label set to the four most anatomically distinct poses (2, 8, 10, 11), the LOSO mean test accuracy of both the LogReg and 1D CNN increase to ∼0.80 on the linear array (Figure 7, middle bar), against a chance baseline of 0.25. The pose-specific sources of the performance gap in the six-pose classification problem are reported in Appendix D.7. The implications are discussed in Sections 4.2 and 4.5.

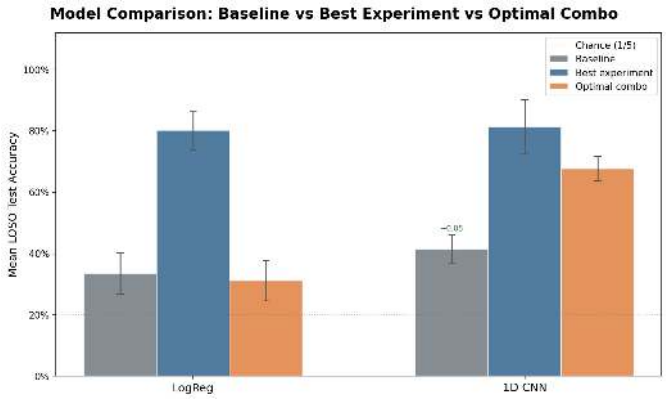

Figure 7: Comparison of LogReg and 1D CNN at three operating points: six-pose baseline, best individual experiment, and the optimal channel/pose combination for the linear array geometry.

Geometry comparison. The CA and SW geometries were trained and evaluated under the same protocol and 1D CNN architecture. The intended inherent data augmentation from the CA did not result in an improved classification score compared to the LA geometry, thus, results are not present in this work. The SW geometry falls behind the LA on every metric, with an overall-mean drop of ~ 0.07 in absolute test accuracy. A detailed comparison between the SW and LA geometries is shown in Appendix D.5, Figure 16,17.

To conclude, the 1D CNN is the only architecture in the comparison that produces non-trivial cross-subject test accuracy on the simulated data, with performance remaining stable across different temporal downsampling factors, pose-set selections, and transducer geometries.

4 Discussion

Interpretations of results is presented in this section. The mechanisms responsible for the observed performance differences across transducers geometries and model architectures, as well as their implications for the development of a functional wristband system are shown. The physical realism of the *k-Wave* setup including the wavelength-to-anatomy ratio, the role of tissue boundaries as reflectors, and the simplifications of the 2D solver are being addressed in Appendix C. The sections below focus on model behavior and inter-subject generalization.

4.1 Generalization of 1D CNN compared to Linear Models

Linear models (LogReg, MLP) flatten the $8 \times \sim 650$ input matrix, splitting the echo structure into independent scalars. This prohibits domain generalization by accurately scoring unseen subjects, since the location of the anatomical structures (Section 2.2.2) differs between subjects. Because the used linear models have more features than samples per fold, they fit the training set perfectly but fail entirely on the cross-subject test.

The 1D CNN behaves differently because of its adaptive average pooling layer. The temporal axis is collapsed to a single scalar per filter, so the classifier reads the shape of an acoustic echo burst and is insensitive to where that burst sits along the time axis. The features it produces are therefore robust to translation across subjects. Adding more simulated data to the 1D CNN

helps up to the three-block ($\sim$ 248 k-parameters) design, and then plateaus. The four-block variant ($\sim$ 465 k parameters) raises validation accuracy to 0.837 while LOSO test accuracy hardly moves, sitting at 0.418. This is the pattern of memorizing the three available training subjects, not of learning a better feature representation. A more detailed analysis is given in Appendix D, particularly sections D.1 and D.6.

4.2 Inter-Subject Generalization Gap

A validation–test gap of approximately 0.30 in absolute accuracy is one of the most stable observations across architectures, temporal downsampling factors, and pose subsets (Section 3.2). The obvious interpretation of such a gap is overfitting, where the network has too much capacity relative to the data and memorizes training noise. However, the evidence collected here suggests that the observed performance gap is mainly driven by inter-subject domain shift. With only four subjects, the three training subjects used for training may not span the anatomical and acoustic variability of the held-out subject such as forearm size and tissue distribution, making cross-subject generalization particularly challenging.

This interpretation is supported by the strong dependence of test accuracy on the identity of the held-out subject, with per-fold accuracy ranging from 0.33 and 0.46. Such variability is consistent with the cohort size being the limiting factor. Moreover, reducing model capacity to logistic regression limits its learning ability but does not eliminate the validation–test gap. Both flat models reach perfect training accuracy while test accuracy remains close to chance level, suggesting that they learn subject-specific patterns from the three training subjects rather than generalizable pose-related features. Similarly, increasing CNN capacity from two to four blocks improves validation accuracy, but does not improve LOSO test accuracy.

Label set and ceiling. As discussed in Section 3.2, LOSO mean test accuracy raises to $\sim$0.80 (Figure 7) when restricting the label set to a cohort composed of poses 2, 8, 10 and 11. The per-fold confusion matrices reported in Appendix F, Figure 22 show that this pose subset removes cross-subject confusions, as they are recovered more consistently on every held-out subject. The left out poses 5, 7 and 8, are the ones that are consistently misclassified between each other across folds and this explains the performance improvement.

These findings suggest that cross-subject classification is viable with a limited number of subjects as the training set, *when* the target label set is selected to account for anatomical differences. Therefore, this is suitable for a wearable that does not need specific pose recognition rather broader gesture categories.

Impact of the evaluation protocol. Cross-subject accuracy rely on the evaluation protocol it implicitly assumes. Figure 8 re-runs the same LogReg pipeline under two different splits, the LOSO consistently used in this paper and the Same-Subject Split used by Yang, X [14]. Under the same-subject split, with a training on the first 80 % of contiguous snapshots from a single (subject, session) and testing on the remaining 20 % of the same session, mean test accuracy reaches ≈ 0.731 across twelve folds. On the other hand, with a LOSO split applied to the same pipeline and the same data, mean test accuracy drops to ≈ 0.172 across four folds, indistinguishable from the 0.167 chance level. The same subject accuracy is not wrong, as it correctly measures within-user temporal generalization, resulting in a meaningful magnitude for a personalized and calibrated device. However, it does not support claims about cross-user generalization. The large gap between Same-Subject accuracy and LOSO accuracy provides an indication of how strong inter-subject domain shift is in practice.

Real-data validation. The same comparison can be done for 1D CNN. Figure 9 reports LOSO test accuracy for LogReg and for the three-block 1D CNN on the same real wrist-ultrasound

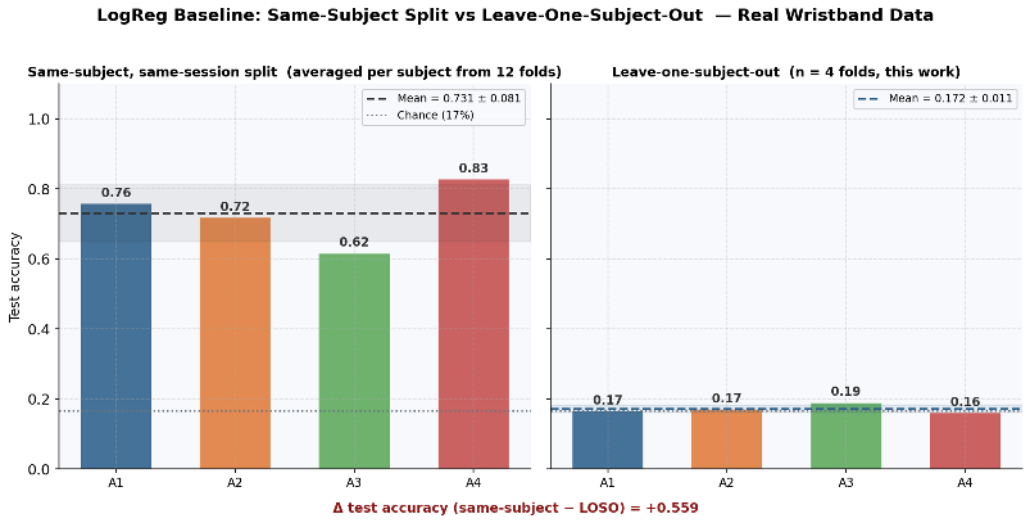

Figure 8: LogReg accuracy on real wrist-ultrasound data under same-subject splits (left, twelve folds) compared to LOSO splits (right, four folds). Each column corresponds to a different

dataset tested earlier, with one bar per held-out subject. Both models sit at chance on every fold from 0.17 to 0.18, with no meaningful separation between them. This result provides context for Section 3.2. The mean test accuracy for 1D CNN (0.422) is not inflated by the evaluation procedure, since the same protocol applied to the available real data puts the same network at chance probability. Moreover, the controlled acoustics of *k-Wave* make the cross-subject problem easier when compared to real life data, due to the simulator not modeling transducer-to-skin coupling variability, per-channel gain drift, motion artifacts, or the multi-path associated with real tissue. A data augmentation test applied after the simulation did not improve these results. As a result, the real data validation-test gap is wider than the synthetic one, suggesting the gap is driven by the cohort rather by the model. The same chance-level real data outcome in combination with the selected simplifications of the simulator (Section 4.4), indicates synthetic signal cannot be disregarded and there is an intrinsic difference with the real signal.

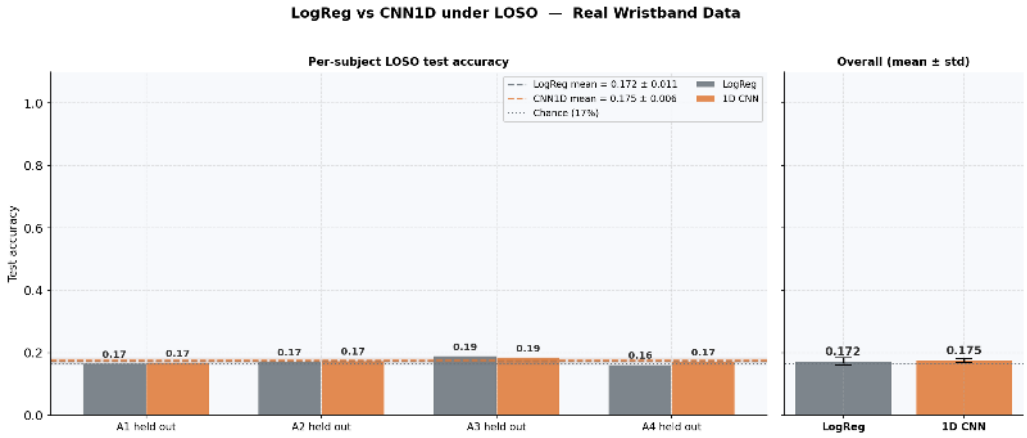

Figure 9: Per-subject LOSO test accuracy on real wrist-ultrasound data for LogReg and 1D CNN. Both models sit at the same level on every fold.

Context against the published state of the art. Parameter-efficient architectures for ultrasound-based human-machine interfaces, including UDACNN (77.7 %) and XceptionTime (75.4 %), are reported by Lykourinas *et al.* [34]. For a six-pose problem set, an evaluation on the

Ultra-Pro dataset under intra-session and same-subject splits is conducted. Therefore, they are comparable to the ≈ 0.73 same-subject LogReg in Figure 8. The validation scores are computed after training on the three-subject fold cohort, whereas the same-subject split trains and tests within a single subject/session. As a result, the difference in reported accuracies reflects the evaluation protocol used, rather than simply the modeling capacity of the classifier. To reduce the gap, more subjects are required rather than more advanced networks.

4.3 Transducer Geometry Design Comparison

Three geometries were evaluated: an 8-element LA, an 8×4 CA, and a SW configuration of one wide transmitter and four compact receivers. They were selected to span two design orientations that any wearable ultrasound HCI has to offer. Spatial sensoring favors an extended array, while a compact device favors a small one. The LA maximizes spatial resolution, whereas the SW configuration recreates compactness in a typical wristband configuration. The CA is used as a means to reduce the computational need of the simulation architecture by providing 16 times the data per simulation. Acoustically, it behaves like the LA, and it is treated as such.

The SW geometry underperforms on accuracy tests compared to the other two geometries, with an overall-mean drop of approximately 0.07 in absolute LOSO tests (Section 3.2). This is due to the use of a single wide transmitter and four small receivers, which provide one transmit firing per sample instead of eight, and only four receive signals per firing instead of eight, leading to an order of magnitude smaller for the feature arriving to the 1D CNN compared to the LA. Additional simulations on the SW geometry showed that scaling up the data volume helped Subjects 1 and 2 but saturated on Subjects 3 and 4. The limiting factor appears to be the anatomical complexity of each smartwatch sample acquisition. The practical implication for a wearable wrist-pose HCI that includes the SW geometry is to implement a per-subject model fine-tuning to increase the classification performance.

4.4 Current Limitations

- The MRI dataset comprises four subjects scanned for a prosthetics-oriented study, not for ultrasound simulation. The sub-millimeter in-plane resolution is sufficient, but the small subject number and the manual segmentation workload it imposes, are the primary constraints on dataset scale and, consequently, on cross-subject generalization. Further work is needed for predictive segmentations to be a valuable addition to synthetic dataset generation, as the current method yields unsatisfactory results as a data-generation input.
- The *k-Wave* setup rests on three deliberate simplifications. First, tissue-specific attenuation exponents was required to be a global scalar exponent ($y = 0.8$) and was applied to all tissue classes. Second, wave propagation was computed in 2D via the `kspaceFirstOrder2D` solver. However, the human forearm is a 3D cylindrical object, therefore, out-of-plane scattering is not captured in the current setup. Third, spatial resolution was reduced to PPW $= 4$ to keep computation manageable, at the expense of a less detailed representation of wave propagation. Translating the simulated setup to a physical prosthetic device introduces two challenges. The *k-Wave* model assumes a 6 mm water-like layer between the virtual transducer and the skin surface. In practice, achieving comparable acoustic coupling is extremely difficult, as the ultrasound gel layer used in real devices may evaporate, shift, or have air gaps. Furthermore, in a real prosthetic socket, skin and muscle are compressed under the mechanical load of a tight-fitting shell. This effect is absent from the simulation.
- The group studied in this work consists of four subjects. Each LOSO fold therefore holds out 25 % of the available anatomical manifold, and three training subjects cannot span a fourth held-out one.

- Physical augmentation was applied at acquisition time, not at minibatch time. The random transducer-placement jitter, sound-speed jitter and Gaussian-pulse/ Hann-apodization choices used in the simulator are drawn once and then frozen into the `.npz` archives, so the network sees a single realization per (subject, pose) acquisition instead of a distribution of realizations. This pattern incentivizes the 1D CNN to memorize the per-acquisition acoustic signature rather than to learn an invariance to it. Experimental evidence of this failure mode, in which physically motivated additions to the simulation *lower* LOSO test accuracy by ~ 0.09, is documented in Appendix E. Moving these augmentations to training time is the most actionable simulator-side fix and is taken up in Section 4.5.

4.5 Summary of results

The results in Section 3.2 show a successful synthetic ultrasound pipeline that produces datasets containing a valuable cross-subject classification score. The main LOSO accuracy of 0.422 for the chosen three-block, ~ 248 k-parameter 1D CNN sits well above the 0.167 chance line for the six-pose problem, and restricting the label set to the four anatomically distinct poses (2, 8, 10, 11) raises the same metric to $\sim$0.80. The 6-pose problem number remains below the same-subject accuracies reported in the literature for comparable parameter-efficient networks, while the 4-pose problem outperforms the literature, where the evaluation protocol does not separate the test subject from the training cohort.

The same architecture and pipeline collapse to chance under LOSO on real wristband data, and the same logistic-regression baseline reaches ≈ 0.73 on that data under a same-subject split. The persistent ~ 0.30 validation–test gap therefore seems to come from the limited anatomical spanning of the four-subject cohort, not from a deficit in model capacity. Any subset of three subjects does not span the fourth subject anatomically, with large subject-specific differences indicating some are more similar than others.

The work above points to multiple continuation paths. The largest gain would come from expanding the subject pool, either through new data acquisition or through much improved augmentation. A second promising path, which is simulator-side and cheaper to attempt, is to redraw the physical augmentations (transducer-placement jitter, sound-speed jitter, Gaussian pulse with Hann apodization) on every minibatch instead of freezing them at acquisition time. This converts the physical realism into a real regularizer without modifying the underlying physics or artificially inflating training and destroying test scores. A third direction is specific to the smartwatch geometry: adding an inductive bias toward lateral (channel-axis) translations would directly counter the geometry-specific deficit of that configuration. This appears imperative for any wearable design in which transducer placement varies across sessions.

5 Conclusion

This work developed an end-to-end synthetic-ultrasound pipeline that maps forearm MRI data to hand-pose classification through manual tissue segmentation, *k-Wave* acoustic simulation, and a 1D convolutional classifier. The pipeline was evaluated across three transducer geometries using a LOSO protocol to assess pose classification.

On the six-pose problem, the selected three-block, ~ 248 k-parameter 1D CNN reaches a LOSO mean test accuracy of 0.422 on the linear array, significantly above the 0.167 chance baseline, while the smartwatch geometry performs ~ 0.07 lower. The most positive finding is on the four anatomically distinct poses 2, 8, 10, 11. Under the same protocol and the same network, LOSO test accuracy reaches $\sim$0.80 against a 0.25 chance baseline.

When applied to a real wristband dataset, the same pipeline remained at chance level on

every LOSO fold. Together with the residual validation–test gap of approximately ~ 0.30 on the synthetic data, this indicates that the current limitation is not primarily model capacity, but the limited anatomical coverage of the four-subject MRI cohort. The main constraint on the present study is, therefore, the small number of available subjects, which limits the strength of the conclusions that can be drawn about cross-subject generalization.

Overall, the results indicate that MRI-derived synthetic ultrasound can provide a useful signal for pose classification, but that broader anatomical coverage is required before the approach can be considered robust for subject-independent wristbands gesture recognition.

Acknowledgements

The authors would like to thank our supervisors Victor Chuman Alvarado and Andrea El Haddad for helping us during the development of the project. Also, special thanks to Chinmay Pendse, for providing a baseline LogReg machine learning model for training and identifying the different gestures used in the project.

The authors also thank Bohan Wang, George Matcuk and Jernej Barbic for the forearm dataset, and the *ITK-SNAP* and *k-Wave* development teams for the segmentation software and simulation toolbox.

Conflict of interest

The authors declare no conflict of interest. No benefits have been received or will be received related to the delivery of this report.

APPENDIX

A Physics of ultrasound waves

A.1 Nature of the wave and propagation

Unlike electromagnetic waves, acoustic waves (in this case, ultrasound) in biological soft tissues propagate as longitudinal waves, where the displacement of the medium's particles is parallel to the direction of energy transport. These mechanical waves create periodic compression (higher density) and rarefaction (lower density) of the particles [35, 36]. The intensity of the wave is proportional to the square of its amplitude [36].

The relationship between velocity v, frequency f, and wavelength λ is given by:

$$v = f\lambda \tag{2}$$

In this case, for the forearm anatomy, the average speed of sound in soft tissue is usually assumed to be $v \approx 1540\,\text{m/s}$ [37]. High-frequency ultrasound waves offer better resolution, but they suffer from greater attenuation; thus, 5 MHz is often identified as a suitable frequency for wrist/forearm applications to balance depth and signal-to-noise ratio [37].

A.2 Acoustic impedance

Acoustic impedance is the physical property of a medium that determines the reflection of sound waves at a certain boundary [35]. It is defined as:

$$Z = \rho v \tag{3}$$

where v is the phase velocity and ρ the mass density of the tissue [36]. At boundaries between tissues with different impedances (such as bone and muscle), a significant fraction of the wave's energy is reflected back to the sensor [37, 35].

A.3 Wave-tissue interaction

When the ultrasound wave enters the forearm, its behavior is governed by two main phenomena [35]:

1. **Transmission and reflection:** When the ultrasound wave encounters a boundary between two media with different impedance (Z_1 and Z_2), part of the energy is reflected, and these reflections (called echoes as well) are captured by the wristband to identify the anatomical changes [37]. The remaining energy will be transmitted into the deeper tissue and partially absorbed. For normal incidence, the Intensity Reflection Coefficient R is [36]:

$$R = \left(\frac{Z_2 - Z_1}{Z_2 + Z_1}\right)^2 \tag{4}$$

2. **Attenuation α:** As the ultrasound wave propagates through the forearm, its intensity I will decrease exponentially with the distance x as follows [36]:

$$I(x) = I_0 e^{(-2\alpha x)} \tag{5}$$

   The attenuation coefficient α depends on the tissue type, and it typically increases approximately linearly with frequency [16].

In addition to reflection and attenuation, ultrasound waves may also undergo refraction and scattering when propagating through biological tissues. Refraction occurs when the wave changes direction as it passes between media with different propagation speeds, while scattering arises from small-scale inhomogeneities within the tissue, causing the wave to spread in multiple

directions [35]. These phenomena contribute to the complexity of ultrasound wave propagation in realistic biological environments.

Wave propagation is described by the acoustic wave equation, which captures how pressure evolves in space and time as a function of the medium's mechanical properties [16]:

$$\nabla^2 p - \frac{1}{c^2}\frac{\partial^2 p}{\partial t^2} = 0 \tag{6}$$

where p represents the acoustic pressure and c is the speed of sound in the medium [16].

Spatial pressure variations drive temporal changes, propagating pressure waves through the medium. The speed of sound (set by the mechanical properties of each tissue) controls how fast those disturbances travel and varies between tissue types [16]. This equation is the starting point for numerical solvers such as *k-Wave*.

B Segmentation Details

B.1 Slicing and segmentation images

The six selected poses are shown in Figure 10, and the per-subject slice counts breaking down the 68-slice total are given in Table 1; pose 2 has slightly fewer slices in two subjects due to MRI artifacts and ambiguous tissue boundaries near the wrist joint.

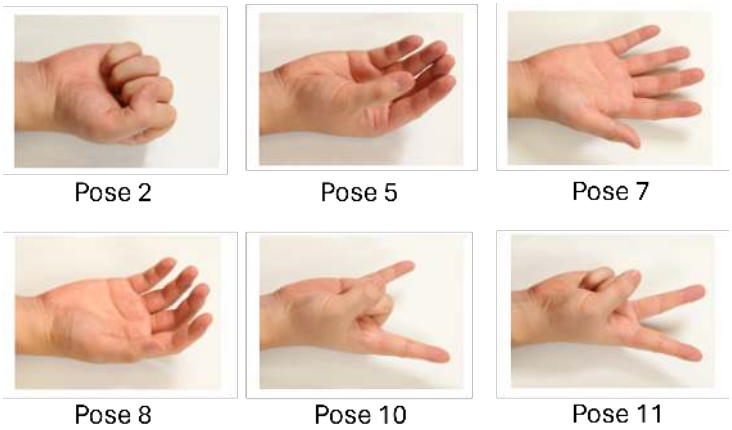

Figure 10: Hand and forearm poses retrieved from MRI dataset [17].

Table 1: Number of slices per subject and pose.

		Pose					
		2	**5**	**7**	**8**	**10**	**11**
Subject	**1**	3	3	3	2	3	3
	2	3	3	3	3	3	3
	3	2	3	3	3	3	3
	4	2	3	3	3	3	2
Total		10	12	12	11	12	11

B.2 Per-fold cross-validation

Per-subject DSC scores under LOSO and the corresponding training-loss curves are shown in Figures 11 and 12. Scores vary noticeably between folds, mirroring the cross-subject variance seen in the downstream classifier.

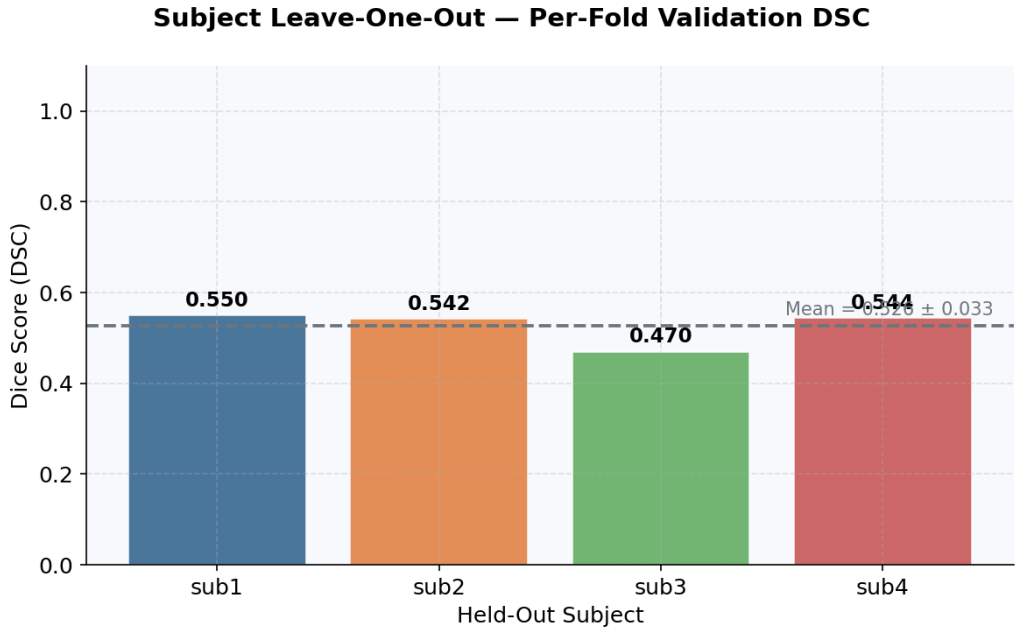

Figure 11: Per-subject mean DSC under LOSO cross-validation.

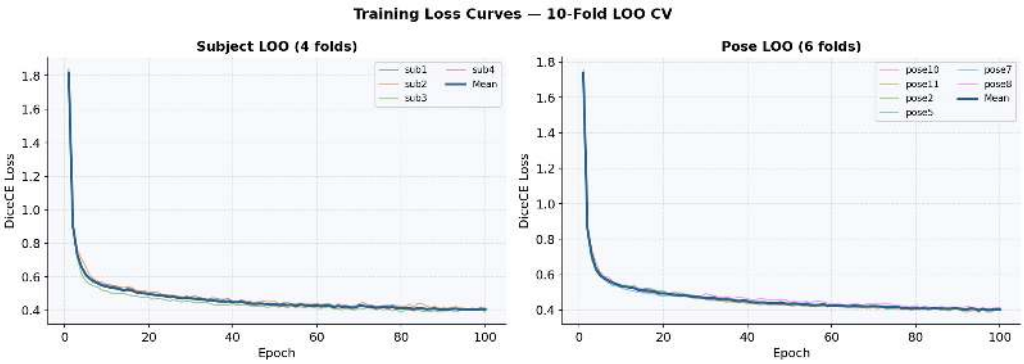

Figure 12: Per-fold training loss curves for the four LOSO splits.

Figure 13 shows a qualitative LOSO prediction for Subject 1. Bone, muscle, and the water-like background are recovered more consistently, while tendon and joint/cartilage are often confused with nearby tissues. These two classes also have fewer labelled pixels, which makes them harder to predict. This confusion creates noisy borders and makes the predicted segmentations unsuitable for the acoustic simulation.

B.3 U-Net Segmentation Experiment and Failure Modes

To explore whether the manual segmentation process could be scaled automatically, a U-Net based on `segmentation_models_pytorch` [38] with a ResNet-34 backbone [39] was trained to predict tissue label maps from the MRI slices. The encoder reduces the image resolution to extract contextual features, while the decoder restores the original resolution. Skip connections help preserve tissue boundaries, which are important for the acoustic simulation. The output is an integer label map of shape $H \times W$ with six numerical classes per pixel: five anatomical tissue classes and one background or coupling-medium class used by the simulator.

The model was evaluated using the same LOSO protocol as the downstream classifier. In each fold, three subjects were used for training, and the remaining subject was used for test-

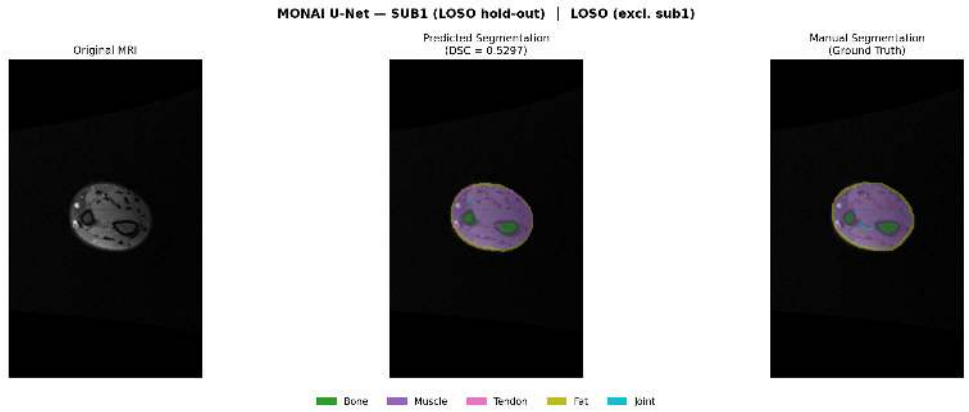

Figure 13: Qualitative LOSO prediction for Subject 1: input, ground truth, and prediction.

ing. Training used a combined Dice and cross-entropy loss, with early stopping based on the validation Dice Similarity Coefficient (DSC).

After several training attempts with increasing augmentation budgets, the best mean DSC across all U-Net configurations was approximately 0.62 (Figure 14). A threshold of around 0.70 was defined as the minimum quality needed for physically meaningful *k-Wave* simulations. Below this level, predicted tissue boundaries, especially bone borders, varied by several pixels between slices and could change the simulated reflection patterns. For this reason, U-Net predictions were not used in the final *k-Wave* pipeline. All simulated ultrasound data were generated from the manual segmentations and their MONAI-augmented versions.

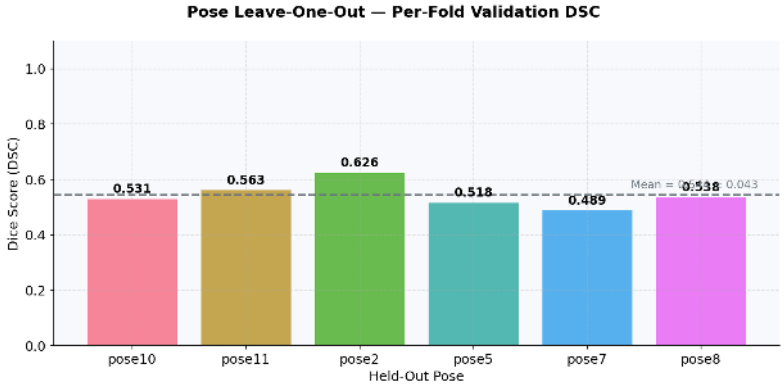

Figure 14: Per-pose DSC under LOSO cross-validation.

C *k-Wave* Simulation Details

C.1 Mathematical Formulation of *k-Wave*

This software, instead of solving this second-order wave equation directly, as presented in Appendix A, models the acoustic field by solving a system of coupled first-order partial differential equations based on the conservation of momentum, mass, and the pressure-density relation [16]:

$$\frac{\partial \mathbf{u}}{\partial t} = -\frac{1}{\rho_0}\nabla p \quad \text{(momentum conservation)} \tag{7}$$

$$\frac{\partial \rho}{\partial t} = -\rho_0 \nabla \cdot \mathbf{u} \quad \text{(mass conservation)} \tag{8}$$

$$p = c_0^2 \rho \quad \text{(pressure-density relation)} \tag{9}$$

Where $\mathbf{u}$ is the acoustic particle velocity, ρ is the acoustic density, and ρ_0 is the ambient density. According to the *k-Wave* User Manual, there are three principal advantages when using the first-order instead of second-order mathematical formulation of the wave equation [16]:

1. Makes easier the inclusion of both mass and force source terms into the discrete equations.
2. Improves numerical accuracy by making the pressure and the particle's velocity be computed on dispersed spatial grids.
3. Enables the implementation of a Perfectly Matched Layer (PML) to anisotropically absorb acoustic waves when they reach the edges of the computational domain.

To solve the acoustic equations efficiently and save computer memory, *k-Wave* uses a k-space pseudo-spectral method instead of traditional finite-difference methods. Traditional methods calculate how the wave changes by looking only at neighboring points in a local approach. On the other hand, *k-Wave* calculates these changes across the entire grid at once in a global way by fitting a series of Fourier sine waves to the data [16]. Because it uses these continuous waves, the spatial sampling follows the Nyquist limit. This means the software theoretically only needs two grid points per wavelength to accurately represent the wave, making the simulation much faster and lighter [16].

C.2 Anatomical preprocessing

The pipeline runs four sequential steps on each segmentation. (1) **Canonical reorientation**: PCA aligns the principal anatomical axis horizontally (`register_labels_to_canonical_axis`) [20]. (2) **Bone-side correction**: the bone centroid is checked to sit in the lower half of the image, furthest from the ; if not, a 180° rotation is applied (`enforce_bones_in_lower_half`). This matters because bone is the dominant acoustic reflector, so any inconsistency in its position relative to the transducer array would introduce geometry-driven signal differences [22, 40]. (3) **Size standardization**: the forearm section is rescaled to a fixed width of 55 mm, representative of an adult forearm section (`rescale_labels_to_target_size`) [21]. (4) **Reference placement**: the resized anatomy is centered within a fixed canvas so the forea distance is consistent across segmentations (`place_labels_in_reference_frame`). Nearest-neighbour interpolation is used throughout to preserve integer tissue labels [41].

C.3 Tissue acoustic properties

Acoustic properties were drawn from the compilation by Goss et al. [22]. Where multiple measurements existed for the same tissue, values were selected for healthy, fresh or *in vivo* preparations at frequencies close to 5 MHz. In addition, comparable mammalian data were used when human values were unavailable. Attenuation follows the power law $\alpha(f) = \alpha_0 f^y$ [15, 16]; tissue-specific exponents were estimated from the literature but a single global value $y = 0.8$ was used in the simulation for simplicity. The estimated values are listed for completeness in Table 2.

C.4 Simulation Grid Sizing: Implementation Notes

Two earlier implementations were discarded. A fixed 320×320-pixel canvas clipped some forearm anatomies. A subsequent 3200×3200 canvas was introduced alongside an unresolved bug that set the rescaling target to 5.5 mm instead of 55 mm (see Section 3.1), inflating the grid area by two orders of magnitude and making computation prohibitive. In the corrected implementation, the grid dimensions are determined automatically from the forearm bounding box, and all necessary clearance margins are added automatically.

Class	Tissue	c (m/s)	ρ (kg/m^3)	α_0 (dB/cm/MHz^y)	Estimated y	Simulation y
0	Water-like medium	1500	1000	0.00	1.00	0.80
1	Bone	3300	1970	12.00	1.50	0.80
2	Muscle	1580	1070	1.10	0.95	0.80
3	Tendon	1750	1100	4.70	0.79	0.80
4	Fat	1475	937	0.61	0.84	0.80
5	Joint/cartilage	1665	1100	5.00	0.83	0.80

Table 2: Acoustic properties assigned to the six tissue classes. The estimated y values were obtained from the reference literature during parameter selection, whereas the simulation y column reports the scalar value actually used in *k-Wave* [22].

C.5 Transducer geometry parameters

The numerical settings of the three simulation scripts are summarized in Table 3. These parameters define the element dimensions, transducer placement, and output tensor structure used in each simulation script.

Parameter	Linear	Clustered	Smartwatch
Number of transmitters / receivers	8 / 8	32 / 32	1 / 4
Total number of elements	8	32	5
Element width × height	5 × 1 px	5 × 1 px	Tx: 25 × 1 px; Rx: 13 × 1 px
Transducer offset (px)	30	30	30
Output tensor shape	$(8, 8, N_t)$	$(32, 32, N_t)$	$(1, 4, N_t)$

Table 3: Per-geometry transducer parameters used in the three simulation configurations.

C.6 Physical Realism of the Simulation

At $f_0 = 5\,\text{MHz}$ in soft tissue ($\lambda \approx 0.3\,\text{mm}$), a 55 mm forearm spans roughly 180 wavelengths. Internal tissue boundaries are therefore large relative to the wavelength and act as strong reflectors whose echo timing and amplitude change with forearm pose, and this is exactly the signal the classifier learns to distinguish. The remaining simplifications (2D wave propagation, PPW = 4, a single global attenuation exponent $y = 0.8$, and an idealized water coupling layer) are deliberate concessions to the available compute budget and are treated as limitations in Section 4.4.

D Machine Learning Details

Supporting numbers for Section 3.3: the full architecture-search table, per-fold baseline accuracies, a note on why the early dataset inflated scores, and the training curves of the chosen CNN.

D.1 Architecture search

The full set of CNN designs evaluated is listed in Table 4. Each row is a complete LOSO mean over the four subjects on the linear-array dataset. The chosen design is the bolded three-block model, which scores best on the held-out test fold.

Across the search, additional depth was more useful than additional width. The two-block variants barely move regardless of BatchNorm or channel count, while the three-block design produces the largest gain in test accuracy. Larger models (four-block, or three-block with a temporal pool) raise validation accuracy further but do not lift the test set, a pattern consistent with the network memorizing training-subject anatomy without learning transferable features.

Architecture	Params	Val acc	Test acc
Original two-block ($k = 7, 5$; $32 \to 64$)	12 k	0.562	0.283
+BatchNorm +Dropout only	13 k	0.553	0.317
two-block wider ($64 \to 128$)	46 k	0.545	0.303
three-block ($64 \to 128 \to 256$, $k = 15/9/5$)	**248 k**	**0.717**	**0.422**
4-block ($k = 21/11/5/3$)	465 k	0.837	0.418
three-block + 4-bin temporal pool	253 k	0.855	0.390
three-block + 2-layer FC head	280 k	0.795	0.407

Table 4: Full LOSO architecture search on the linear-array dataset (6 poses, 4 subjects).

D.2 Per-fold baseline accuracies

Table 5 reports per-fold logistic-regression scores on both linear-array datasets. The mean drops between the early and the balanced version, but as discussed below, that drop is largely structural rather than a regression in simulation quality. Subject 3 consistently scores lowest across both datasets, partly explained by a severe amplitude outlier in pose 2 absent in the other subjects (Figure 6).

Held-out subject	early dataset (5 cls.)	balanced dataset (6 cls.)
sub 1	0.558	0.298
sub 2	0.607	0.378
sub 3	0.322	0.245
sub 4	0.617	0.417
Mean	0.526 ± 0.120	0.334 ± 0.067

Table 5: Per-fold logistic-regression LOSO accuracies on the two linear-array datasets.

D.3 Why the early dataset scored higher

The two datasets are not directly comparable. The early version was unevenly sampled: one subject had no scans for a pose, and pose 10 had only six scans per subject against 20–40 in the balanced version. Under LOSO, missing classes artificially inflate the test scores on certain folds and reduce the cost of errors on sparse poses.

D.4 Training curves

Figure 15 shows training and validation curves for the chosen three-block CNN on the linear-array dataset. Both settle within the first ~ 30 epochs and remain stable.

D.5 Smartwatch vs. Linear Array

Figures 16 and 17 report per-fold model accuracies for both geometries on a five-pose subset (chance baseline 1/5); the geometry-level interpretation is given in Section 4.3.

These plots correspond to a five-pose simulation setting, so the chance baseline is 1/5. The CNN is the best classifier on both geometries and the same baseline ordering holds. The smartwatch CNN falls behind on every metric compared to the linear array. The single wide transmitter spreads energy more diffusely across the forearm, giving each recorded trace less spatial detail than a focused linear-array element provides. With less discriminative information per trace, the classifier has a harder time learning pose-specific patterns that generalize to unseen subjects.

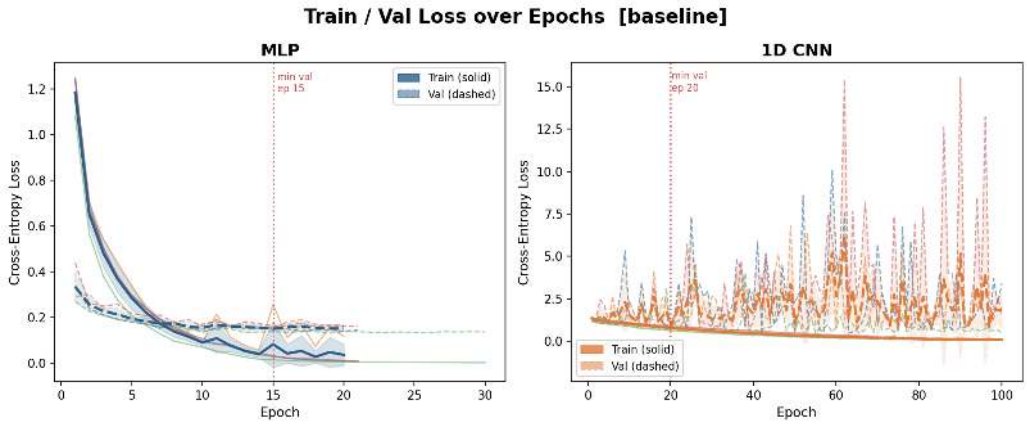

Figure 15: Training and validation loss/accuracy curves for the chosen three-block CNN.

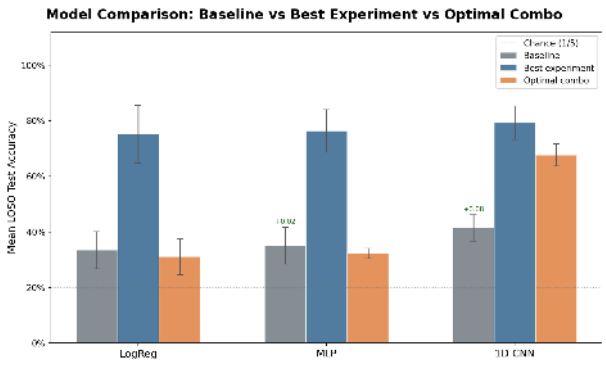

Figure 16: Model comparison for the linear-array geometry.

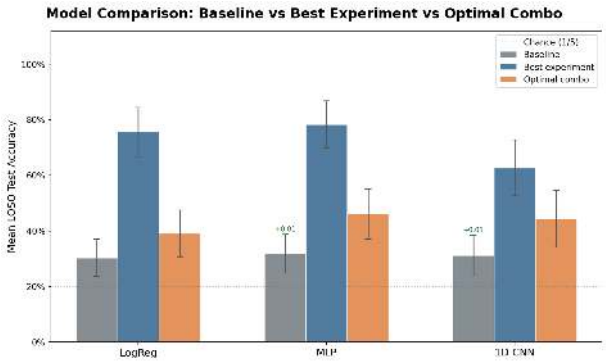

Figure 17: Model comparison for the smartwatch geometry.

D.6 Additional explanation of 1D CNN generalization

The input to the classifier is an $8\times \sim 650$ tensor: eight receiver channels over time. Logistic regression and MLP both flatten this into roughly 5 000 scalar amplitudes before learning, dissolving the per-channel echo structure. With more features than training samples per fold, these models overfit to the training subjects' amplitude statistics and fail on a new anatomy — producing the train = 1.0 / test $\approx$ chance pattern visible for both baselines in Figure 21.

The 1D CNN avoids this by sliding small filters along the time axis, detecting local echo patterns regardless of where in time they appear. The adaptive average pooling layer at the end of the network reinforces this: it collapses the time axis entirely so the classifier sees the *shape* of the echo pattern rather than its absolute arrival time, which varies across subjects as different anatomies place the same tissue boundary at different depths.

D.7 Supporting Analyses: Temporal Decimation and Pose-Class Structure

Temporal decimation. Figure 18 shows test accuracy as a function of decimation factor. The CNN stays flat up to 100×, then drops sharply between 100× and 200× (from 0.356 to 0.272), and approaches the chance baseline at 400×; logistic regression increases up to 50× before dropping. Discriminative features collapse beyond $\sim$ 100×, at which point test accuracy drops to chance.

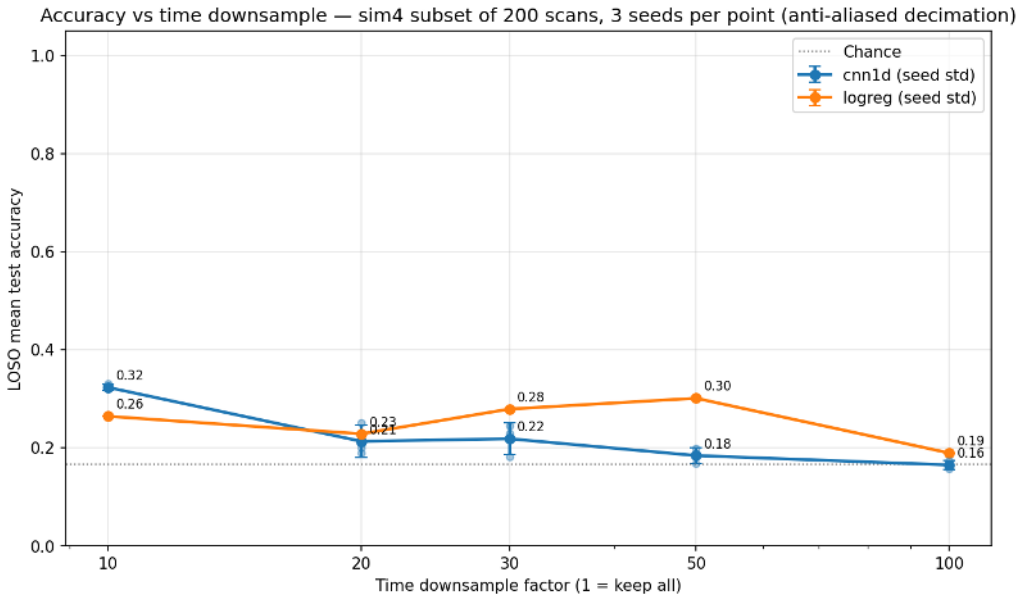

Figure 18: Test accuracy versus temporal decimation factor for the 1D CNN and the logistic-regression baseline.

Pose-class structure. Figure 19 shows a leave-one-pose-out sweep (left) and ranked four-pose subsets (right). Both confirm the same structure: removing pose 2, 10 or 11 causes the largest accuracy drops, while removing pose 5, 7 or 8 barely changes performance. The top four-pose subsets consistently combine poses from (2, 8, 10, 11). The confusion between poses is further detailed in Figure 22.

E Simulation-Realism Enhancements

This appendix documents a side experiment in which three physics-motivated additions to the *k-Wave* simulator, intended to improve acoustic realism and act as a regularizer for the

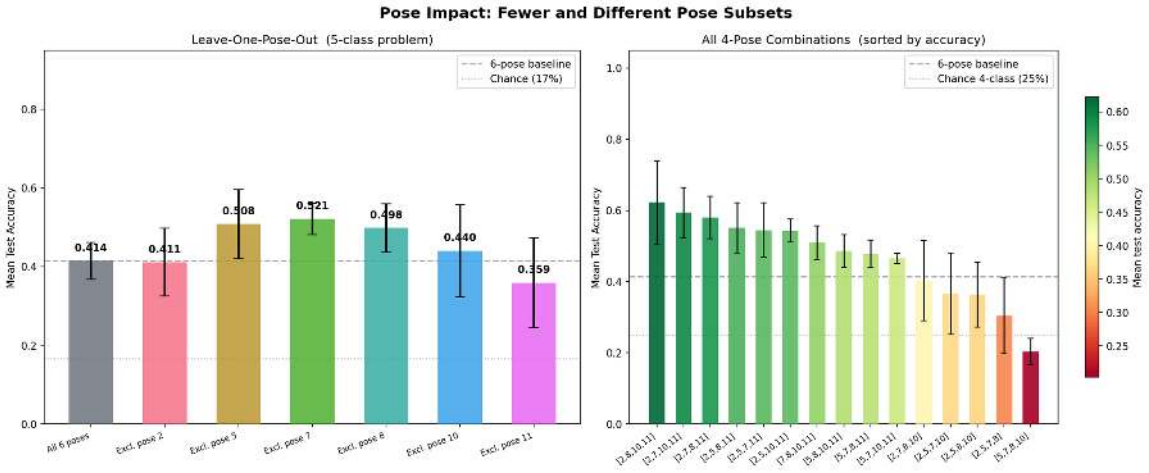

Figure 19: Leave-one-pose-out accuracy (left) and ranked four-pose subsets (right) for the 1D CNN.

downstream CNN, were negative for cross-subject generalization: LOSO test accuracy fell from 0.414 ± 0.047 to 0.327 ± 0.047 on the same dataset and protocol. The diagnosis is reproduced here because the mechanism is informative for further continuation of the project (see Section 4.4).

E.1 Hypothesis and physical motivation

Three changes were applied simultaneously to the simulator described in Appendix C, each intended to make the simulated signals harder to memorize.

- **Sound-speed jitter.** The bulk soft-tissue sound speed was perturbed per acquisition, $c \leftarrow c \cdot (1 + \delta)$ with $\delta \sim \mathcal{U}(-0.03, +0.03)$, to model the ± 1–$3\,\%$ variability in tissue density and temperature seen across individuals and sessions.
- **Transducer-placement jitter.** The transducer was shifted axially (± 4 pixels) and laterally (± 2 pixels) per acquisition to model imperfect transducer contact and small positional offsets, as would occur in a real wearable.
- **Gaussian-modulated tone burst with Hann apodization.** The rectangular excitation was replaced with

$$p(t) = \exp\left(-\frac{t^2}{2\sigma^2}\right) \sin(2\pi f t), \qquad \sigma = \frac{N_{\text{cyc}}}{6\,f},$$

combined with a per-pixel Hann window across each element. Both changes are standard practice to suppress frequency sidelobes and bring the excitation closer to a real transducer.

E.2 Observed behavior

The same three-block CNN1D and the same LOSO protocol were applied to the baseline and updated simulators. Table 6 reports the per-fold ranges and the means. Every fold moved in the same direction: training accuracy rose, validation accuracy fell mildly, test accuracy fell on every subject, and both the train–test and validation–test gaps widened. The mean test drop of ~ 0.09 is therefore consistent across folds and is not driven by a single subject held out from the fold.

Quantity	Baseline	Updated	Change
Mean LOSO test accuracy	0.414 ± 0.047	0.327 ± 0.047	$\downarrow\sim 0.09$
Per-fold train accuracy	0.65–1.00	0.91–0.99	tightened, raised
Per-fold validation accuracy	0.58–0.87	0.62–0.78	mildly lower
Per-fold test accuracy	0.33–0.46	0.28–0.37	lower on every fold
Validation–test gap	~ 0.20	~ 0.30	widened
Train–test gap	~ 0.40	~ 0.60	widened

Table 6: Baseline versus updated configuration on the same LOSO protocol and the same dataset. The updated configuration introduces sound-speed jitter, transducer-placement jitter, and the Gaussian-pulse / Hann-apodization excitation simultaneously.

E.3 Why training accuracy rose and test accuracy fell

Frozen per-acquisition jitter. A proper augmentation draws fresh perturbations on every step, which forces the model to learn invariance to them. With this configuration, each jitter value is drawn once and then frozen into the combined pose archive, so all Tx firings within a fold share the same jitter. The classifier thus sees an acoustic signature that is correlated with the pose label of that acquisition. With roughly 144 effective training samples and $\sim 248\,\mathrm{k}$ parameters, memorizing those signatures is a cheaper way to lower the loss than learning pose-anatomy invariance, resulting in a first data quality reduction.

Excitation bandwidth and trace content. The Gaussian-modulated tone burst combined with the Hann apodization narrows the frequency bandwidth of the excitation and suppresses angular sidelobes. The training loss surface becomes smoother, and the network converges faster. However, the resulting traces also carry less spatial detail: closely spaced reflectors are harder to resolve, and off-axis tissue contributes less to each recording. The classifier therefore has less information to work with, even though it fits the training set more easily. This is not a configuration or code error, but it does result in a decrease in information content in the data for the model to train on.

Lateral jitter and the first convolutional layer. In the `Conv1d` architecture used here, the first convolutional layer binds each receive element to a fixed channel slot. A lateral shift of ± 2 pixels corresponds to roughly $40\,\%$ of one element's footprint, which is enough to move the underlying tissue from one channel slot to a neighboring one. Within the training set, the model can memorize the lateral offset of every acquisition. The same shortcut doesn't help on a held-out subject, where the offsets are independent of those seen during training. This jitter thus results in a waste of valuable model space. Model parameters are used during training to remember the lateral jitter but are useless during evaluation. The result again is a decrease in model test performance, while simultaneously explaining the perfect training scores.

E.4 Widening of validation–test gap

The validation split operates at the Tx-firing level; thus, neighboring Tx firings from the same acquisition share the same frozen perturbation values. If a subset of this split is in the training set, the validation samples drawn from the rest can be classified by recognizing the shared per-acquisition signature instead of by reading the anatomy. The test fold, which sees a subject never used during training, cannot exploit the same shortcut, resulting in an increased validation–test gap.

F Supplementary Figures

Figures that support the main text but would disrupt the narrative flow if placed inline.

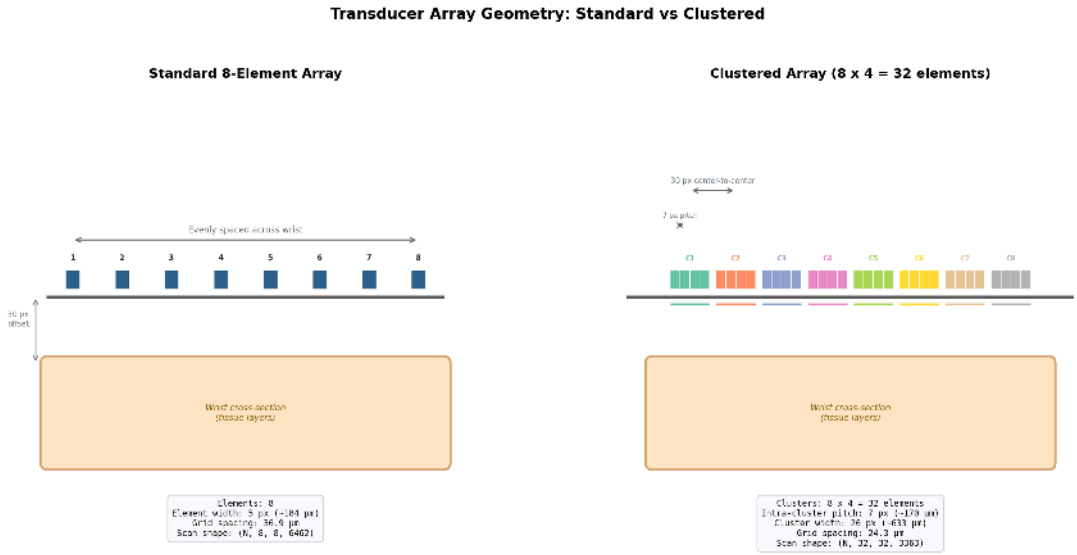

Figure 20: Linear 8-element array (left) and clustered 8×4 array (right). Numerical parameters for all three geometries are given in Appendix C.

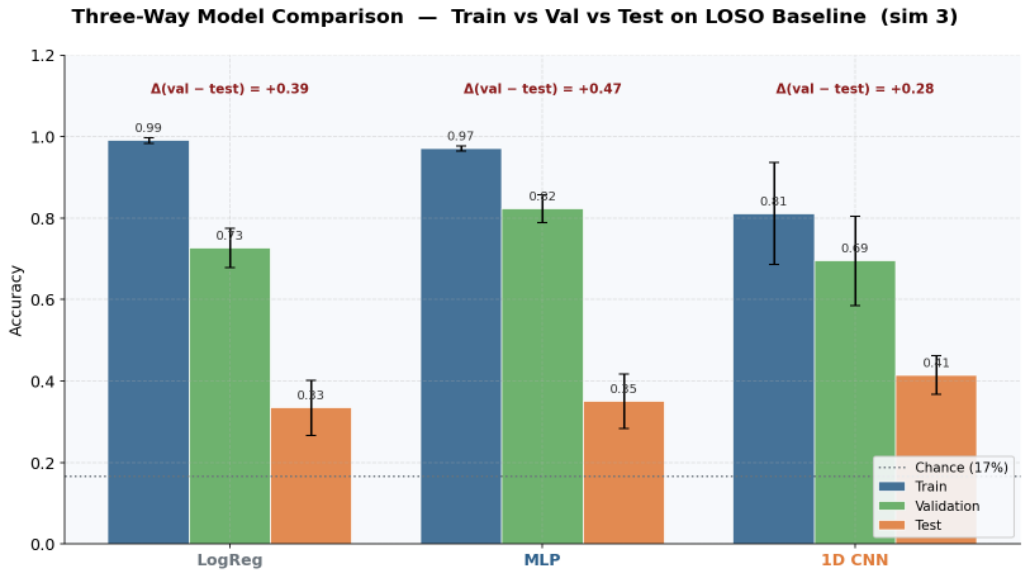

Figure 21: Train, validation, and test accuracy for logistic regression, MLP, and the 1D CNN under the LOSO baseline simulation. The flat baselines reach near-perfect training accuracy but collapse to chance-level test accuracy, while the 1D CNN retains higher cross-subject performance.

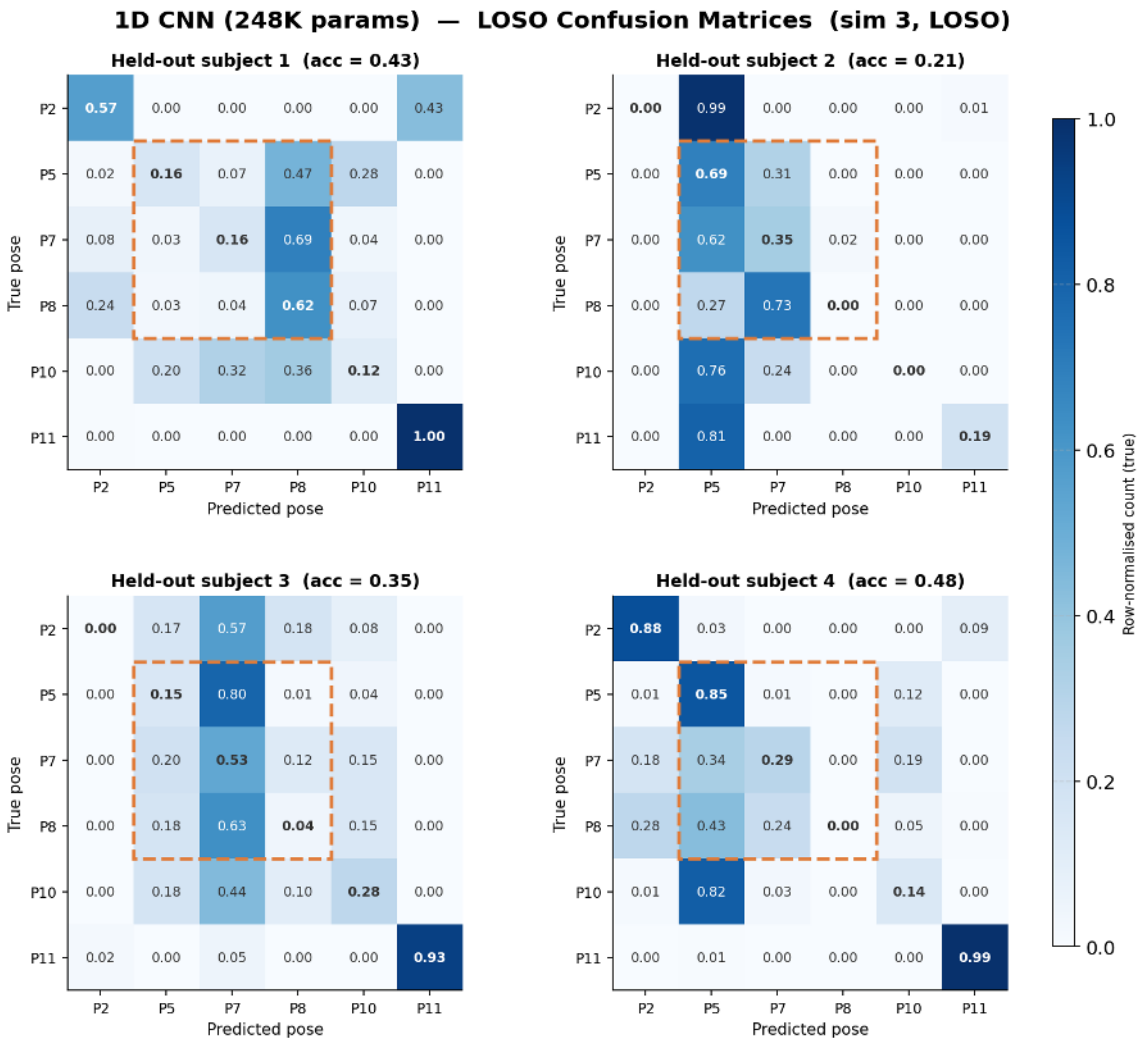

Figure 22: Per-fold confusion matrices of the chosen 1D CNN under LOSO. Poses 2, 10 and 11 are recovered cleanly on most folds; poses 5, 7 and 8 are mutually confused across every fold and subject.

References

[1] J. M. Fernández-Batanero, M. Montenegro-Rueda, J. Fernández-Cerero, and I. García-Martínez, "Assistive technology for the inclusion of students with disabilities: a systematic review," *Educational technology research and development*, vol. 70, pp. 1911–1930, 10 2022.

[2] W. H. O. (WHO), *Global Report on Assistive Technology*. World Health Organization, 2022.

[3] R. Zhang, Y. Hong, H. Zhang, H. Dong, and L. Dang, "A wearable electromyography armband for gesture recognition," *Biomedical Signal Processing and Control*, vol. 112, p. 108378, 2 2026.

[4] N. Tacca, C. Dunlap, S. P. Donegan, J. O. Hardin, E. Meyers, M. J. Darrow, S. C. IV, A. Gillman, and D. A. Friedenberg, "Wearable high-density emg sleeve for complex hand gesture classification and continuous joint angle estimation," *Scientific Reports*, vol. 14, p. 18564, 8 2024.

[5] C. A. Boles, S. Kannam, and A. B. Cardwell, "The forearm," *American Journal of Roentgenology*, vol. 174, pp. 151–159, 11 2012.

[6] Vijayalaxmi, M. Fatahi, and O. Speck, "Magnetic resonance imaging (mri): A review of genetic damage investigations," *Mutation Research/Reviews in Mutation Research*, vol. 764, pp. 51–63, 4 2015.

[7] C. L. MacIver, S. Ebden, and E. C. Tallantyre, "Mri: how to understand it," *Practical Neurology*, vol. 21, pp. 216–224, 6 2021.

[8] E. J. van Beek, C. Kuhl, Y. Anzai, P. Desmond, R. L. Ehman, Q. Gong, G. Gold, V. Gulani, M. Hall-Craggs, T. Leiner, C. T. Lim, J. G. Pipe, S. Reeder, C. Reinhold, M. Smits, D. K. Sodickson, C. Tempany, H. A. Vargas, and M. Wang, "Value of mri in medicine: More than just another test?," *Journal of Magnetic Resonance Imaging*, vol. 49, 6 2019.

[9] E. Eddy, E. J. Scheme, and S. Bateman, "A framework and call to action for the future development of emg-based input in hci," in *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pp. 1–23, ACM, 4 2023.

[10] K. A. Dishner, B. McRae-Posani, A. Bhowmik, M. S. Jochelson, A. Holodny, K. Pinker, S. Eskreis-Winkler, and J. N. Stember, "A survey of publicly available ¡scp¿mri¡/scp¿ datasets for potential use in artificial intelligence research," *Journal of Magnetic Resonance Imaging*, vol. 59, pp. 450–480, 2 2024.

[11] L. Oakden-Rayner, "Exploring large-scale public medical image datasets," *Academic Radiology*, vol. 27, pp. 106–112, 1 2020.

[12] T. Prioleau, A. Bartolome, R. Comi, and C. Stanger, "Diatrend: A dataset from advanced diabetes technology to enable development of novel analytic solutions," *Scientific Data*, vol. 10, p. 556, 8 2023.

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

[14] X. Yang, Y. Liu, Z. Yin, P. Wang, P. Deng, Z. Zhao, and H. Liu, "Simultaneous prediction of wrist and hand motions via wearable ultrasound sensing for natural control of hand prostheses," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 30, pp. 2517–2526, 2022.

[15] B. E. Treeby and B. T. Cox, "k-Wave: MATLAB toolbox for the simulation and reconstruction of photoacoustic wave fields," *Journal of Biomedical Optics*, vol. 15, no. 2, p. 021314, 2010.

[16] B. Treeby, B. Cox, and J. Jaros, *k-Wave: A MATLAB toolbox for the time domain simulation of acoustic wave fields User Manual*, 2012. Manual Version 1.0.1.

[17] B. Wang, G. Matcuk, and J. Barbič, "Hand MRI dataset," 2020. http://www.jernejbarbic.com/hand-mri-dataset.

[18] P. A. Yushkevich, J. Piven, H. Cody Hazlett, R. Gimpel Smith, S. Ho, J. C. Gee, and G. Gerig, "User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability," *Neuroimage*, vol. 31, no. 3, pp. 1116–1128, 2006.

[19] MONAI Consortium, "MONAI: Medical Open Network for AI," 2023.

[20] I. T. Jolliffe, *Principal Component Analysis*. New York: Springer, 2 ed., 2002.

[21] M. A. McDowell, C. D. Fryar, and C. L. Ogden, "Anthropometric reference data for children and adults: United states, 1988–1994," Vital and Health Statistics, Series 11 249, National Center for Health Statistics, 2009.

[22] S. A. Goss, R. L. Johnston, and F. Dunn, "Comprehensive compilation of empirical ultrasonic properties of mammalian tissues," *The Journal of the Acoustical Society of America*, vol. 64, pp. 423–457, Aug. 1978.

[23] L. Morchi, A. Mariani, A. Diodato, S. Tognarelli, A. Cafarelli, and A. Menciassi, "Acoustic coupling quantification in ultrasound-guided focused ultrasound surgery: Simulation-based evaluation and experimental feasibility study," *Ultrasound in Medicine and Biology*, vol. 46, no. 12, pp. 3305–3316, 2020.

[24] J. Robertson, E. Martin, B. Cox, and B. E. Treeby, "Sensitivity of simulated transcranial ultrasound fields to acoustic medium property maps," *Physics in Medicine and Biology*, vol. 62, no. 7, pp. 2559–2580, 2017.

[25] C. Holmes, B. W. Drinkwater, and P. D. Wilcox, "Post-processing of the full matrix of ultrasonic transmit–receive array data for non-destructive evaluation," *NDT & E International*, vol. 38, no. 8, pp. 701–711, 2005.

[26] K. Guk, G. Han, J. Lim, K. Jeong, T. L. Kang, E.-K. Lim, and J. Jung, "Evolution of wearable devices with real-time disease monitoring for personalized healthcare," *Nanomaterials*, vol. 9, no. 6, p. 813, 2019.

[27] J. A. Jensen, S. I. Nikolov, K. L. Gammelmark, and M. H. Pedersen, "Synthetic aperture ultrasound imaging," *Ultrasonics*, vol. 44, pp. e5–e15, 2006.

[28] T. L. Szabo, *Diagnostic Ultrasound Imaging: Inside Out*. Academic Press, 2 ed., 2014.

[29] F. J. Harris, "On the use of windows for harmonic analysis with the discrete fourier transform," *Proceedings of the IEEE*, vol. 66, no. 1, pp. 51–83, 1978.

[30] J. A. Jensen, "Field: A program for simulating ultrasound systems," *Medical & Biological Engineering & Computing*, vol. 34, pp. 351–353, 1996.

[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015.

[32] Anthropic, "Claude code," 2026. AI coding assistant. https://claude.com/claude-code.

[33] OpenAI, "ChatGPT," 2026. Large language model, GPT-5.3 (Pro). https://chat.openai.com.

[34] A. Lykourinas, C. Pendse, F. Catthoor, V. Rochus, X. Rottenberg, and A. Skodras, "Parameter-efficient deep learning for ultrasound-based human-machine interfaces," 2026. Submitted to ICPR 2026.

[35] R. N. Bakhru and W. D. Schweickert, "Intensive care ultrasound: I. physics, equipment, and image quality," *Annals of the American Thoracic Society*, vol. 10, no. 5, pp. 540–548, 2013.

[36] R. Resnick, D. Halliday, and K. S. Krane, *Physics*. John Wiley & Sons, 5th ed., 2001.

[37] N. Hettiarachchi, Z. Ju, and H. Liu, "A new wearable ultrasound muscle activity sensing system for dexterous prosthetic control," in *2015 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1415–1420, IEEE, 2015.

[38] P. Iakubovskii, "Segmentation Models Pytorch," 2019.

[39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[40] S. Gupta, S. Gahagnon, P. Tournoux, M. Pernot, and J.-F. Aubry, "Effect of the acoustic impedance mismatch at the bone–soft tissue interface as a function of frequency in transcranial ultrasound: A simulation and in-vitro experimental study," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 68, no. 5, pp. 1742–1754, 2021.

[41] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Pearson, 4 ed., 2018.