

Deep Learning for Localized-Haptic Feedback in Tactile Surfaces

Présentée le 2 février 2023

Faculté des sciences et techniques de l'ingénieur
Laboratoire d'actionneurs intégrés
Programme doctoral en robotique, contrôle et systèmes intelligents

pour l'obtention du grade de Docteur ès Sciences

par

Camilo HERNÁNDEZ MEJÍA

Acceptée sur proposition du jury

Dr D. Gillet, président du jury
Prof. Y. Perriard, directeur de thèse
Prof. B. Lemaire-Semail, rapporteuse
Dr X. Liu, rapporteur
Prof. S. Carrara, rapporteur

You never fail
until you stop trying.
— Albert Einstein

En memoria de mi padre Carlos Hernández Ahman y mi amigo Emilio Pineda Correa...

Dedicada al amor de mi vida, a mi madre, a mi familia y a mis amigos...

Acknowledgements

This thesis is the result of several collaborations and interactions with amazing human beings, without whom this work would not have been possible. I would like to use these few lines to recognize and thank the implication of the numerous people that supported me during my learning path at the Laboratory of Integrated Actuators (LAI) at EPFL. I am honored to have had the chance to meet each one of you and to have the opportunity to learn by your side.

I would like to start by expressing my deepest gratitude to Prof. Yves Perriard, my thesis director, for the fantastic opportunity to join his laboratory and for the excellent atmosphere at LAI provided by an unparalleled group of talented humans. Yves not only provided support throughout the tough times of my doctoral journey, but he encouraged crazy ideas while helping me to adapt to the academic world. I was always inspired by his leadership and by the confidence that he deposits in me and my colleagues. The atmosphere of the lab would not be the same without the experience of the SENIOR collaborators from the LAB. I am grateful to Yoan for his unconditional support and feedback, Alexis and Christian for the technical discussions and advice, and without a doubt, big thanks to Aline, Magda, Corinne, and Sandrine for their fundamental support and guidance on all administrative matters.

I am equally thankful to my scientific advisors Florian, Jonathan, and Xiaotao for the extended discussions, valuable recommendations, and rigorous guidance that they provided in the distinct stages of the project.

This project would not be a reality without the ICST team at ZHdK, I am deeply grateful to the whole team but especially to Stefano, Hanna, Sébastien, Eric, and Yuri for all the experiment sessions, extended discussions, haptics knowledge, and no doubt the fun. I am also indebted to the Swiss National Science Foundation (FNS) for funding this project.

Similarly, I would like to thank the members of the jury, Prof. Betty Lemaire-Semail, Dr. Xinchang Liu, Prof. Sandro Carrara, and Dr. Denis Gillet for their time and careful revision of my thesis, the challenging questions, and the valuable discussions during the private defense. I am also thankful to the different students that gave me the opportunity to supervise your master's and semester projects. Especial thanks to Jeremy, Marc, and Theophile.

Four years are a relevant share of the life of a person, and the best part of all is to have the chance to learn and grow with different generations. I had the chance to learn from the old guard Jasha, Xinchang, Guillaume, Louis, Morgan, Valentin, David, Lorenzo, and Trung. We created a new generation (that now is not new anymore), Thank you Sofia, Pooneh, Thomas, Guzmán, Kenny, Armando, Francesco, and Silje for the interesting work together and the fun activities. I want to highlight my core team, the teammates that were there during the

Acknowledgements

whole journey, Raphaël, Adrien, Sean, and Patricio. Thank you, guys, for the great ideas, encouragement, and support. Naturally, fresh blood comes to re-ignite the flame that is overshadowed by time and fatigue. I appreciate the energy from Stefania, Quentin, Marjan, Simon, Amine, Bhawnath, and Marc.

I deeply appreciate being part of the LAI team, which felt more like a family, not the blood family but the family that we choose. I really hope to stay connected with each of you, my friends.

Paolo, you should have been mentioned in each of the previous sections (Especially in the SENIOR Collaborators haha :). Without question, I owe you a big THANK YOU. Not only for the scientific and technical advice but the personal support during the fun and turbulent moments of my doctoral expedition. Thank you for your personal support, especially over the last two years. I consider you a trustworthy friend and an important member of the family that we choose.

I am thankful to Kamilo (BIROB), Sebastian (Veridas SA), Sylvain (PVLAB), Joël, and Tatjana (MLO) for the discussions and your support before and during my Ph.D.

Lastly, I want to thank my wife/girlfriend/best friend Susi, for her patience, love, and character which drive me every day. You are my everything. Thanks to our dog Lupe for always cheering us up. Thanks to my mother, my father, and my brother that have built my character and continue to teach me important lessons, even from the distance. I need to thank my family and my friends for keeping me up. You encouraged me to keep fighting and made my life happier during these fun but challenging steps.

Basel, December the 22nd, 2022

Camilo Hernández Mejía

Abstract

Touchscreens are nowadays the preferred choice for user interfaces in consumer electronics. Significant technological advances have been made in terms of touch sensing and visual quality. However, the haptic feedback offered by commercial products is still primitive, primarily because it affects the whole touch surface and can only render vibration buzzes.

Several researchers are currently concerned with enriching haptic feedback on touch surfaces. One approach is to use an array of transducers, bonded to the surface, and exploit the wave propagation phenomenon to operate in the far field of the actuators. For instance, the time-reversal method has been used to create elastic wave-fronts and obtain localized vibrations. One of the limitations of a wave-focusing strategy is the appearance of secondary displacement peaks in undesired locations, which lowers the contrast of the focalization.

This thesis explores the potential of state-of-the-art Deep Learning strategies to create alternative signals that allow to reduce the number of actuators required, improve the contrast ratio, and evoke novel feelings to the users of the device with vibrotactile feedback.

In this work, machine learning models are used to extract relevant features from an impact signal and predict the location where the impact has occurred. A transformation of the time-domain signal permits to obtain a representation that improves the precision of the prediction of the impact source location.

Furthermore, a new approach to optimize the localized peaks obtained with time-reversed impulse response signals is developed using Deep Neural Networks (DNNs) and Reinforcement Learning (RL). The optimization increases the peak amplitude and contrast ratio while ensuring that the peaks appear at the desired location.

Moreover, a novel approach to storing and generating time-reversed signals is developed using Generative Adversarial Networks (GANs). The effect of the diversity provided by deep generative models on the generated signals is evaluated. The effect on the properties of the localized peaks created with the GAN-generated signals is studied in an experimental setup. This effect inspires the development of a novel stimulation pattern for vibrotactile feedback. A transparent haptic demonstrator surface is developed. The demonstrator can create localized vibrations within the human perception range. This haptic surface is used to perform a human perception experiment that compares the perceived alertness provided by the novel pattern with the traditional stimulation pattern. The novel pattern is perceived as more alerting and evokes different perceptual sensations that can lead to novel vibrotactile cues.

Abstract

This exploratory work introduces state-of-the-art machine learning techniques into the time-reversal haptics field of study. It demonstrates the potential of these approaches to bring new knowledge into the haptics research field.

Keywords: Deep Learning, Digital Musical Instruments, Generative Adversarial Networks, Haptics, Localized Vibrotactile Feedback, Piezoelectric Transducers, Reinforcement Learning, Time-reversal Acoustics.

Résumé

Les écrans tactiles sont aujourd'hui le choix privilégié pour les interfaces utilisateur dans l'électronique grand public. D'importantes avancées technologiques ont été réalisées en termes de détection du toucher et de qualité visuelle. Cependant, le retour haptique offert par les produits commerciaux est encore basique, principalement parce qu'il affecte toute la surface tactile et ne peut reproduire que des vibrations simples. Plusieurs travaux s'intéressent actuellement à l'enrichissement du retour haptique sur les surfaces tactiles. Parmi tant d'autres, une approche consiste à utiliser un réseau de transducteurs, collés à la surface, et à exploiter le phénomène de propagation des ondes afin d'opérer dans le champ éloigné des actionneurs. De plus, la méthode d'inversion temporelle a été utilisée pour créer des fronts d'ondes élastiques et obtenir des vibrations localisées. L'une des limites de la stratégie de focalisation des ondes est l'apparition de pics de déformation secondaires à des endroits non désirés, ce qui diminue le contraste de la focalisation.

Cette présente thèse explore le potentiel des stratégies de Deep Learning les plus modernes pour créer des signaux de nature différente qui permettent de réduire le nombre d'actionneurs nécessaires, d'améliorer le rapport de contraste du pic et de créer des ressentis inédits aux utilisateurs d'un dispositif avec retour vibrotactile.

Dans ce travail, des modèles d'apprentissage automatique sont utilisés pour extraire des caractéristiques pertinentes d'un signal d'impact et prédire l'endroit où ce dernier s'est produit. Une transformation du signal temporel permet d'obtenir une représentation qui améliore la précision de la prédiction de l'emplacement de la source de l'impact.

En outre, une nouvelle approche pour optimiser les pics localisés obtenus avec des signaux à réponse impulsionnelle inversée dans le temps est développée en utilisant des réseaux neuronaux profonds (DNN) et l'apprentissage par renforcement (RL). L'optimisation augmente l'amplitude du pic ainsi que le rapport de contraste tout en garantissant que les pics apparaissent à l'endroit souhaité.

En outre, une nouvelle approche du stockage et de la génération de signaux inversés dans le temps est développée à l'aide de réseaux antagonistes génératifs (GANs). L'effet de la diversité fournie par les modèles génératifs profonds sur les signaux générés est évalué. L'effet sur les propriétés des pics localisés créés avec les signaux générés par les GANs est également étudié dans une configuration expérimentale. Cet effet a inspiré le développement d'un nouveau motif de stimulation pour le retour vibrotactile.

Résumé

Une surface de démonstration haptique transparente a été développée. Le démonstrateur peut créer des vibrations localisées dans la gamme de la perception humaine. Cette nouvelle surface haptique a été utilisée pour réaliser une expérience de perception humaine qui compare la perception du niveau d'attraction de l'attention fournie par le nouveau motif avec le motif de stimulation traditionnel. Le nouveau motif est perçu comme plus alertant et évoque des sensations perceptives différentes qui peuvent conduire à de nouveaux signaux vibrotactiles. Ce travail exploratoire introduit des techniques d'apprentissage automatique de pointe dans le domaine d'étude de l'haptique à inversion temporelle. Il démontre le potentiel que ces approches ont pour apporter de nouvelles connaissances dans le domaine de la recherche haptique.

Mots-clés : Apprentissage profond, Apprentissage par renforcement, Haptique, Instruments de musique numériques, Réseaux antagonistes génératifs, Retournement temporel, Retour vibrotactile localisé, Transducteurs piézoélectriques.

Contents

| | |
|--|------------|
| Acknowledgements | i |
| Abstract (English/Français) | iii |
| Notation | xi |
| 1 Introduction | 1 |
| 1.1 Scope | 2 |
| 1.2 Thesis Overview | 2 |
| 2 State of the Art and Discussion | 5 |
| 2.1 Introduction | 5 |
| 2.2 Haptics | 6 |
| 2.2.1 Types of Haptic Feedback | 6 |
| 2.3 Musical Haptics: Digital Musical Instruments (DMIs) and Haptic feedback | 7 |
| 2.4 Mechanoreceptors: Human Perception of Vibrations | 9 |
| 2.5 Surface Haptics | 11 |
| 2.5.1 Direct Actuation | 11 |
| 2.5.2 Remote Actuation | 15 |
| 2.6 Selection of the Surface Haptics Approach | 20 |
| 2.7 The Time-Reversal Method | 21 |
| 2.7.1 TRM Signal Acquisition | 22 |
| 2.7.2 TRM - Metrics and Tradeoffs | 25 |
| 2.8 Actuators for Surface Haptics | 29 |
| 2.8.1 Eccentric Rotating Mass (ERM), Voice-coil Actuators and Linear Resonant Actuator (LRA) | 29 |
| 2.8.2 Piezoelectric Actuators | 30 |
| 2.8.3 Actuators Comparison | 31 |
| 2.9 AI, Machine Learning, and Deep Learning | 32 |
| 2.9.1 Machine Learning Implementation | 33 |
| 2.9.2 Types of Machine Learning Tasks | 33 |
| 2.9.3 Deep Learning for Signal Processing | 34 |
| 2.9.4 Deep Generative Models | 35 |
| 2.9.5 Discussion on Deep Learning Motivation for Haptics | 37 |

| | |
|--|-----------|
| 2.10 Conclusion | 38 |
| 3 Machine Learning for Impact Position Detection | 39 |
| 3.1 Introduction | 39 |
| 3.1.1 Traditional Approaches for Touch Detection and Impact Position Detection | 39 |
| 3.2 Methodology | 40 |
| 3.2.1 Domain Knowledge: Piezo-transducer and Plate Coupling | 41 |
| 3.2.2 Experimental Setup and Data Acquisition | 47 |
| 3.2.3 Data Pre-processing | 50 |
| 3.2.4 ML and DL Models, Training, and Testing | 51 |
| 3.3 Impact Position Detection Experiments | 52 |
| 3.3.1 Impact Position Detection: Iteration 1 | 52 |
| 3.3.2 Discussion | 57 |
| 3.3.3 Impact Position Detection: Iteration 2 | 58 |
| 3.4 Effect of the Impact Contact Duration | 65 |
| 3.5 Conclusion | 66 |
| 4 Reinforcement Learning for Localized Peak Generation | 69 |
| 4.1 Introduction | 69 |
| 4.2 Domain Knowledge on Peak Generation with TRM | 70 |
| 4.2.1 Impulse Response (IR) Acquisition | 70 |
| 4.2.2 Peak Generation and Comparison | 74 |
| 4.3 Methodology: Localized Peak Optimization | 82 |
| 4.3.1 Reinforcement Learning (RL) | 82 |
| 4.3.2 Actor: Signal Generation Strategies | 84 |
| 4.4 Pre-train the Decoder NN | 87 |
| 4.4.1 Supervised Training of the Actor | 89 |
| 4.4.2 NN Architecture Selection | 89 |
| 4.5 RL Experiments | 90 |
| 4.5.1 Reward Function | 91 |
| 4.5.2 RL Training | 92 |
| 4.6 Results and Discussion | 95 |
| 4.7 Conclusion | 97 |
| 5 Generative Adversarial Networks for Vibrotactile Feedback | 99 |
| 5.1 Introduction | 99 |
| 5.2 Generative Adversarial Networks (GANs) | 100 |
| 5.2.1 GAN Definition and Working Principle | 100 |
| 5.2.2 GANs for Time-domain Signals and Model Selection | 101 |
| 5.2.3 Generated Signal Evaluation Metrics | 102 |
| 5.3 Experiments with GANs | 103 |
| 5.3.1 Methodology | 103 |
| 5.3.2 Stage 1: Unconditional GAN | 103 |

| | | |
|----------|--|------------|
| 5.3.3 | Stage 2: Conditional GAN | 111 |
| 5.3.4 | GAN Diversity Effect on the Localized Peak | 117 |
| 5.3.5 | Discussion: Diversity Perception Effect | 119 |
| 5.4 | Conclusion | 120 |
| 6 | Human Perception Experiments and Time-reversal Haptics Demonstrator | 123 |
| 6.1 | Introduction | 123 |
| 6.2 | Methodology | 124 |
| 6.2.1 | Ethics Committee Approval (HREC) | 125 |
| 6.3 | Development of the Tr Haptics Demonstrator | 125 |
| 6.3.1 | Aluminum Beam Experimental Setup | 126 |
| 6.3.2 | Single-Piezo Glass Plate | 128 |
| 6.3.3 | Eight-Piezo Glass Plate | 133 |
| 6.4 | Perception Experiments | 142 |
| 6.4.1 | Single Vs Randomly-distributed Peak | 142 |
| 6.4.2 | Threshold Experiment Outlook | 150 |
| 6.5 | Outlook on the Haptics Demonstrator as a DMI | 150 |
| 6.6 | Conclusion | 151 |
| 7 | General Conclusion | 153 |
| 7.1 | Original Contributions | 155 |
| 7.2 | Outlook | 155 |
| A | Neural Network Architectures | 157 |
| A.1 | Impact Position Detection NN Architectures | 157 |
| A.1.1 | Fully Connected NN | 157 |
| A.1.2 | 2D Convolutional NN (2DCNN) Architecture | 158 |
| A.2 | RL Actor NN Architectures | 159 |
| A.2.1 | RL Actor 1: Fully Connected NN Architecture | 159 |
| A.2.2 | RL Actor 2: 1D Convolutional Neural Network (1DCNN) Architecture | 159 |
| A.3 | GAN Architectures | 160 |
| A.3.1 | WaveGAN | 160 |
| A.3.2 | Contitional WaveGAN (cWaveGAN) | 161 |
| B | One-Bit Piezo Driving Electronics Development | 165 |
| B.1 | Single-piezo Diver | 165 |
| B.1.1 | Components Layout | 165 |
| B.1.2 | Components Calculation | 167 |
| B.2 | Eight-channel Piezo Driver | 169 |
| | Bibliography | 171 |
| | Curriculum Vitae | |

Notation

Acronyms (abbreviations)

| | |
|-----------------|--|
| 1D | One Dimensions |
| 2D | Two Dimensions |
| 2DCNN | Two-dimensional Convolutional Neural Network |
| 3D | Three Dimensions |
| AI | Artificial Intelligence |
| ANNs | Artificial Neural Networks |
| CAD | Computer-Aided Design |
| cGAN | conditional Generative Adversarial Network |
| CLV | Compact Laser Vibrometer |
| CNC | Computer Numerical Control |
| CNNs | Convolutional Neural Networks |
| cWaveGAN | conditional Waveform GAN |
| DC | Direct Current |
| DCGAN | Deep Convolutional GAN |
| DDPG | Deep Deterministic Policy Gradient |
| DL | Deep Learning |
| DMIs | Digital Musical Instruments |
| DNNs | Deep Neural Networks |
| DST | Dispersive Signal Technology |
| EPFL | École Polytechnique Fédérale de Lausanne |
| ERM | Eccentric Rotating Mass |
| FA | Fast Adapting |
| FEA | Finite Element Analysis |
| FFT | Fast Fourier Transform |
| FPGA | Field-programmable Gate Array |
| GANs | Generative Adversarial Networks |
| GPU | Graphics Processing Unit |
| HAPTEEV | HAPtic TEchnology and EValuation for digital musical instruments |
| HiL | Hardware-in-the-Loop |

Acronyms (abbreviations)

| | |
|----------------|---|
| HMI | Human-Machine Interaction |
| HREC | Human Research Ethics Committee |
| ICST | Institute for Computer and Sound Technology at ZHdK |
| IR | Impulse Response |
| LAI | Integrated Actuators Laboratory at EPFL |
| LIG | Linear Impact Generator |
| LRA | Linear Resonant Actuator |
| LReLU | Leaky ReLU |
| MAE | MEan Absolute Error |
| ML | Machine Learning |
| MOSFET | Metal–Oxide–Semiconductor Field-Effect Transistor |
| MRF | MagnetoReological Fluids |
| NN | Neural Network |
| PC | Portable Computer |
| PCA | Principal Component Analysis |
| PCB | Printed Circuit Board |
| PDMS | PolyDiMethylSiloxane polymer also known as silicon |
| PMMA | Poly Methyl MethacrylAte or Acrylic |
| PZT | Lead Zirconium Titanate. The most common piezoelectric material |
| RANDOM | Random Stimulus, randomly distributed vibration |
| ReLU | Rectified Linear Unit |
| RL | Reinforcement Learning or Deep Reinforcement Learning |
| RMS | Root-Mean-Square |
| RMSE | Root-Mean-Square Error |
| SA | Slow Adapting |
| SAW | Surface Acoustic Waves |
| SINGLE | Single Stimulus, Single-point localized vibration |
| SNSF | Swiss National Science Foundation |
| STFT | Short Time Fourier Transform |
| ToA | Time of Arrival method |
| TrIR | Time-reversed Impulse Response signal |
| TrIRDs | Time-reversed Impulse Response Dataset |
| TRM | Time-Reversal Method |
| VAEs | Variational Auto-encoders |
| VHB | 3M Very-High Bonding Tape |
| WaveGAN | Waveform GAN |
| ZHdK | Zurich University of the Arts |

Symbols

| | | |
|-----------------|--|------------------|
| A | Location where the impulsive stimulus is given/reconstructed | — |
| B | Location where the elastic waves are recorded/induced | — |
| β | Non-dimensional piezo and plate material property ratio | — |
| B_s | Piezo driving signal bandwidth | Hz |
| C_r | Contrast Ratio | — |
| C'_r | Modified Contrast Ratio | — |
| $CLV_{out}(t)$ | Compact Laser Vibrometer output over time | V |
| C_ϕ | Phase velocity | m/s |
| C_{rMAX} | Upper bound for the maximal achievable Contrast ratio | — |
| D | Plate's bending stiffness | Nm |
| d_{31} | Piezoelectric coefficient | pm/V |
| Δ | Actuator free piezoelectric strain | V/m |
| ΔL | Piezo material deformation in the polarization direction | m |
| $\Delta L'$ | Piezo material deformation perpendicular to the polarization direction | m |
| D_{piezo} | Diameter of the piezoelectric transducer | m |
| E | Young's Modulus | N/m ² |
| F | Finger pressing force | N |
| f_{max} | Highest frequency contained in the piezo driving signal | Hz |
| f_r | Localized peak repetition frequency | Hz |
| f_s | Sampling rate | Hz |
| f_{start} | Swept-sine start frequency | Hz |
| f_{stop} | Swept-sine stop frequency | Hz |
| γ | Piezo-actuator stress/strain ratio | N/m ² |
| h | Surface Thickness | m |
| J | Total mechanical energy provided by the piezo actuators to the plate | J |
| k_f | Flexural wave wavenumber | rad/m |
| λ | Wavelength | m |
| λ_{min} | Minimum achievable wavelength | m |
| l | Surface length | m |
| L_r | ML Learning Rate | — |
| M | Number of points in Y for impact acquisition or vibration scanning | — |
| m | Mass | kg |
| μ_z | GAN latent vector Gaussian noise mean | — |
| $m_x = m_y$ | Internal bending moment induced by the piezoelectric transducer | Nm |
| N | Number of points in X for impact acquisition or vibration scanning | — |
| n_s | Signal number of samples | — |
| ν | Poisson's Ratio | — |
| n_z | Length of the GAN latent vector or latent vector dimension | — |
| ω | Angular velocity | rad/s |

Symbols

| | | |
|-----------------|---|-------------------|
| P | Estimated power consumption | W |
| P_Q | Average power provided by each transducer | W |
| Q | Number of transducers | – |
| Q_a | Number of active transducers | – |
| R | Reinforcement Learning Reward Function | – |
| ρ | Mass density | kg/m ³ |
| ρ_z | Non-dimensional piezo to plate thickness ratio | – |
| R_s | Spatial resolution for the localized peak | m |
| R_{scan} | Vibration scan spatial resolution | m |
| R_t | Temporal resolution for the localized peak | s |
| S | Surface area of the plate | m ² |
| σ^2 | DDPG Noise variance | – |
| σ_z | GAN latent vector Gaussian noise standard deviation | – |
| T | Time period | s |
| t | time | s |
| t_{peak} | focusing time or instant when the localized peak appears | s |
| τ | Attenuation factor for the surface or plate | s |
| T_c | Plate time constant | s |
| $T_{contact}$ | Impact contact duration | s |
| T_s | Piezo Driving Signal Duration or Time-reversed signal length | s |
| T_{sweep} | Swept-sine duration | s |
| t_z | Piezoelectric transducer thickness | m |
| u | Displacement of the surface | μm |
| $u_a(t)$ | Displacement at location A over time | μm |
| u_{peak} | Peak amplitude | μm |
| V | Voltage across the electrodes of the piezoelectric transducer | V |
| V_{drive} | Piezo Driving voltage | V |
| V_{piezo} | Piezo driving signal or recorded voltage | V |
| w | Surface width | m |
| x_a | Desired peak, impact, or vibration scanning location in the X axis | m |
| x_b | Piezoelectric transducer location in the X axis | m |
| $x_{desired=c}$ | Desired peak location in the X axis, also GAN condition c | m |
| x_{end} | Impact, or swept-sine acquisition, or vibration scanning ending point | m |
| x_{peak} | Measured location for the localized peak | m |
| x_{start} | Impact, or swept-sine acquisition, or vibration scanning starting point | m |
| y_a | Desired peak, impact, or vibration scanning location in the Y axis | m |
| y_b | Piezoelectric transducer location in the Y axis | m |
| y_{end} | Impact, or swept-sine acquisition, or vibration scanning ending point | m |
| y_{start} | Impact, or swept-sine acquisition, or vibration scanning starting point | m |

1 Introduction

The interest in developing haptic capabilities for tactile surfaces has increased in the last several years. Touch screens are the most relevant interface in the context of human-computer interaction. Nevertheless, information feedback is usually achieved via visual and auditory modalities. These modalities have proven slower than the tactile channel and might not be viable under certain circumstances (bright and/or noisy environments). Moreover, academia and industry rely on the assumption that adequately designed haptic interfaces can improve tactile device user experience, performance, and usability [1, 2]. For instance, it has been demonstrated that using rich vibrotactile feedback can increase the quality of multi-touch or multi-user interactions with tactile screens [3].

Digital musical instruments (DMIs) can significantly gain from haptic technology. Previous studies demonstrated that, while playing traditional instruments, the musician's haptic channel is involved in a complex action-perception loop [4]. This is why re-establishing a rich haptic exchange between musicians and digital interfaces would improve user experience, enhance performance control, and support expressivity. In addition, haptic musical interfaces have a high potential for providing guidance in musical tuition, large ensembles, and remote performance. Moreover, haptics could grant access to music for persons with somatosensory loss, the visually and even the hearing impaired.

The EPFL – *Laboratoire d'actionneurs intégrés* (LAI), in partnership with the ZHdK – *Institute for Computer-music and Sound Technology* (ICST), carried out the HAPtic TEchnology and EValuation for digital musical instruments (HAPTEEV) project funded by the Swiss National Science Foundation (SNSF Grant # 178972). This project pursued the design, implementation, and evaluation of novel haptic technologies for rendering rich vibrotactile feedback DMIs. A crucial part of the HAPTEEV project addresses the modeling, design, and implementation of an advanced haptic surface and is the key motivator for this thesis.

1.1 Scope

The main driver of this thesis is to propose new methods to obtain multi-touch and localized vibrotactile feedback over a surface. Furthermore, having a transparent surface is of high interest to enable the future integration of a screen for multi-modal interaction. For this purpose, wave-focusing strategies appear attractive because they allow to move the actuators to the border of the surface while creating peaks of displacement on the far field of the actuators. The localized peaks can be repeated to create a localized vibration and provide feedback to the users.

In general, wave propagation strategies have one drawback: additional displacement peaks appear in undesired locations over the surface. To provide high-quality localized feedback, it is key to lower the amplitude of these peaks with respect to the main peak. The ratio between the amplitude of the peak and the displacement elsewhere is called the "contrast ratio". Different mathematical and experimental approaches have provided solutions to improve the contrast ratio and increase the amplitude of the localized peaks (they are further described in [Chapter 2](#)).

Building on these approaches, this thesis aims to evaluate the potential of different state-of-the-art Deep Learning (DL) techniques to model and generate the signals that are used to create a converging wave field to obtain localized peaks of displacement over a surface. The main hypothesis is that "Deep Learning strategies can create alternative signals that permit to reduce the number of actuators required, improve the contrast ratio, and evoke novel feelings to the users of the device with vibrotactile feedback." The scientific approach is to answer a set of research questions. Their answers can be considered as the building blocks to validate this hypothesis. The relevant research questions and the associated hypothesis are presented at the beginning of each chapter.

1.2 Thesis Overview

This thesis is divided into seven chapters as follows:

[Chapter 2](#) presents a state-of-the-art review of the most relevant concepts to develop this thesis. The concept of haptics and digital musical instruments are first introduced. Then, the most relevant surface haptics methods are compared, and the selection of the wave-focusing method is motivated. Later, the selection of the actuation strategy is discussed. Lastly, the relevant topics in machine learning and deep learning are introduced, and the motivation to use them to improve the vibrotactile feedback is presented.

[Chapter 3](#) proposes an iterative approach to develop an impact position detection system based on Machine Learning that can be used to detect the position of a finger contact over a surface. This study allows us to understand how Machine Learning (ML) and DL can extract relevant features from the impact signals to predict the position of the impact. The impact

signals are acquired using an experimental setup with a single transducer. The focus of the chapter is to understand the best representation of the impact signal that helps the ML model to predict the impact source location. In addition, the effect of the impact contact duration on the prediction error is studied.

Chapter 4 develops an alternative approach based on deep learning to generate and optimize a localized peak. First, two different methods to acquire the time-reversed signals that can create a localized peak are compared. One of the two methods is selected and used to acquire a dataset of time-reversed signals. A deep neural network is pre-trained to generate the time-reversed signals. Then it is optimized to maximize the contrast ratio and amplitude of the localized peak. The signal representation proposed in Chapter 3 helps to simplify the optimization process. The properties of a peak obtained with the traditional time-reversed signal are compared to a peak obtained with a signal from the optimized neural network.

In Chapter 5, deep generative models are used to establish an alternative method to store and generate the time-reversed signals. An existing generative model is modified to control the type of signal generated, thus the location where the peak will appear. The chosen deep generative model provides some diversity compared to traditional neural networks. This means that the signals that are generated are not always the same. Profiting from the diversity in the generation process, a novel pattern of stimulation is developed. The novel pattern is hypothesized to increase the perceived alertness compared to the vibration from the traditional wave-focusing method.

In Chapter 6, the time-reversal method is used to conceive a transparent haptic demonstrator. The effect of the boundary conditions on localized peak amplitude is studied. The amplitude, contrast ratio, and resolution of the localized peak are compared to the theoretical values. Some alternatives to further improve the quality of the peak are proposed. Moreover, a model to predict the peak amplitude based on the driving voltage and the number of actuators is developed. A perception experiment is designed, and the haptic demonstrator is used to validate the hypothesis from the novel pattern. A threshold perception experiment is carried out to understand the effect of the novel pattern on the perception threshold.

Chapter 7 concludes the work of this thesis by answering the research questions from each chapter. The main contributions of this thesis are highlighted, and the research outlook is presented.

2 State of the Art and Discussion

2.1 Introduction

In the context of human-machine or human-computer interaction, it is of critical importance to give feedback to the user. When interacting with objects in the real world, humans have the possibility to perceive static and dynamic properties of the objects they are manipulating. This ability is deprived when using a touch-screen to interact with virtual objects, digital musical instruments, or remotely operated robots.

This chapter introduces the concept of haptics and links it with the different applications in the musical context. The different receptors that are involved on the perception of vibration are described. In addition, an overview of the main methods for surface haptics is presented, which motivates the choice of surface haptics method and drives the selection of the actuating technique. The Time-Reversal Method (TRM) is studied and its limitations are explained. The advancements in Machine Learning (ML) for Signal processing inspire the exploration of novel strategies to improve the haptic feedback while reducing the amount of energy.

In Section 2.2, the concept of haptics is first introduced and defined, then a list of the different types of haptic feedback is presented based on the sensation they can evoke. This is then followed by Section 2.3 in which the importance of haptics in digital musical instruments is brought forward and some of the most interesting works are presented. Then, Human vibrotactile receptors and their properties are detailed in Section 2.4.

After that, in Section 2.5, the most relevant methods for surface haptics are compared and the physics behind them is explained. An emphasis in wave-focusing techniques is made, specifically the TRM.

Section 2.6 motivates the selection of the surface haptics approach implemented in this thesis. While Section 2.7 goes further into the TRM and the associated engineering tradeoffs. In Section 2.8, the most relevant actuators to vibrate a surface are listed and the criteria to choose the actuating method are argued.

Finally, in Section 2.9, different ML techniques for signal processing and signal generation are introduced. In this section, the inspiration to develop novel techniques for vibration rendering, based on Deep Learning (DL), is discussed along with some recent applications of DL in haptics.

2.2 Haptics

Haptic technology was developed to recreate the experience of touch by means of motion, forces, or vibrations. The main objective is to be able to recreate virtual objects and to interact with them. For instance, computer simulations, augmented reality, remote operation of machines in hazardous environments, and high-precision surgical tools are some of the applications where haptic technology can be used to reproduce, in a natural way, the interaction with different objects in the real world.

The term "Haptics" was introduced nearly one century ago, when psychophysicists studied human touch-based manipulation and perception of objects [5]. Nevertheless, the first haptic technology was implemented 30 years later. By adding vibration into the control joystick of a supersonic aircraft, it was possible to alert the pilots of dangerous flight conditions (e.g. stall) [6]. Additionally, force feedback from the servo systems was integrated into the joystick to recreate the sensation of directly manipulating the control surface of the vehicle (e.g. aileron, elevator, rudder). Nowadays, bulk vibration for tactile stimulation is incorporated into everyday devices (e.g. smartphones, touchscreens, and cars). While this might be sufficient for daily use, this avenue to provide feedback to the user is limited in providing high-quality and high-efficient interaction for professional scenarios.

2.2.1 Types of Haptic Feedback

Haptic feedback can be classified in different manners. Three main categories are defined based on how they are implemented and what are the sensations that they can recreate:

- **Force Feedback** stimulates the kinesthetic sense by means of force and motion. It is able to reproduce the sensation of manipulating or holding a real object and the dynamic aspects of such interaction, for instance, the weight and inertia of a body [7]. An example is shown in Fig. 2.1. In general, this type of interaction is referred to as kinesthetic Haptic feedback.
- **Vibration or Vibro-tactile Feedback** stimulates the tactile sense, so properties like texture, temperature, pressure, or vibration can be perceived [9]. This category is the main focus of this thesis and will be further discussed in Sections 2.5 and 2.8. An illustration can be seen in Fig. 2.2. The implementation of this feedback is known as surface haptics.
- **Contact-less Haptics or mid-air haptics** is a newer implementation of haptic technology that stimulates the sense of touch by using focused acoustic radiation (i.e. ultrasonic



Figure 2.1: Force Feedback Haptics. Device and figure from [8].



Figure 2.2: Vibrotactile Feedback Haptics. Prototype and figure from [10].

waves focusing), allowing the user to interact with a 3D space without having physical contact with an input device. In 2013 Carter *et al.* presented UltraHaptics [11]. This mid-air haptics system uses a phased array of ultrasonic transducers, operating at 40 kHz, to project haptic feedback in multiple points in front of a screen. The working principle is represented in Fig. 2.3.

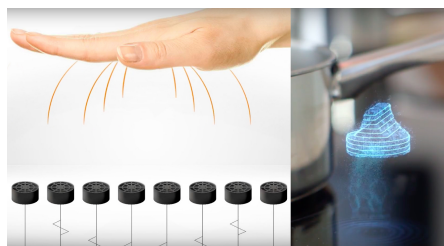


Figure 2.3: Mid-air Haptics. Technology and image from [12].

2.3 Musical Haptics: Digital Musical Instruments (DMIs) and Haptic feedback

Digital Musical Instruments (DMIs) are musical instruments that rely on computers or electronic circuits to generate sound. The increasing trend for including touchscreens in different devices has also influenced the DMI industry. For this reason, touch surfaces are now being used as open-ended interfaces that can be reconfigured for unlimited musical applications.

To give some examples, the most popular DMIs operated by touch are the AlphaSphere, the Soundplane, the Reactable, the Haken Continuum fingerboard, the Sensel Morph, and the Kaossilator [13]. Some of these devices are presented in Fig. 2.4. In addition, there is an immeasurable quantity of commercial applications for touchpads. For example the GeoShred Ipad app, Rotor Ipad app, Bebot Ipad app, among others.



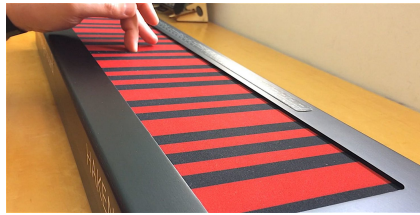
(a) AlphaSphere [14].



(b) The Madrona labs Soundplane [15].



(c) Reactable Mutli-touch surface [16].



(d) Haken Continuum [17].



(e) Sensel Morph [18].



(f) Korg Kaossilator[19].

Figure 2.4: Examples of Touch-based Commercial Digital Musical Instruments.

While these devices provide a robust platform to control the sound effects, none of them has haptic feedback. On DMIs the only output to the user is sound. Nonetheless, the auditory feedback is conceived to express musical performance, rather than presenting interaction information to the musician. This is a major drawback since it has been found that there is a complex action-perception loop between musicians and traditional acoustic instruments [4]. On this action-perception loop, the musician not only perceives the sound but expects to receive vibrational cues that can inform him about the musical process and provide a stronger performance control (e.g. Timing and dynamic accuracy). To unlock the full potential of DMIs it is vital to try to recreate this loop. One of the best ways to do so is to incorporate rich and multi-touch vibrotactile feedback into the device.

Different studies on vibrotactile and kinesthetic haptic feedback concluded their importance for instrument identification, pitch control, tempo-following tasks, among others. For example, in [20] the virtuoso and highly deaf percussionist *Evelyn Glennie* described her ability to recognize pitch, based on vibrations felt on different parts of her body. In [21], the au-

thors describe the positive impact of well-designed vibrotactile cues in gesture-based virtual instruments, improving temporal accuracy, velocity, and amplitude control.

Seeing that, some efforts have been made to develop haptic DMIs, by recreating the stimuli observed in real instruments or defining new paradigms for haptic feedback design. For instance, magneto-rheological technology and linear actuators have been used to recreate the touch response and dynamic behavior of keyboard instruments [22]. In addition, voice-coil and piezoelectric actuators have been used to induce vibrations on acoustic and electroacoustic instruments, and by these means, induce or dampen the resonance [23]. Also, friction control was used in [24] to study the effect in the Helmholtz motion in bow instruments (e.g. Violin).

More recently, Papetti *et al.* [25] proposed to add an array of piezoelectric actuators to the surface of a multi-touch interface, which allowed them to control the vibrations in the area of each transducer. The problem with this approach is the number of actuators required to cover the entire area. Besides, wave propagation and reverberation have to be taken into consideration. Moreover, this approach may limit the possibility to combine transparent touch surfaces with screens to achieve a more immersive interaction.

The musical applications of haptic feedback pose an interesting challenge to science and engineering. The development of such a prototype requires a 3D sensing technology (X and Y for position over the screen and Z for the Pressing force) and the implementation of multi-point localized vibrotactile feedback. The development of such devices will not only benefit the Music community but also the whole tactile industry. Since touchscreens capable of force sensing and haptic feedback can open a world of possibilities for Human-computer interaction.

2.4 Mechanoreceptors: Human Perception of Vibrations

To properly design haptic feedback, it is important to understand the interaction between the finger and the tactile surface. A **mechanoreceptor** is a specialized sensory receptor. It detects mechanical pressure or distortion in the skin and produces an electrical signal that is transmitted to the central nervous system, evoking sensations of touch.

Tactile perception theories rely on four mechanoreceptors to explain the perception of touch (the Meissner corpuscles, the Merkel cells, the Ruffini endings, and the Pacinian corpuscles) [26, 27, 28]. These receptors are organized by their sensory adaptation (FA for Fast Adapting, SA for Slow Adapting) and by their receptive field size (type I for a small receptive field, type II for large receptive field). Although there are four major receptors specialized in providing information about touch, this study will focus on Meissner's and Pacinian corpuscles, since they have been identified as the main FA mechanoreceptive channels that transduce information about vibrotactile perception. These mechanoreceptors are mainly located in the hairless areas of the body (hands, lips, and feet) also known as the glabrous human skin. Fig. 2.5 presents the distribution of the four mechanoreceptors in the human-finger pulp.

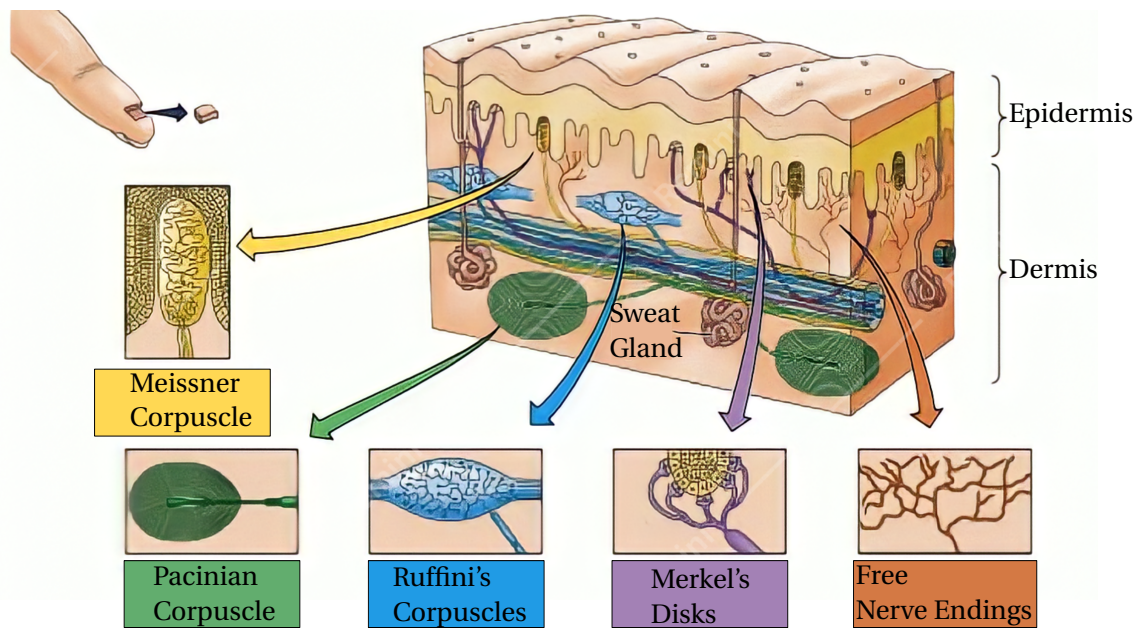


Figure 2.5: Mechanoreceptors in the glabrous skin (hairless skin) of the human finger [29].

The **Meissner's corpuscles** lie just under the epidermis of the fingers, palms, and soles. In the finger-pad, there is a high density of these receptors, around 100 corpuscles per square centimeter. This receptor is considered a fast-adapting (FA) type I channel. For one thing, it means that the electrical signal that is sent to the nervous system decays quickly when continuous mechanical stimulation is perceived (FA). For another that they have a small receptive field or a short sensitivity area (Type I), between 3 mm to 5 mm. Therefore, they are precise to locate the source of the perceived stimulus and they are very efficient for detecting low-frequency vibrations (10 - 100 Hz [26]) that may occur while exploring textured surfaces and shapes. Nonetheless, they are not very sensitive to low frequency (in the order of a few hertz) due to their fast adapting characteristics. Some interesting properties of this channel are that its detection threshold does not have a high dependence on the vibration frequency and that it is not subject to spatial or temporal summation.

The **Pacinian or Lamellar Corpuscles** are located deeper in the skin, in the subcutaneous tissue. There is a lower density of these receptors (around 30 corpuscles per square centimeter). This channel is FA and has a larger sensitivity area (Type II). Therefore, the spatial precision is low and the perceived vibrations are rather diffuse, non-localized, and perceived as a smooth vibration. The Pacinian corpuscles have a lower response threshold than Meissner's corpuscles and are mainly activated by higher frequencies (40 - 800 Hz [26]). For those reasons, they are related to the exploration of fine textures and are considered the main receptor for vibrational stimuli. The sensitivity of this channel depends on different factors. For instance, the threshold plot has an up-side-down U-shape with the highest sensitivity around 250 Hz [30], where the receptor can detect vibratory amplitudes down to $0.01 \mu\text{m}$ [31]. An illustration of these thresholds can be seen in Fig. 2.6. An interesting property of the FA II channel is that it is

capable of spatial and temporal summation. This means that a longer stimulus will have a lower threshold (i.e. the probability to perceive it is higher) compared to a shorter pattern with the same intensity. In the same way, the contact area has an effect on the perception threshold, a bigger contact area lowers the detection threshold or increases the perceived intensity.

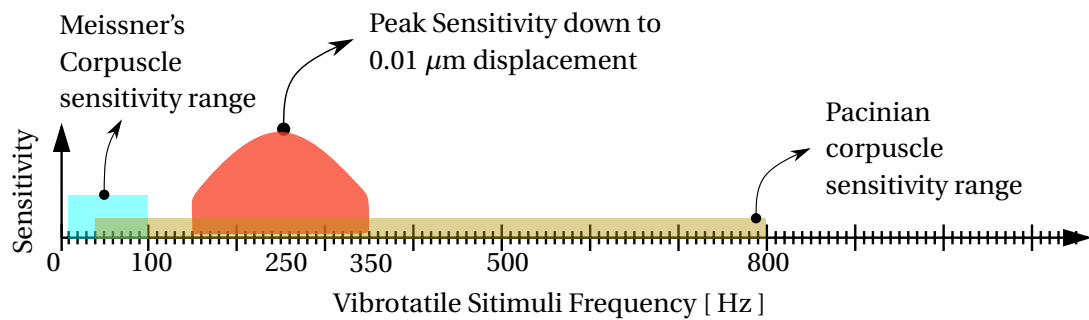


Figure 2.6: Sensitivity range for vibrotactile mechanoreceptors and illustration of the threshold for the Pacinian channel.

2.5 Surface Haptics

The development of sophisticated haptic feedback for interactive surfaces is attracting a lot of research interest. As a result, different strategies to evoke it have been developed. On a surface, mainly two kinds of feedback can be rendered. First, there is dynamic feedback or texture rendering, where the goal is to recreate the sensation when the fingers are moving on a certain surface. Then, there is static feedback, which aims to render clicks and localized vibrations. For example, pressing a button or feeling the vibrations on different positions of an acoustic instrument.

To create these haptic cues one can control the contact forces between the surface and the fingers. The different approaches to doing so can be classified into two main groups. Some of them involve the discrete deformation of the surface or independent moving elements that are in contact with the finger, which will be called direct actuation. Other methods involve a continuous surface that vibrates to stimulate the finger, these methods are grouped as remote actuation.

2.5.1 Direct Actuation

These methods are localized and can be multi-touch by design. In general, a discrete number of positions are actuated by different means and independently controlled to provide feedback to the users.

Pin-Array

The most common and one of the first tactile interfaces is the pin-array display. It consists of an array of pins whose motion is driven by an actuator to stimulate the finger or to create a specific texture to convey information to the user. The most common actuation methods are electromagnetic [32, 33], piezoelectric [34], shape memory alloys [35], shape memory polymers [36], or even electric motors [37]. One of the main applications of these devices is the representation of braille characters, but they can also be used to display figures, maps, or other geometrical figures. An example can be seen in Fig. 2.7. This approach provides the highest spatial resolution. But the main drawback is the long time-response, which limits the highest vibration frequency that can be presented for vibrotactile feedback.

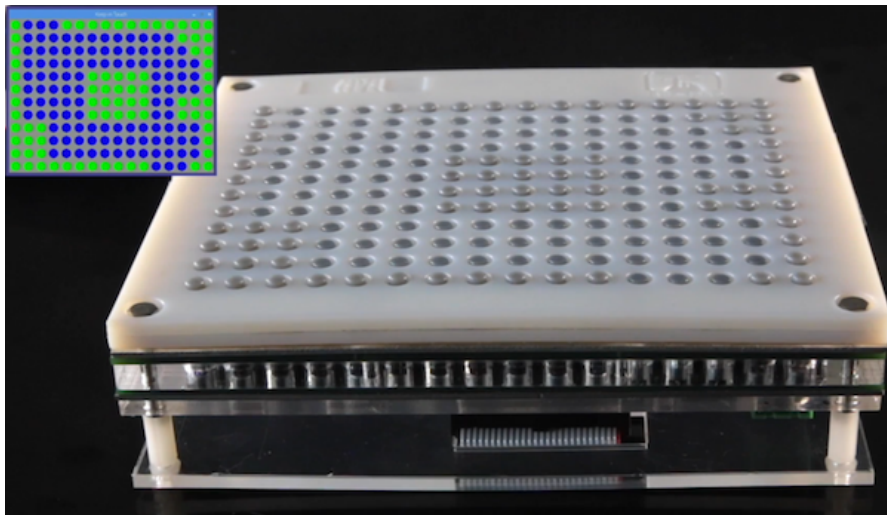


Figure 2.7: "Keep in Touch (KiT)": Pin-Array Haptic display presenting a map of a room. Original figure from [32].

Surface Deformation

Another method is the direct deformation of the tactile surface, which is generally a flexible material. In [38] a flexible surface was filled with magnetorheological (MR) fluid. The shape of the surface is modified by a magnetic field induced by an array of coils. The prototype is displayed in Fig. 2.8. In the same manner, air and particle jamming were used in [39] to control the shape and roughness of the surface.

In principle, these technologies can also render localized vibrational feedback. But, they are generally limited by the response time of the actuation technology that is used to control the discrete stimulation. For instance, for a pin-array device using shape memory alloy actuators [40] the maximum actuation frequency is limited to 1.5 Hz.

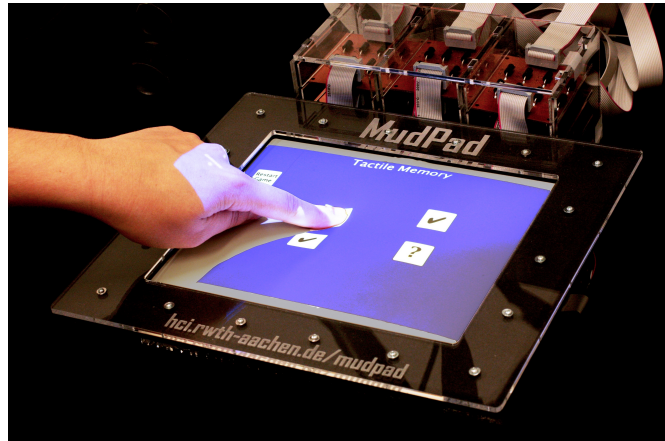


Figure 2.8: "MudPad": Magnetorheological (MR) fluid and an array of coils were used to modify the shape of the surface. Original figure from [38].

Thin Actuator Array

An alternative is to attach an array of thin actuators over the tactile surface and use them to locally render vibrotactile feedback. In [25] an array of 284 piezoelectric transducers were incorporated into the Madrona labs soundplane to render local vibrations in the area of the actuator, the device structure is presented in Fig. 2.9. This approach is affected by wave propagation. As a result, the vibrations from one actuator are perceived all over the surface. Recently, Hudin *et al.* [41] developed the Spatio-temporal inverse filtering to cancel the effect of one actuator (i.e. active wave propagation dampening) in the locations of the other actuators, enabling sectorized vibrotactile feedback.

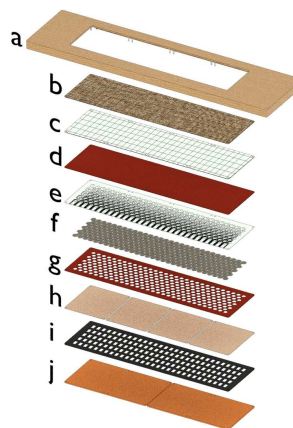


Figure 2.9: HSoundplane, Haptics enabled version of the Madrona labs sound plane as presented in [25]. Multi-layered construction of the HSoundplane: a) wood Enclosure; b) touch surface (wood veneer, 0.5 mm); c) Plexiglas plate (1 mm); d) natural rubber foil (1.3 mm); e) flexible PCB (0.3 mm); f) Honeycomb piezo actuators (0.2 mm); g) natural rubber holed foil (1.3 mm); h) carrier antennas; i) dielectric; j) pickup antennas. Original figure from [25].

A major drawback of the previously mentioned methods is that the drive and control are relatively massive since each position needs to be actuated independently. In addition, the spatial resolution is limited by the number of actuators and the position where they are located. Lastly, these methods are hard to integrate with a screen since the pins, flexible surface, and actuator arrays are generally non-transparent. There are some exceptions where an image was projected [38], but, the localization of the feedback is still limited by the discrete number of actuated areas.

Electrostatic Friction Modulation

A different approach is to create an electrostatic force that can be used to increase the friction between the finger skin and a conductive surface. This principle relies on the dielectric behavior of dry skin when an alternating voltage is applied. This voltage or current can be modulated in time, according to the position of the finger, to control the intermittent attraction force. Depending on the conductive surface geometry and voltage control this approach can provide localized and multi-touch feedback. This principle was used in [42] to recreate the sensation of different textures when a finger is exploring the tactile surface. The working principle is presented in Fig. 2.10.

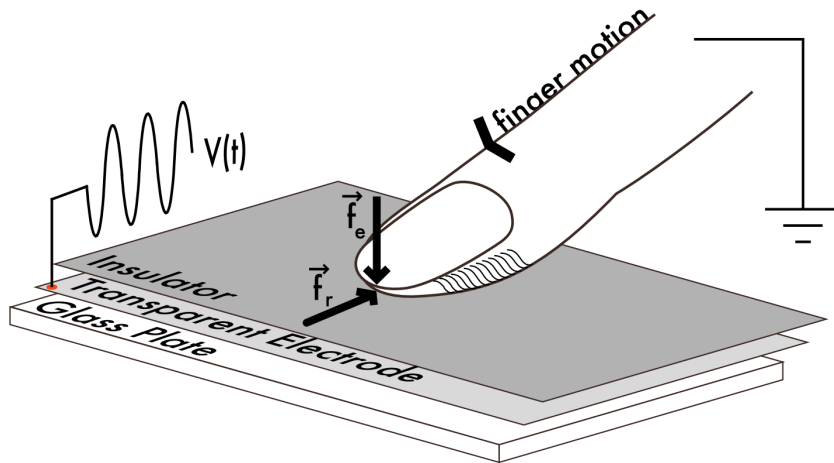


Figure 2.10: TeslaTouch: Operating principle for the electrostatic friction modulation system. Original figure from [42].

Controlling an electric charge is much simpler than controlling a vibrating surface. But, this method is highly affected by the conditions of the finger. To name some, variations on the humidity and resistance of the skin can drastically change the perceived sensation.

2.5.2 Remote Actuation

Instead of having multiple actuators directly in contact with the user, an interesting approach is to strategically move the transducers close to the boundaries of the surface. Then one can benefit from the waves that propagate inside the material to indirectly stimulate the fingers in contact with the surface. A big advantage of these approaches is that they can be implemented on transparent surfaces. Therefore, one can integrate a screen to provide visual information which leads to multi-modal interactions.

Global Vibration

Vibrating actuators have been integrated into handheld devices (e.g. mobile phones) to alert the user about an incoming call or confirm a button press. In general, Eccentric Rotating Mass (ERM) and Linear Resonant Actuator (LRA) are integrated into the structure of the device to vibrate the whole body. This basic yet broadly adopted method can only convey global feedback and simplified vibrational cues to the user. Thus, limiting the quality of the interaction with the device and affecting the control over the task that is being carried out.

Friction Modulation - Squeeze Film Effect

In a similar manner, a set of actuators can be attached to a thin plate and permit the control of the normal (i.e. out-of-the-plane) vibrations of the whole surface, as a result, an air gap or Squeeze Film is created. Then, by controlling the frequency of the vibration and taking advantage of the eigenmodes of the plate, it is possible to lubricate (with air) the contact between the skin and the surface. This spatial modulation of the friction coefficient is what the mechanoreceptors perceive as a texture. In [43] a single piezoelectric transducer is used to actuate a circular disk, the position of the contact is tracked and the friction reduction is activated accordingly. In [44], the authors used the Stroboscopic analysis to investigate the interaction of a glass plate vibrating at ultrasonic frequencies on the finger that is in contact with the glass, they confirmed the existence of an air cushion between the finger and the glass plate, which creates a dynamic levitation that reduces the contact between the skin asperities and the surface, thus reducing the friction.

In [45], an array of 12 piezo patches are actuated at 31.2 kHz to produce a vibration amplitude of 1.5 μm . The system consumes only 400 mW of power (for a voltage amplitude of 150 V_{pp}) and the prototype is implemented on a $93 \times 65 \times 0.9 \text{ mm}^3$ glass plate. The prototype is presented in Fig. 2.11.

The downside of the "squeeze film" effect and electrostatic forces actuation is that they require the skin (e.g. finger in contact) to have a movement relative to the surface, in order to have an effect. Also, these methods are highly affected by changes in the properties of the finger skin (e.g. humidity, resistance, or dirt on the surface).



Figure 2.11: Transparent tactile stimulator presented in [45] uses the Squeeze-film effect to control the friction coefficient. Original figure from [45] ©2012 IEEE.

Active Lateral Forces: lateral vibrations and electrostatic force

An alternative method to overcome this problem is described in [46], where an active lateral force of 400 mN is generated on the finger, thanks to the relative phase of lateral ultrasonic oscillations (around 30 kHz) and out-of-plane electrostatic forces. The working principle is depicted in Fig. 2.12. This method can be used to create both in-plane friction modulation and lateral vibrations. This stimulus is similarly perceived by the Pacinian channel since this receptor cannot easily differentiate between an out-of-the-plane and a lateral vibration.

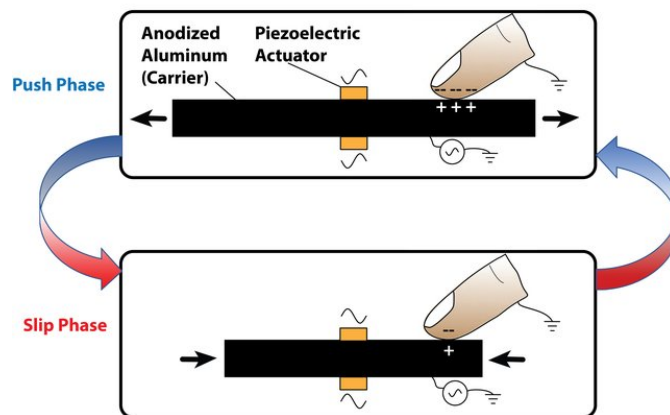


Figure 2.12: On the Ultrashiver, an active lateral force is generated by synchronizing the lateral vibration of the surface and the electro-adhesive effect. During the push phase the finger is attracted to the surface, while during the slip phase the adhesion is reduced to let the skin slide back. Original figure from [46] ©2018 IEEE.

Modal Composition

Using multiple actuators, it is possible to independently control the resonant modes of the surface (i.e. a thin surface can be considered a membrane). Different modes can be excited by manipulating the phase and amplitude of each actuator. By exploiting general inverse methods, one can create localized vibrations on a small region of the surface [47]. The principle is to create constructive and destructive waves to ensure the existence of vibration in the desired region. One approach is to create a map of the surface and pre-calibrate the actuating frequency for each actuator so that the vibration amplitude is maximized at the desired spot and minimized elsewhere. This principle was studied in [48]. Another option is to assign an actuator for each desired vibration mode and then create a pulse at the desired location. In general, this method can localize a vibration over a relatively large area. For instance, in [47], 34 actuators were able to actuate a spot of $50 \times 50 \text{ mm}^2$ area over a $268 \times 170 \times 0.7 \text{ mm}^3$ glass plate.

The three methods aforementioned are simple to implement, nevertheless, they can only be used for a single-touch interaction, since they affect the entire surface.

An alternative approach, proposed in [49], demonstrates that it is possible to synthesize a desired velocity field, at a given time, by controlling the rise time of a reduced amount of natural modes. The authors used 4 piezoelectric transducers, fixed to an aluminum bar, to excite the 6 modes that were chosen. In [50], perceptual experiments are carried out to confirm the human perception of the localized velocity field, the target velocity field, and the achieved velocity field are illustrated in Fig. 2.13.

One of the major drawbacks of this approach is the limited maximal frequency. Due to the long rise time of the lowest frequency modes, the focusing time is around 25 ms, which limits the maximum frequency to 20 Hz. Therefore, higher peak amplitudes are required since this stimulation frequency is far from the detection threshold peak of sensitivity.

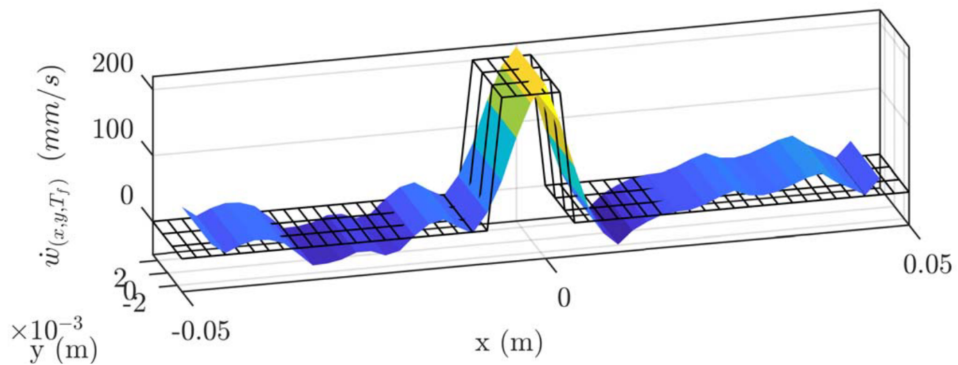


Figure 2.13: The target velocity field is presented as a black mesh. The colored surface represents the measured velocity field. Which is obtained after a truncated modal superimposition is carried-out. A limited number of natural modes are used. Figure from [50] ©2018 IEEE.

Wave focusing with the Time-Reversal Method (TRM)

An alternative to increase the frequency and focus on a smaller area (i.e. higher spatial resolution), is to use wave focusing strategies to create a localized peak of displacement. This peak can be repeated at a given rate to obtain a localized vibration. In [11], the phased array method was used to focus ultrasonic waves in a non-dispersive medium (air), evoking tactile sensations in the midst of the air, the device is presented in Fig. 2.3. The challenge with flexible waves in solid materials is that they are dispersive, affecting the focusing resolution [51]. In addition, the reflection of the waves in the borders of thin plates or "cavities" induces reverberations, which complicates things even more.

In 1992, *Prof. Mathias Fink* presented the Time-Reversal Method (TRM) whose principles are described in [52]. In TRM there are 2 steps, signal acquisition and wave focalization. First, an impulsive stimulus is generated at **position A** while the flexural waves that propagate over the surface are recorded at **position B**. Then, when the recorded signal is time-reversed and reproduced at **position B**, the initial pulse excitation is restored at **position A**. This strategy can be used to obtain a localized peak of displacement at a desired location. Since this approach takes into account the reverberation of flexural waves in a closed domain, a single actuator is able to reconstruct the vibrational field at virtually any position over the surface. The reconstruction of a wave field in a chaotic chamber using a single transducer was first presented in [53].

This method has multiple applications. Ranging from medical imaging and non-destructive testing to non-invasive medical treatment (kidney stones, micro-calcifications, contrast agents) [54]. It can also be used to localize the position of a finger impact on a thin-rigid surface, then the recorded signals can be sent back to drive an array of voice-coil actuators and provide haptic feedback [55]. In [56], the author used time-reversed impact signals to create localized peaks of displacement on an aluminum beam. A single piezoelectric transducer was driven in an analog manner. Similarly, in [51], the authors used pre-recorded time-reversed impulse response signals to obtain localized haptic feedback in a transparent surface (i.e. Borosilicate glass plate). They used 32 piezoelectric transducers driven in a digital fashion. They reported the generation of localized peaks with an amplitude of $0.7\text{ }\mu\text{m}$, focused in an area of 20 mm^2 with a power consumption of 45 mW, and repeated at a maximum rate of 500 Hz. The localized peak of displacement is presented in Fig. 2.14.

The main difficulty with the TRM is the appearance of secondary peaks due to the reverberations on the surface. The ratio between the main peak and the average displacement elsewhere is called the **contrast ratio**. Similar to the "signal-to-noise ratio" in signal processing, this metric measures the quality of the focalization. To ensure multi-touch operation and to improve the quality of the feedback, it is key to increase the contrast ratio. The existing solutions to overcome this challenge include increasing the number of actuators [57, 58], which makes driving more complex and increases the amount of energy that is required.

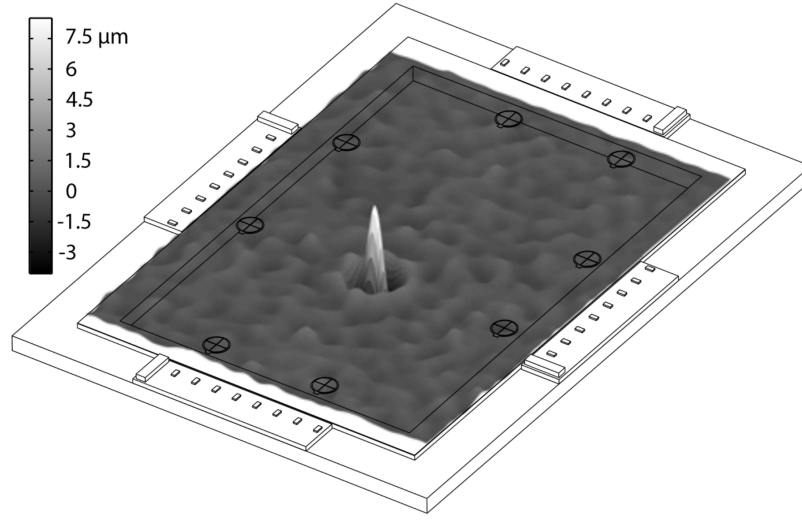


Figure 2.14: Displacement of the glass surface at the instant when the localized peak appears (focus time $t_{peak} = 2$ ms). Original figure taken from [51] ©2015 IEEE.

An alternative method to improve the contrast ratio was presented in [56]. In this case, the method was developed for an aluminum beam (i.e. a plate where the length is much larger than the width $l \gg w$). The author relies on mathematical modeling to find the locations where the associated secondary peaks appear. In this 1-dimensional setup, two types of peaks are characterized. By sending an inverted peak (i.e. a peak with the opposite direction) with a given gain, it is possible to cancel the peaks that contribute to the background noise. An example is presented in Fig. 2.15.

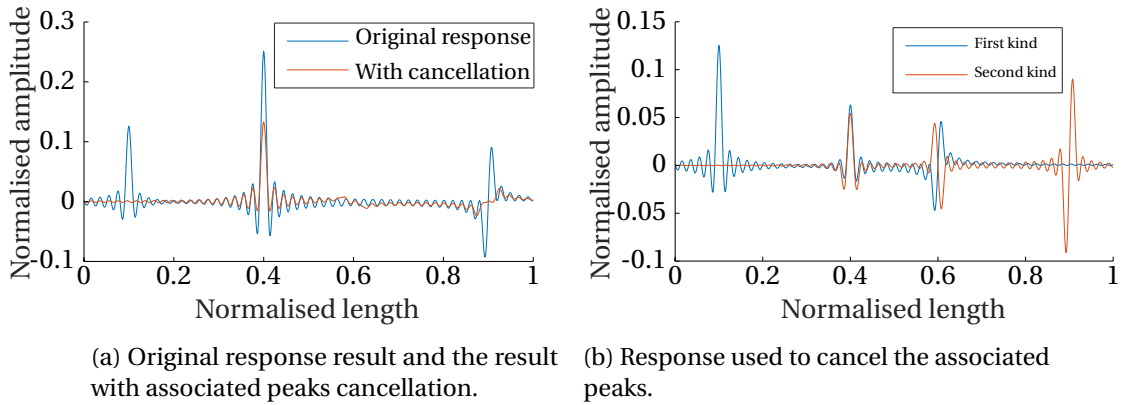


Figure 2.15: Cancellation of associated secondary peaks using one piezoelectric actuator, the actuator is located at $x_b = 0.25$ (i.e. 25% of the normalized bar length). The calculation is based on the approximate form of the surface amplitude with number of eigenmodes $n = 100$ and desired peak location $x_a = 0.4$ (i.e. 40% of the normalized bar length). Original figure from [56, p. 81].

The main drawback of this approach is that the cancellation of these additional peaks also affects the amplitude of the main focus point. As a consequence, it is necessary to increase the amount of energy that is given to the system. This is done either by adding more actuators or increasing the power provided by each actuator. In addition, this mathematical model has to be adjusted for every particular type of plate. In a 2D plate, the cancellation becomes more complicated and the types of peaks also increase.

2.6 Selection of the Surface Haptics Approach

Up to this point, an overview of the most relevant approaches to evoke surface haptics has been presented. The main motivation of this thesis is the development of haptic feedback methodologies that can be integrated into Digital Musical Instruments (DMIs). Musical instruments rely on multi-touch interactions to achieve higher control of the generated sounds. Furthermore, rendering rich and independent vibrational feedback to each finger result in more natural interactions and enable new interaction designs [4]. This is not only the case for musical applications, in general multi-touch interactions are more natural to the user and could impact a broad number of applications. Lastly, the possibility to have a transparent surface allows more immersive and multi-modal interactions. For these reasons, the focus of this research project is the generation of localized, multi-point, vibrotactile feedback on transparent surfaces.

Table 2.1 presents a comparison of the different surface haptics methods that were reviewed in this section. The criteria to compare them are: easiness of implementation (Difficulty), potential to be implemented on a transparent surface (Clear Surface), the capability of Multi-touch interaction (Multi-touch), ability to provide localized vibrational feedback (Localized Vibrations), requires finger movement relative to the surface (Needs Finger Movement).

Table 2.1: Surface Haptics methods comparison.

| Method / Criteria | Difficulty | Clear Surface | Multi-touch | Localized Vibrations | Needs Finger movement |
|-----------------------------------|--------------|---------------|-------------|----------------------|-----------------------|
| Pin-Array | Complex | No | Yes | Limited | No |
| Direct Surface Deformation | Complex | No | Possible | Limited | Yes / No |
| Thin Actuator Array | Complex | Limited | Yes | Limited | No |
| Electrostatic Friction Modulation | Simple | Yes | Possible | Limited | Yes |
| Global Vibration | Simplest | Yes | No | No | No |
| Friction Modulation | Intermediate | Yes | No | No | Yes |
| Active Lateral Forces | Complex | Yes | No | Yes | No |
| Modal Composition | Simple | Yes | Limited | Limited | No |
| Wave Focusing (TRM) | Intermediate | Yes | Yes | Yes | No |

The most suited methods for this thesis are modal composition and the TRM. Both methods can be implemented on a transparent surface and can provide feedback to a static finger. The modal composition method is simpler to implement. But, the locations where feedback can be provided are limited by the natural modes of the surface. In general, the spatial resolution is low and multi-touch interaction is hard to achieve since lower frequency modes have an effect on a big portion of the surface. While the TRM is slightly more complex, it allows to create sharp and localized peaks at any location over the surface. More importantly, it is possible to obtain simultaneous peaks at given locations by superposing the time-reversed signals. For these reasons, our choice is to continue exploring haptic feedback generation with the TRM.

2.7 The Time-Reversal Method

The Time-Reversal Method or Time-Reversal Mirror (TRM) is a signal-processing strategy to solve the problem of wave focusing through an inhomogeneous medium, which is an important problem in the acoustics research field. TRM benefits from the reversibility of wave propagation and the most remarkable property is that it is valid in in-homogeneous, dispersive and even scattering medium, as long as the medium is stationary [52].

In thin plates, where the thickness h of the plate is small compared to the bending wavelength, the wave propagation phenomena can be described by the Kirchhoff-Love plate theory. The wave propagation equation is defined as:

$$D\nabla^4 u + \rho h \frac{\partial^2 u}{\partial t^2} = \delta(t)\delta(x - x_a)\delta(y - y_a), \quad (2.1)$$

where ρ is the mass density, u the displacement of the surface, t time, $\delta(x, y, t)$ is the Dirac delta function that represents a unit point impulsive force in coordinates (x_a, y_a) , and D is the plate's bending stiffness described by:

$$D = \frac{h^3 E}{12(1 - \nu^2)}, \quad (2.2)$$

where E represents the Young's modulus and ν the Poisson ratio of the plate.

Since the wave equation (Eq. 2.1) contains only even order derivatives, it exhibits temporal symmetry. In other words, the solutions to this equation are invariant to the transformation of t to $-t$. This observation is the basis of the TRM.

If an impulsive force is produced at a certain location A (x_a, y_a) and the temporal evolution of the wave-field is measured at a discrete location B (x_b, y_b). Then, it is possible to reconstruct the initial state of a wave field in A. To do this, one can drive a transducer, located in position B, with the time-reversed version of the recorded signal. The principle is presented in Fig. 2.16. In essence, the TRM transforms the dispersive-propagating waves into a converging wave field at the focusing location. A comprehensive review of the TRM principle is described in [59].

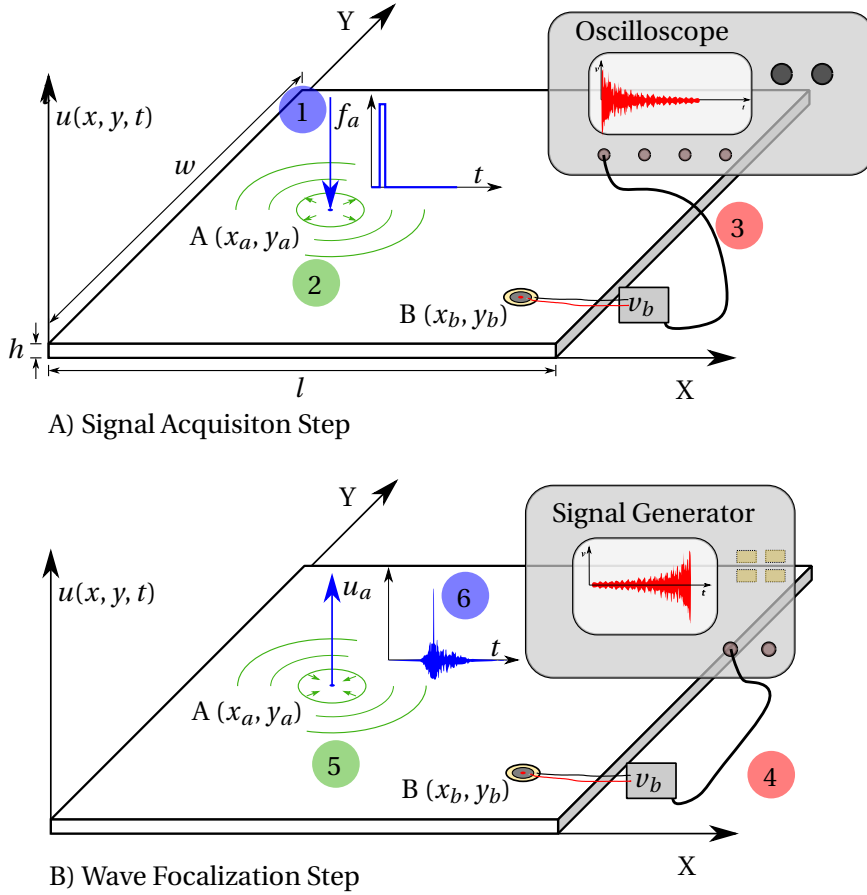


Figure 2.16: Illustration of the time-reversal method to create a localized peak of displacement on a bounded propagation medium. ① An impulsive force stimuli is produced in position A (x_a, y_a). ② A diverging wave field propagates on the surface. ③ The flexural waves are recorded in position B (x_b, y_b) during a period T of time. ④ The recorded signal is reversed in time and sent back to the transducer in position B. ⑤ A converging wave field is recreated on the surface. ⑥ After T seconds the initial wave-field is reconstructed and a peak of displacement appears at location A.

2.7.1 TRM Signal Acquisition

There are two approaches to obtaining the localized peak signal. The first one relies on a mathematical model or a Finite Element Analysis (FEA) simulation. The model is used to calculate the voltage signal on the piezo when an impulsive force is given at the location where the localized peak is desired. The second approach is to experimentally acquire impact signals.

Alternatively, one can experimentally find the impulse response between the actuator and different locations of a thin plate.

Impulse Response Mathematical Model Solution:

To solve the impulse-response wave equation (Eq. 2.1) one can assume a rectangular plate with simply-supported boundary conditions, then:

$$\begin{aligned} u(0, y, t) = u(l, y, t) = u(x, 0, t) = u(x, w, t) = 0, \\ u_{xx}(0, y, t) = u_{xx}(l, y, t) = u_{yy}(x, 0, t) = u_{yy}(x, w, t) = 0, \end{aligned} \quad (2.3)$$

where l and w are the length and width of the plate. The solution to the Kirchhoff-Love plate equation proposed by Kirchhoff can be defined in terms of the modal expansion:

$$u(x, y, t) = \sum_{m=1}^{\infty} \phi_m(x, y) q_m(t), \quad (2.4)$$

where $\phi_m(x, y)$ is the eigenfunction of the m -th mode of the plate that is assumed to be simply supported, and $q_m(t)$ is the modal Cartesian coordinate to be estimated. As demonstrated in [60, p. 15], the normalized eigenfunctions for Eq. 2.4 are represented by:

$$\phi_m(x, y) = \frac{2}{\sqrt{wl\rho h}} \sin\left(\frac{m\pi x}{w}\right) \sin\left(\frac{n\pi y}{l}\right). \quad (2.5)$$

For the initial conditions, it is assumed that the plate is at rest. Then, the initial displacement and speed of the plate are:

$$u(x, y, 0) = \dot{u}(x, y, 0) = 0, \quad \text{where} \quad \dot{u} = \frac{\partial u}{\partial t}. \quad (2.6)$$

One can substitute Eq. 2.6 into Eq. 2.4 to translate the initial conditions into the modal space:

$$q_m(t) = \dot{q}_m(t) = 0, \quad \text{where} \quad \dot{q}_m(t) = \frac{\partial q_m}{\partial t}. \quad (2.7)$$

Then, the solution to $q_m(t)$ is found by replacing Eq. 2.4 and Eq. 2.5 into Eq. 2.1:

$$q_m(t) = \frac{2}{\Omega_{mn}\sqrt{wl\rho h}} \sin\left(\frac{m\pi x_a}{w}\right) \sin\left(\frac{n\pi y_a}{l}\right) \sin(\Omega_{mn} t), \quad (2.8)$$

where:

$$\Omega_{mn} = \sqrt{\frac{D}{\rho h} \left[\left(\frac{m\pi}{w} \right)^2 + \left(\frac{n\pi}{l} \right)^2 \right]}. \quad (2.9)$$

Lastly, the substitution of Eq. 2.8 and Eq. 2.9 into Eq. 2.4 give the solution into $u(x, y, t)$:

$$u(x, y, t) = \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \frac{4}{\Omega_{mn} w l \rho h} \sin\left(\frac{m\pi x}{w}\right) \sin\left(\frac{n\pi y}{l}\right) \sin\left(\frac{m\pi x_a}{w}\right) \sin\left(\frac{n\pi y_a}{l}\right) \sin(\Omega_{mn} t). \quad (2.10)$$

Eq. 2.10 is the impulse-response function for an impulsive force at location (x_a, y_a) to any location (x, y) over the surface of the plate.

Impulse Response Simulation

Another approach to studying wave propagation is to use FEA Simulations. A transient analysis (i.e. a time-dependant simulation) can be used to estimate the displacement of the surface after an impulsive force stimulus is inputted at a given location. Tools such as ANSYS or COMSOL are commonly used for this purpose.

The main issue with simulation and mathematical modeling is that slight differences in the boundary conditions have a strong impact on the obtained signal. The principal effect occurs on the attenuation factor of the system.

Experimental Impact or Impulse Response Acquisition

The most practical approach is to acquire the signals using a real experimental setup. In general, there are two approaches to doing so. The first one is to generate mechanical impacts at a given location. Then, the flexural waves are acquired by one or multiple transducers at pre-defined locations. This method has been used in [55, 56]. This approach and the associated experimental setup are further described in Chapter 3.

The second approach is to acquire the impulse response (i.e. the transfer function) between the transducers and different locations on the surface of the plate. This approach has been studied in [51, 57, 58] and it is further explored in Chapter 4 and Chapter 5. It will be demonstrated that the Impulse response acquisition is easier to obtain (compared to mechanical impact acquisition). In addition, it contain much more information (i.e. natural modes) which leads to a better reconstruction of the localized peaks.

2.7.2 TRM - Metrics and Tradeoffs

As with every other method, the TRM has some limitations. First, the focalization duration should be limited. To reconstruct a localized peak, a portion of the time-reversed signal needs to be sent into the transducer. This means that it takes some time to obtain the localized peak. The longer the signal, the better the reconstruction. Nonetheless, the probability for signal distortion due to thermal drift increases with the length of the signal [57]. Second, the maximal repetition rate is limited. The duration of the signal plus the attenuation time (i.e. decay time) limit the maximum frequency of repetition that can be achieved. Third, the sharpness of the localized peak is affected by wave propagation. This method relies on flexural waves that propagate. Thus, it is natural that additional peaks of displacement appear on the surface. To achieve a high-quality haptic feedback, it is of essence to ensure that the peak amplitude at the desired location is larger than the average displacement on the surface.

To evaluate the quality and efficiency of the feedback with TRM, it is important to understand the most relevant metrics and the associated tradeoffs. Among them, are the contrast ratio (C_r), the amplitude of the peak (i.e. the displacement at the focus point A), the repetition frequency, the spatial resolution, and the power consumption (i.e. the energy balance).

Contrast Ratio C_r

The contrast ratio is similar to the "signal-to-noise ratio" in signal processing. In this case, the amplitude of the localized peak is compared to quadratic average displacement at all other locations. By analogy, this average displacement is called background noise in the context of this thesis. This metric is used to measure the quality of the localized peak. In [61] the contrast ratio is modeled in terms of the number of actuators, the mechanical characteristics of the plate, and the bandwidth of the time-reversed signal, C_r is defined as:

$$C_r = \sqrt{B_s T_c} \sqrt{\frac{Q\tau(1 - e^{-\frac{2T_s}{\tau}})}{(Q+1)\tau(1 - e^{-\frac{2T_s}{\tau}}) + T_c}}, \quad (2.11)$$

where B_s and T_s are respectively the bandwidth and duration of the driving signal, Q is the number of actuators, τ the attenuation factor for the vibrations in the plate, and T_c is the time constant for the plate, which depends on the mechanical and dimensional properties of the plate. T_c is represented by:

$$T_c = \frac{\sqrt{3}S}{h} \sqrt{\frac{\rho(1 - \nu^2)}{E}}, \quad (2.12)$$

where S , h , ρ , E , and ν are the surface area, thickness, mass density, Young's modulus and the Poisson ratio of the plate respectively.

Eq. 2.11 can be simplified by assuming the effects of modal decay can be neglected (i.e. when attenuation is negligible), this can be done when the length of the signal is much shorter than the attenuation constant, $\tau \gg T_s$. Then, Eq. 2.11 becomes:

$$C_r = \sqrt{B_s T_c} \sqrt{\frac{2QT_s}{(2Q+1)T_s + T_c}}. \quad (2.13)$$

From Eq. 2.13 it is possible to deduce the maximal achievable contrast ratio. This upper bound is equal to $C_{rMAX} = \sqrt{B_s T_c}$. This means that for a given plate, the signal bandwidth B_s has a proportional effect on the squared contrast ratio C_r^2 .

In addition, when attenuation is negligible, the contrast is maximized when $QT_s \gg T_c$. Seeing that, when a low attenuation plate is used, one can increase the time-reversed signal duration which enables the reduction of the number of actuators. Nonetheless, the impulse response and impact signals can drift significantly with the temperature variations, owing to the changes in the mechanical properties. This affects the performance of the focusing process. A solution to this problem is proposed in [57], [61], and [62]. It has been demonstrated that reducing the duration of the driving signal T_s reduces the sensitivity of the focusing process to environmental changes.

Peak Amplitude u_{peak}

Another important metric is the amplitude of the peak u_{peak} . In general, the human detection thresholds are given in terms of the displacement amplitude [30]. To ensure effective haptic communication, it is of key importance to ensure that the localized peak has an amplitude bigger than the detection threshold. In [57] the author demonstrated that u_{peak} is proportional to the number of transducers and the time-reversal window, following,

$$u_{peak} \propto Q\tau \left(1 - e^{-\frac{2T_s}{\tau}}\right), \quad (2.14)$$

then, Q and T_s not only have a proportional effect on the contrast ratio but also on the amplitude of the peak. However if the time-reversed signal duration is longer than the attenuation factor, $T_s \geq \tau$, the gain on the amplitude is attenuated.

Repetition Frequency F_r

A single peak of displacement is one way to provide feedback to the user. But given the temporal summation characteristics of the mechanoreceptors in the finger (which was described in Section 2.4). It is more practical to sustain the stimulation with a train of pulses. For repeated stimuli, the detection threshold decreases and the mechanoreceptors can perceive peaks with lower amplitudes. This train of pulses is perceived as a localized vibration. The frequency of

this vibration is limited by the attenuation factor.

To understand the limitations of this frequency, one can visualize the focusing process. To obtain a localized peak, the piezo is driven with the time-reversed signal during a period T_s , at the instant $t = T_s$ the wavefront converges and a peak of displacement is obtained on the target location. Right after this instant, $t > T_s$, the wavefront diverges from this location, and the oscillations decay with a time constant τ . When the repetition frequency is too high, this is when $T_r = 1/f_r$ is smaller than the plate attenuation factor, the background noise increases since the decaying wave fields build up. Thus, the contrast ratio is decreased. An expression for the effective contrast ratio is proposed in [51],

$$\hat{C}_r = C_r \sqrt{1 - e^{-\frac{2T_r}{\tau}}}, \quad (2.15)$$

where C_r is the achieved contrast ratio for a single peak. When the repetition frequency is lower than one over the attenuation factor, $f_r \leq (1/\tau)$, the initial contrast is conserved by more than 93%.

Spatial Resolution R_s and Temporal Resolution R_t

The time-reversal method permits achieving a localized peak in space and time. Given that the time-reversed signal has a finite bandwidth, the focus point has a finite dimension and duration in time. In general, the spatial resolution R_s is defined as the width of the localized peak at half maximum amplitude (approximately -3dB width). An example of the spatial resolution for a localized peak on an aluminum beam is presented in Fig. 2.17 (This experimental setup and the localized peak generation are presented in Chapters 3 and 4).

The spatial resolution is limited by the diffraction limit, which dictates that the minimal width of a peak on a vibrating medium depends on half the minimal wavelength of the flexural waves, λ_{min} , excited by the actuators,

$$R_s = \frac{\lambda_{min}}{2}. \quad (2.16)$$

Furthermore, the temporal resolution, R_t , is the duration of the peak at half-maximal amplitude. This value depends on the maximum frequency that is excited by the transducers, f_{max} . Then:

$$R_t = \frac{1}{2f_{max}}. \quad (2.17)$$

Knowing that the flexural waves are propagating in a bounded propagation domain (i.e. a finite plate), the wavelength and frequency are related by the dispersion law ([63, p. 236]),

then:

$$\lambda_{min}^2 = \pi \frac{h}{f_{max}} \sqrt{\frac{E}{3\rho(1-\nu^2)}}. \quad (2.18)$$

To find the relation between temporal and spatial resolution one can input Eq. 2.16 and Eq. 2.17 into Eq. 2.18. Then:

$$\frac{R_s^2}{R_t} = \frac{\pi h}{\sqrt{12}} \sqrt{\frac{E}{\rho(1-\nu^2)}}. \quad (2.19)$$

By definition, spatial resolution and temporal resolution are related. As a consequence, they cannot be chosen arbitrarily (i.e. spatial focusing with TRM is a transitory phenomenon). What's more, the spatial resolution depends only on the mechanical properties of the plate (e.g., the plate's bending stiffness, D) and the highest frequency stimulated by the transducers.

The spatial resolution is an essential parameter. It sets the minimal distance between two localized peaks, which is important for multi-touch interaction since it defines the minimal distance between two adjacent fingers.

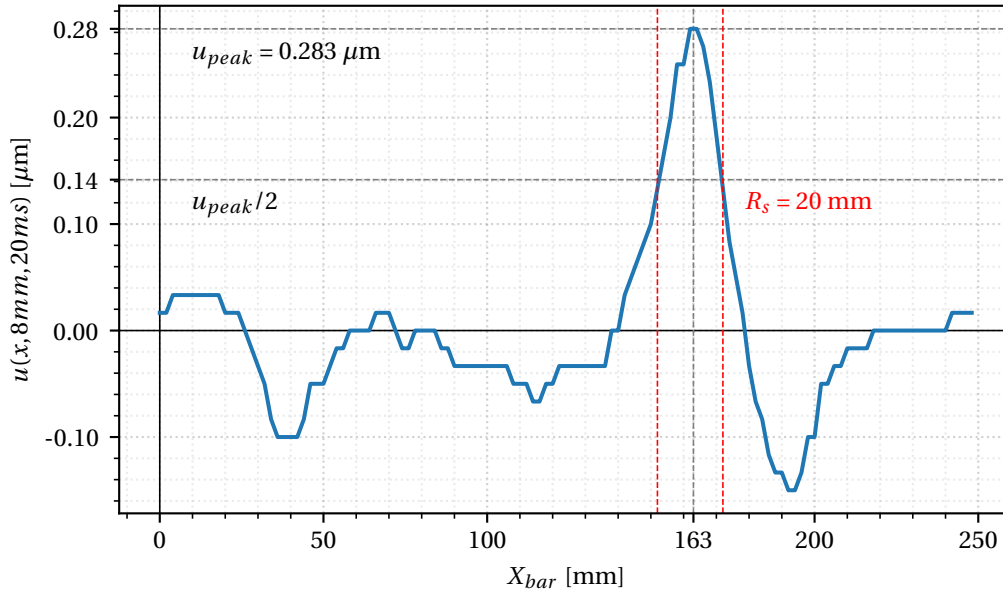


Figure 2.17: Localized peak spatial resolution. The figure presents the surface displacement scan for a localized peak created on an aluminum beam ($250 \times 16 \times 2 \text{ mm}^3$) with a single piezoelectric transducer. The peak is created using a time-reversed Impulse response signal with $T = 20 \text{ ms}$ and the focusing location is at position $x_a = 163 \text{ mm}$, $y_a = 8 \text{ mm}$.

Power Consumption P

This is the most important metric. The effectiveness of TRM for haptic feedback depends on the amount of energy that is needed to create a localized peak, with a given amplitude u_{peak} and spatial resolution R_s . The energy balance is obtained by evaluating the total mechanical energy in the plate, J , at the instant of focalization [51]:

$$J = \frac{\pi^3}{18} \frac{E}{1 - \nu^2} h^3 \left(\frac{u_{peak}}{R_s} \right)^2. \quad (2.20)$$

In other words, J represents the total energy that the actuators transfer into the plate during the time-reversal window T_s . Let the average input energy per transducer be $J_Q = J/Q$. Then the average power on each actuator is $P_Q = J/(QT_s)$. From Eq. 2.20 it is evident to conclude that the thickness of the plate h is a crucial parameter since the total energy is proportional to its cube.

These metrics will be used and further studied in Chapter 6 for the conception of a time-reversal haptic demonstrator and the validation of the effect of the novel haptic feedback strategies.

2.8 Actuators for Surface Haptics

The most common actuators, to create vibrations on a haptic surface are: Eccentric Rotating Mass (ERM), Voice-coil actuators including the Linear Resonating Actuator (LRA), and Piezo-electric disc or patch (Piezo). Even though the principle is the same, a driver which controls an actuator to generate vibrations, each one has its own advantages and limitations. Choosing the right actuator for the application is an important step in the design process. As presented in Section 2.5 there are other actuation technologies that are primarily used to render textures and directly stimulate the finger by deforming the surface, these actuation approaches are not the main focus of this thesis and they will not be further studied.

2.8.1 Eccentric Rotating Mass (ERM), Voice-coil Actuators and Linear Resonant Actuator (LRA)

An ERM is composed of a DC motor which has an off-center mass attached to the rotor. When it is activated, the oscillation of the mass creates simultaneous accelerations in two directions (X, Y, or Z axis). This is perceived as a vibration. This actuator is cheap and simple to control. But, the mass must reach a minimal speed to start generating a significant vibration. This is commonly known as start-up time, which varies between 50 - 100 ms for ERMs [64].

In addition, it takes the same time to slow down. Therefore, the overhead is around 100 - 200 ms to render a click, which is quite slow. The frequency of vibration can be controlled by

varying the input voltage. However, the amplitude is fixed, which restricts the diversity of the waveforms that can be generated. ERM are widely used in commercial devices, but due to their limitations, they are being replaced by LRA and Piezo actuators.

A **Voice-coil** is an electromagnetic actuator that behaves in the same manner as a loudspeaker. A coil is used to generate a magnetic field which in turn can be used to control the position of a magnet or a ferromagnetic material. This moving part is coupled to the actuated surface. In principle, this actuator can reproduce an arbitrary waveform with a fast response. But, the bandwidth is limited by the dynamic properties of the system (i.e. response time inertia, and coupling). Under certain circumstances, this transducer can be used as a sensor (i.e. to acquire vibrations from the surface). Nonetheless, Voice-coils are generally not optimized for the sensing task.

A **Linear Resonating Actuator (LRA)** is one type of voice-coil actuator. They are made up of a linear motion mass-spring system and a coil. When the driving system applies a current through the coil, a magnetic field is generated causing the movement of a magnet that is attached to the mass and vice-versa. When the system reaches its resonant frequency, it can generate great accelerations (1.7 g) [65], which are felt as vibrations. The acceleration occurs in a single direction, making this actuator more efficient (i.e. less energy consumption). This kind of actuator only works within a low bandwidth around the resonant frequency (± 2 Hz) due to the spring constant and has a start-up time of 20 – 60 ms [66].

2.8.2 Piezoelectric Actuators

Piezoelectric transducers are frequently used in the research community since they can easily convey high bandwidth signals (from a few hertz to several mega-hertz) by controlling a voltage signal. Also, it is easy to attach them to a surface and they can quickly generate high forces. The two major challenges are that they are generally controlled at high voltage (in the order of 60 V to 200 V) and that they generate small displacements (few μm). For these reasons, their adoption in commercial applications has been limited.

The piezoelectric effect occurs in monocrystalline materials and in polycrystalline ferroelectric ceramics. Two principal effects are defined for this materials [67]:

- **Direct Piezoelectric effect:** Piezoelectric coupling converts mechanical energy into electrical energy (i.e. Vibrating a piezoelectric material generates a voltage). Under this effect, the piezoelectric material can be used as a sensor and allows to acquire the vibrations that occur on a surface.
- **Inverse Piezoelectric effect:** Piezoelectric coupling converts electric energy into mechanical energy (i.e. Applying a voltage to a piezoelectric material generates a displacement). Under this effect, the piezoelectric material can be used as an actuator. For example, it can generate flexural waves on a surface (i.e. generate out-of-the-plane displacements).

The equations that relate the electrical and elastic properties to describe the piezoelectric effect are shown in Eq. 2.21 as described in [67]. These relations are valid only for small signal values (Small electrical and mechanical amplitudes):

$$\begin{aligned} D &= dT + \epsilon^T E, \\ S &= s^E T + dE, \end{aligned} \quad (2.21)$$

where D , d , T , and E are the Electric charge density, the Piezoelectric coefficient, the Mechanical stress, and the Electric field respectively. And S , ϵ^T , and s^E are the Mechanical strain, the permittivity (for constant T), and the Compliance or elasticity coefficient (for constant E) in the same order.

Generally, piezoelectric patches are a composite structure with three layers, electrode + Piezoelectric material + electrode. The most used type of electrode is silver. And there are many types of piezoelectric materials. The most common one is lead zirconate titanate also called Lead Zirconium Titanate (PZT).

2.8.3 Actuators Comparison

Table 2.2 presents a comparison of the three main types of actuators that are utilized for generating vibrational haptic feedback.

Table 2.2: Vibrotactile Haptics Actuators Comparison [65].

| Criteria | ERM/LRA | Voice Coil | Piezo |
|---------------------------|-------------------|--------------------|--------------------|
| Size [mm ³] | Bulky [8 x 3 x 3] | Bulky [9 x 3 x 5] | Slim [9 x 9 x 0.2] |
| Acceleration g [-] | 0.6-1.7 | 2.5 | 8 |
| Rise Time [ms] | 20-60 | 15-25 | <1 |
| Operation Voltage [V] | 3 | 3-5 | 15 - 120 |
| Energy per Click [mJ] | 15-17 | 15 | 1 - 8 |
| Custom Waveforms | No | Yes | Yes |
| Bandwidth | Limited | High | Highest |
| Actuation and acquisition | No | Possible | Yes |
| Force Sensing | No | No | Yes |
| Driver Design | Easy - Moderate | Moderate - Complex | Complex |

In general, piezoelectric patches achieve precise haptic actuation as they are able to transduce arbitrary signals with higher bandwidths when compared to ERMs and LRAs. A higher bandwidth permits the creation of richer and custom haptic cues. For example in [68], the

authors used piezo actuators to transduce signals in the range of 20 kHz to 70 kHz (50 kHz bandwidth). In addition, piezo-actuators have faster start-up times and they are able to achieve accelerations up to 8 g [66], resulting in stronger vibration. Furthermore, piezoelectric transducers can work in both directions, excite vibrations when given a voltage input and also acquire vibrations giving a voltage input proportional to the displacement. Owing to these facts, piezoelectric transducers are the most suitable actuators for this research work and will be used in the different experiments in this thesis. The two piezoelectric transducers that are used in this thesis are presented in Fig. 2.18.

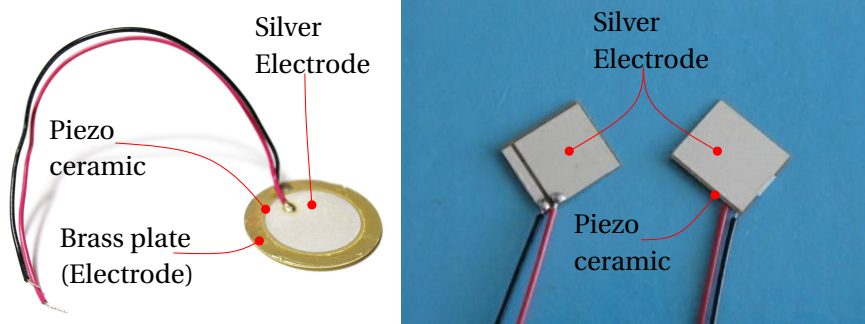


Figure 2.18: Off-the-shelf piezoelectric transducers. LEFT Murata 7BB-12-9 Piezoelectric Diaphragm [9 mm Piezo diameter]. RIGHT Steminc SM412 Piezo Ceraminc Plate [$7 \times 8 \times 0.2 \text{ mm}^3$].

2.9 AI, Machine Learning, and Deep Learning

The term **Artificial Intelligence (AI)** refers to the development of machines that are able to learn, reason, and make decisions by themselves. This field is evolving so fast that its definition changes constantly. However, most of the progress and applications that are mentioned today, refer to a sub-field of AI called **Machine Learning (ML)**. ML makes use of statistical tools to find patterns or "learn" from big sets of data. Later on, these patterns are used to predict or make decisions on new input data. In the same way, **Deep Learning (DL)** is a sub-category of machine learning which makes use of Neural networks, with multiple layers, to extract and amplify the least notorious patterns. This kind of models do not require any specific definition of the target patterns. They rather gather knowledge from experience. So in the multiple or "deep" layers, they have abstract representations of the patterns. I.e. Deep Learning is able to learn complicated concepts by building them out of simpler ones [69].

This section provides an overview of the implementation and the main types of ML / DL tasks. Then the use of Deep Learning for signal processing is presented. What's more, the state-of-the-art generative models that are relevant to raw signal generation are displayed. The main goal is to motivate the use of ML in the context of this thesis and explain the choice of the different ML approaches that will be described in the following chapters.

2.9.1 Machine Learning Implementation

To implement machine learning models, the first step is to obtain a dataset that can properly model the phenomena to study. It is key to have strong domain knowledge (i.e. to understand the nature of the data and its sources). This prior knowledge can guide the selection of the best representation of the data that will be fed into the model. The data set is divided into several subsets of data that are used to train, evaluate and test the model.

The next step is to use the biggest portion of the data to train the model, it is common to experiment with different models or neural network architectures, the objective is to find the one that permits achieving the required precision with the least amount of computational resources (i.e. find the simplest model that can provide the required precision).

The last step is to evaluate the quality of the trained model, for this purpose a small portion of the dataset (i.e. the test set is a data set that has never been seen by the model) is used to evaluate the robustness and generalization of the model.

2.9.2 Types of Machine Learning Tasks

There are two main types of ML tasks. The first one is the data discrimination or feature extraction task that usually contemplates either classification or regression problems [70]. The second one is data generation, which involves the augmentation of existing datasets with similar but not copies of the existing data points [69]. The first problem is studied in Chapter 3, in particular a regression problem. The goal is to find the position of an impact (contrary to generating a localized peak on a given position). By first solving the inverse problem, one can get a deeper understanding of the behavior of this data type in different ML architectures and better inform the generation process.

For data generation, the model is designed to depart from a lower dimensional space of features into generating realistic data points. For example, provided a category "dog" the model can generate a realistic image of a dog. Or given the desired location of a peak " $x_a = 200 \text{ mm}$ ", the model is able to synthesize a time-reversed signal to create a localized peak of displacement in that position.

The simplest approach to this task is to train a neural network to upsample from a single feature point into a realistic signal, this is called one-to-one mapping. This approach is explored in Chapter 4. This method does not yield a lot of diversity (i.e. not a lot of new data points), but, serves as a tool to optimize the signals towards a given goal (e.g. higher contrast ratio).

Another alternative is to rely on robust generational models. Instead of learning a simple mapping from a lower dimensional space into a single output, one can learn the statistical representation of the data. This way the model learns to generate realistically looking data samples while obtaining some diversity (i.e. not all the samples look the same). This is called

a one-to-many mapping. This approach serves as a framework to explore many alternative signals that can yield localized peaks on a given position. This method is explored in Chapter 5.

At the same time, ML problems can be subdivided into supervised and unsupervised learning, depending on the approach that is used to train the model. Supervised learning is the most common way of ML. In this case, the model learns from pre-labeled data points. For each training example, there is a set of input values and a label that serves as a target. The model is trained by minimizing the difference between the predicted class (for classification problems) or value (for regression problems) and the original label. Unsupervised learning differs in the sense that the training set does not contain a label and the model is trained by optimizing a given loss function (e.g. maximizing the separation between the data points).

2.9.3 Deep Learning for Signal Processing

Signal Processing is a branch of electrical engineering that models and analyzes data representations of physical events. Signal processing traces back to Joseph Fourier (1768-1830) when he developed the "Fourier transform" while he was working on the equations governing heat propagation. Signal processing is by definition an intersection of disciplines. Physical stimuli, as the source of information, mathematics as the tool to represent and analyze this data, and informatics as a framework to efficiently implement these models. In this context, a signal is a digitized representation of a physical quantity that changes over time. It can include, images, videos, audio waves, sensor data, among other sources of information.

In recent years, with the growing amounts of data, signal processing needs to rely on powerful processing tools such as deep learning. Many different fields have been positively impacted by ML strategies. Image processing was one of the first domains to make use of Deep Neural Networks (DNNs) to classify and extract features from image data. For this purpose, Deep Convolutional networks were developed [71], this architecture is inspired by the signal processing domain knowledge and is designed to exploit the structure of the data (i.e. images are a two-dimensional array that represents objects that can change in size, orientation and position).

Convolutional and De-convolutional NN (CNNs)

CNNs or **ConvNets**, make use of the convolution operator to reduce the dimensionality of the input data and a fully connected layer to extract the features or labels from the data. They are very effective for processing data with a grid-like topology (e.g. **time series data** being a 1D grid and **images** being a 2D grid of pixels). In general, CNNs have four main operators or blocks: convolution, non-linearization, pooling (i.e. sub-sampling), and classification (i.e. fully connected layer). Lately, these models have been successfully used in image recognition, image feature extraction, and image classification. A **reversed convolutional network** (i.e.

de-convolutional network) can be used to reverse the process, and create data from a set of input labels of features, one example of a Deep convolutional generative model to generate images is presented in [72], in this case, the architecture is an encoder-decoder network, where the first part reduces the dimensionality of the image into a lower dimensional vector called "latent space" and the second part up-samples this smaller vector to reconstruct the original image.

2.9.4 Deep Generative Models

CNNs as well as Deep Neural Networks enabled a whole new pathway to signal processing and drive the creation of many new deep learning architectures for data generation. As mentioned before, generative models aim to create new data samples by learning the distribution of the data. In other words, given a dataset X of data samples $\{x_1, \dots, x_n\} \in X$, one can assume that these examples were drawn from an underlying data distribution $p_X(x)$. Then the parameters θ of a neural network can be trained to approximate this distribution. Such a model could then be used to generate new samples that look like they could have been part of the original dataset.

Conditional Generative Models

Generative models become more useful when one can exert control over the samples that are drawn from the modeled distribution. This can be done by providing the model with a condition signal c , that contains information about the type of sample that should be generated. In this case, the model is trained to fit a conditional distribution $p_X(x|c)$ instead of $p_X(x)$.

The condition signals come in many different shapes and forms. For example, the condition can be a discrete label "cat" or it can be a raw signal itself. They can be sparse or dense conditioning signals (i.e. contain more or less information). The richer the signal the easier the modeling problem becomes because the uncertainty about the desired data point x is reduced.

Types of Generative models

The most common approaches to generative models are flow-based models and Adversarial models.

Likelihood-based models directly parameterise $p_X(x)$. The parameters θ are fitted by maximizing the likelihood of the data under the model:

$$\theta^* = \operatorname{argmax}_{\theta} \mathcal{L}_{\theta}(x) \quad \text{where} \quad \mathcal{L}_{\theta}(x) = \sum_{x \in X} \log p_X(x|\theta). \quad (2.22)$$

Chapter 2. State of the Art and Discussion

The most popular flavors of likelihood-based models are Auto-regressive models [73, 74] (Such as the PixelRNN [75]), flow-based models [76], and Variational Auto-encoders (VAEs) [77].

All of these frameworks have their pros and cons. VAEs allow to perform a compression of the data into a "latent space" and regenerate the original data sample from it. This enables the creation of new samples by manipulating the different values of the lower dimensional vector. The downside is that the generated samples tend to be slightly blurry (i.e. noisy).

Auto-regressive models are simple to train and achieve high quality. But, they are slow for generation and do not provide a "latent space" that can be easily manipulated to affect the generated outcome.

Flow-based models are fast for generation but they required a large number of parameters to be effective, which makes them impractical for mobile applications.

Adversarial models take a different approach to capture the data distribution. In Generative Adversarial Networks (GANs) [78, 79], two neural networks compete against each other. The first one works as a **Generator**. It aims to produce new artificial outputs, after learning the probability distribution of the training dataset. The second network work as a **Discriminator**. It aims to recognize if the output of the generator is real (i.e. if this new data is part of the real data distribution) or fake. This is done by comparing the generated samples with samples from the original dataset. The objective of the generator is to increase the error of the discriminator by producing data that looks as real as the training data. High-quality results have been obtained from different types of data, in [80] a fingerprint synthesis system was developed, and in [81] an authentic voice is created by generating a synthetic spectrogram and using the Griffin-Lim method to reconstruct the time-domain signal. More recently raw audio signals have been modeled with extremely high quality [82, 83, 84]. GANs currently generate the sharpest images and audio clips but the training process can be unstable (this is partially solved when training in a supervised fashion).

While adversarial and likelihood-based models are both ultimately trying to model $p_X(x)$. They approach this task from very different angles. GANs tend to be better at producing realistic examples, but worse at capturing the full diversity of the data distribution, compared to likelihood-based models.

When a model cannot capture all the variability in the data, different compromises can be made. If all examples should be reasonably likely. Then, the model will have to overgeneralize and put probability mass on interpolations of examples that may not be meaningful (**mode-covering**). If there is no such requirement, the probability mass can be focused on a subset of examples. But some parts of the distribution will be ignored by the model (**mode-seeking**) [85]. A key criterion to choose a generative model is the degree to which it has a mode-covering or a mode-seeking nature. On the one hand, Likelihood-based models are by definition mode-covering. This is a consequence of their objective, as they are fitted by maximizing the joint likelihood of the data. On the other hand, Adversarial models are typically mode-seeking,

which leads to higher fidelity. In general, mode-covering behavior is desirable in sparsely conditioned applications, where a big degree of “creativity” (i.e. diversity) is expected from the model. While, mode-seeking behavior is more useful in densely-conditioned settings, where most of the variability is captured in the conditioning signal, and realism of the generated output is preferred over diversity.

When considering time-reversed impact signals or time-reversed impulse responses, the data coming from different locations over the plate is quite homogeneous. In fact, the nature of the signals follows a similar trend (since the system is the same for all signals). In our application, there is no need to prefer a high diversity. But it is critical to ensure high quality in the signal generation. This is key to ensure the localized peak appears in the right location and to keep the noise to a minimum. For these reasons, this thesis will focus on mode-seeking models and in particular on the GANs framework. This model is further studied in Chapter 5.

Initial Application of GANs for Haptics

GANs applications for vibrotactile feedback have recently attracted some attention. There are few publications on the subject. For the moment the existing works focus on fine-tuning vibrotactile signals rather than generating the driving signal. In [86, 87] the authors use GANs, in real-time, to fine-tune the texture rendering. Note that no publications have been found (up to the date of writing the thesis) on the use of deep learning to generate vibrotactile feedback. Our novel approach is presented in Chapter 5.

2.9.5 Discussion on Deep Learning Motivation for Haptics

During the last 20 years, deep learning methods have proven to be successful at modeling different complex systems. In some cases overpassing the accuracy of mathematical models when sufficient data, that can properly describe the phenomena, is available. The tradeoff with this approach is the lack of interpretability of the representations and transformations inside the model. Nevertheless, in the field of time-reversal and wave focusing for vibrotactile feedback, different academic groups have provided an extensive and successful understanding of the interaction between the actuators and the response of the surface [51, 53, 55, 56, 58, 88]. This knowledge can guide the selection of ML strategies and the design of neural network architectures that can contribute to the localized peak generation and to reduce the amount of energy that is required.

One can get inspired by the recent success of GANs for RAW-Audio generation [82, 84] and the similarity in the nature of a time-reversed impact signal with a drum-beat audio signal (i.e. Drum sound effects) [83]. More details are given in Chapter 5. Thus, one can justify the need to explore the ability of deep generative models to capture the distribution of a dataset containing time-reversed impact signals and/or impulse responses, and evaluate the viability of real-time signal generation to obtain localized vibrations. Furthermore, one can use deep

learning methods to optimize the generated signals with the goal of improving the contrast ratio and reducing the amount of energy required to create a localized peak of displacement.

2.10 Conclusion

Different methods to obtain haptic feedback have been presented. It is clear that there is a high interest in the different applications of haptics both on tactile interfaces for Human-machine Interaction (HMI) and for Digital Musical Instruments (DMIs). The time-reversal method is the preferred method for this thesis. It enables multi-touch interactions and can achieve localized vibrations within the perception range. Besides, TRM can be implemented on transparent surfaces, which enables the development of multi-modal interactions by adding screens to the controller. Piezoelectric actuators are chosen as the actuation method since they can be used as sensors and actuators.

The time-reversal method has been properly modeled and remains an active topic among the research community. From the state of the art, it seems that the optimal point for time-reversal haptics has been reached. But, given the assumptions on the mathematical modeling and the potential to exploit the boundary conditions (reflections and particular conditions of the system), it is interesting to explore alternative solutions to improve the contrast ratio while decreasing the amount of energy that is required.

Machine learning and in particular Deep learning are powerful signal processing tools. In recent years they have impacted different research fields and it is gaining interest in the haptics community. In particular generative models can serve to discover novel approaches and alternatives to create localized haptic feedback.

3 Machine Learning for Impact Position Detection

3.1 Introduction

The best approach to introduce Machine Learning (ML) strategies into vibrotactile feedback is to study the inverse problem. This chapter focuses on the detection of the impact source location (impact position), which is indispensable to introducing ML into the peak generation task.

The research question for this chapter is *"How can ML models represent the relevant features of the impact data?"*. The derived hypothesis is that *"ML can accurately extract features from impact signals and improve the accuracy for impact position detection compared to traditional signal processing approaches"*.

In this chapter, the goal is to extract relevant features from the recording of an impact. This recording contains the flexural waves that propagate on the surface of a thin plate when an impulsive force is exerted on it. The chosen feature is the location where the impact occurred.

By solving the impact location problem, one can understand the behavior of impact signals when processed by ML models. Also, it is possible to gain a deeper understanding of the content of such signals. As a consequence, it is possible to find the best representation of the signal that helps on the extraction or estimation of the impact location. Furthermore, this technique can be used to add touch capabilities to large surfaces.

3.1.1 Traditional Approaches for Touch Detection and Impact Position Detection

For many years now, touch-screen technology has positioned itself as the principal interface for human-computer interaction. Different methodologies have been studied and developed to accurately detect the position of a finger that is in contact with a surface.

On the one hand, resistive and capacitive sensing are the most common technologies when it comes to small devices. On the other hand image processing, dispersive signal technology

(DST), surface acoustic waves (SAW), and time-reversal method have been also successfully implemented. The latter have proven more viable (effective) for larger surfaces. The use of touch sensing technologies can transform arbitrarily shaped solid objects into interactive interfaces as shown in [89] or into Digital Musical Instruments (DMI) [90].

A different application for impact position detection, which is highly related, appears in the field of composite materials for aerospace. Structural health monitoring is crucial for composite structures as they suffer of "barely visible impact damage" [91]. Thus, knowing the position and intensity of an impact is crucial to determine the maintenance tasks to preserve structural integrity.

The traditional approach relies on multiple vibration sensors (minimum three) and signal processing strategies. For example, the Time of Arrival (ToA) method or the cross-correlation of the impact signal with a reference signal, are commonly used to find the location of an impact. Nonetheless, these methods are highly sensitive to changes on the system and external sources of noise.

The use of piezoelectric transducers [91] or displacement sensors [92] and Artificial Neural Networks (ANNs), has proven to be an effective solution to reconstruct the impact force, recover the impact location and determine if damage has occurred in the structure. In the work by Jones *et al.* [92], the authors use Finite Element Analysis (FEA) to obtain the frequency response data and strain traces after a simulated impact has occurred. They use this data to train an ANN with a small dataset. Despite the small amount of data, the trained model was able to predict the position of the impact with an error smaller than 8 mm for experimental data on isotropic materials (i.e. an aluminum plate with dimensions $368 \times 584 \times 6.35 \text{ mm}^3$). Similarly, in [93], the authors used data from FEA and ANNs to detect the position where an impact occurred in the wing of an airplane. In [91], experimental impact data is recorded using piezo-ceramic sensors placed on a composite panel. ANNs are used to locate the position of the impact and extract the maximum force that was exerted. What's more, Worden *et al.* presents a Genetic Algorithm approach to find the optimal placement for the sensors. In [94], the authors use a larger number of spectral components as input to the ANN model. They study the uncertainty of the input variables and its propagation during the training of the model. They obtained an error of 14% for the impact force recovery and 9 mm error for the position detection (on a $230 \times 260 \times 3.4 \text{ mm}^3$ composite plate), which is not accurate enough to detect the position of a finger on a touch surface.

3.2 Methodology

To answer the research question, the proposed methodology is to study the problem in separate building blocks. A diagram with the general approach is presented in Fig. 3.1.

The first step is to build domain knowledge of the system. In essence, to understand the system that is being modeled, particularly the interaction of the piezo with the surface.

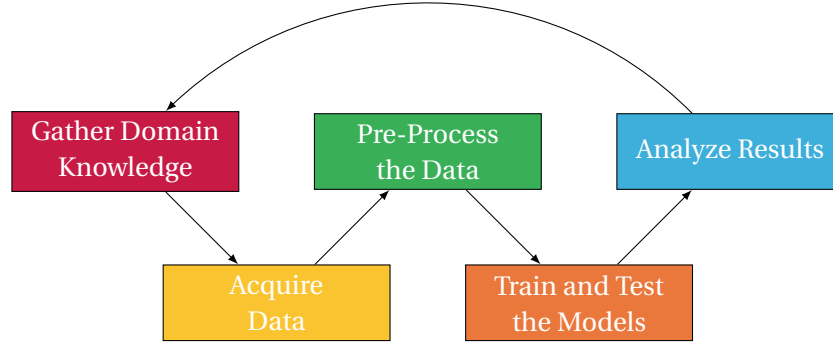


Figure 3.1: Building blocks and flow diagram to represent the proposed approach for impact position detection.

With a clear picture of the piezoelectric transducer and plate interaction, it is possible to propose a methodology to develop an ML strategy for impact source location. The second step is to acquire a dataset of impact signals. For this purpose, an automated experimental setup is developed and two different impact generation systems are evaluated. The third step involves finding and evaluating the best representation of the data. The goal is to minimize the number of features (i.e. the number of inputs) without losing resolution on the prediction of the position. Two different signal transformations are evaluated.

The fourth step is to train different ML and DL models with the acquired dataset. Four models are chosen and for each of them, the predicted value is compared with the known position label to evaluate their performance.

The last step is to analyze the results and define the next steps to improve the prediction error. The proposed methodology is an iterative process, so once the first iteration is complete, the loop goes back to re-designing the acquisition and signal representation strategies.

3.2.1 Domain Knowledge: Piezo-transducer and Plate Coupling

Prior to exploring the impact position detection task, it is critical to understand the interaction between a piezoelectric transducer and a thin plate. Because of the easy integration and the dual status as sensor and actuator, piezoelectric transducers are chosen to acquire the vibrations that occur on a surface and also to excite vibrational modes on it. Because of that, it is key to understand the mechanical interaction of a piezoelectric patch glued to a thin plate.

Consider a piezo-ceramic bonded to the surface of a thin plate. If this structure is thin, compared to the bending radius, and if the size of the piezo is small, compared to the total area of the plate, it can be assumed that the deformation of the surface is not affected by the presence of the piezoelectric patch. Thus, the motion of the surface can still be described by Kirchhoff's plate theory (which is described in Section 2.7). The coupling between the piezo-ceramic and the plate can be described in terms of an internal bending moment.

The transducer can operate in two modes. The first one is sensing, which will be exploited in this chapter. The second mode is actuation, which will be exploited in the following chapters (Chapter 4, Chapter 5, and Chapter 6).

Piezoelectric Sensing Mode

When an impact occurs on a thin plate, flexural waves propagate on its surface. Due to these waves, the surface of the plate is deformed over time. This deformation results in stress inside the piezo-ceramic that is bonded to the surface. According to the direct piezoelectric effect, this stress generates a voltage signal across the electrodes of the transducer. These signals are proportional to the amplitude of the out-of-the-plane deformation. In this way, the piezo actuator is used to acquire the vibrations on the surface.

Piezoelectric Actuator Mode

When a voltage is applied across the electrodes of the piezo-ceramic it will undergo uniform deformation (i.e. expansion in two axes and contraction in one axis). If the actuator is properly fixed to the surface of the plate and due to the bonding interaction, the actuator will generate stress on the surface of the plate. This stress will propagate in the thickness direction. Consequently, strain will occur. This stress can be modeled in the form of an effective bending moment and the generated deformation is equivalent to half the wavelength of a flexural wave that will propagate on the surface of the plate. The coupling effect is illustrated in Fig. 3.2. Note that if the polarity of the applied voltage is inverted the deformation direction is reversed.

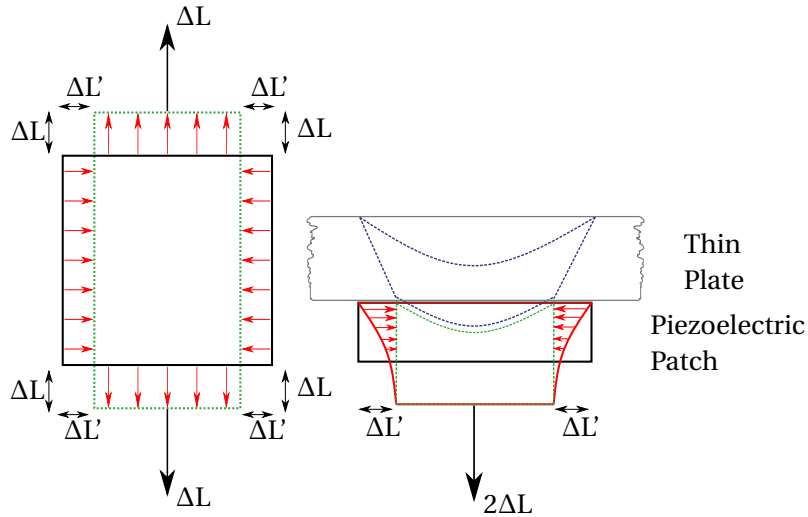


Figure 3.2: Graphical representation of the mechanical interaction of a piezoelectric patch and a thin plate. (LEFT) Unconstrained piezoelectric element deformation in one plane. (RIGHT) Deformation on the plate induced by the deformation of the piezo-patch or vice-versa (cross-section view).

Optimal Piezo Thickness

The strain-stress interaction between the piezo and the plate can also be represented in terms of an internal bending moment. In [95], Kim *et.al.* proposed an analytical model to represent the effective bending moment in terms of the mechanical properties of the piezo and the plate. The model is presented in Eq. 3.1. This model assumes that a pair of piezo-actuators are perfectly bonded (i.e. bonding layer with 0 mm thickness) to the upper and lower faces of a thin surface, and takes advantage of the symmetry plane in the middle of the composite structure. Certain assumptions are made to simplify the problem. For instance, linear strain distribution, equal strains in the x and y directions, and a homogeneous plate are considered.

$$m_x = m_y = \frac{\rho_z (2 + \rho_z)}{1 + \beta \rho_z (3 + \rho_z^2 + 3\rho_z)} \frac{h^2}{4} \gamma \Lambda, \text{ where} \quad (3.1)$$

$$\rho_z = \frac{2t_z}{h}, \quad \beta = \frac{1 - \nu_p}{1 - \nu_z} \frac{E_z}{E_p}, \quad \gamma = \frac{E_z}{1 - \nu_z}, \quad \Lambda = \frac{V}{t_z} d_{31}.$$

In Eq. 3.1, the subscript p refers to the plate and the subscript z refers to the piezo-actuators. ρ_z is the non-dimensional piezo to plate thickness ratio, where h and t_z are the plate and piezo thickness, respectively. β represents a non-dimensional material property ratio and γ is the piezo-actuator stress-strain ratio, where E and ν are the Young modulus and the poison ratio. Finally, Λ is the actuator free piezoelectric strain, where V , is the voltage across the electrodes of the piezo and d_{31} is the piezoelectric coefficient.

Using this equation, it is possible to plot the effective bending moment m_x for different values of the non-dimensional thickness ratio ρ_z . Thus, one can find the optimal thickness of the piezo actuator for different substructure materials. The results for eight different materials are presented in Fig. 3.3.

Since steel has the highest bending moment of all the studied materials, it could be thought that a steel structure would exhibit the largest vibration amplitude. However, the actual vibration induced by the piezo-actuators depends on the ratio of the effective moment to the bending stiffness (also called flexural rigidity) [95]. The bending stiffness is defined in Eq. 2.2 in Chapter 2. The expected displacement can be estimated by integrating the effective bending moments from the piezo actuator as the external plate loads of the Kirchhoff wave equation. Then, the wave propagation equation becomes:

$$D \nabla^4 u + \rho h \frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 m_x}{\partial x^2} + \frac{\partial^2 m_y}{\partial y^2}. \quad (3.2)$$

Fig. 3.4 shows the expected vibration amplitudes for the different substrates. Since the flexural rigidity of aluminum is approximately one-third compared to steel, the induced vibration is

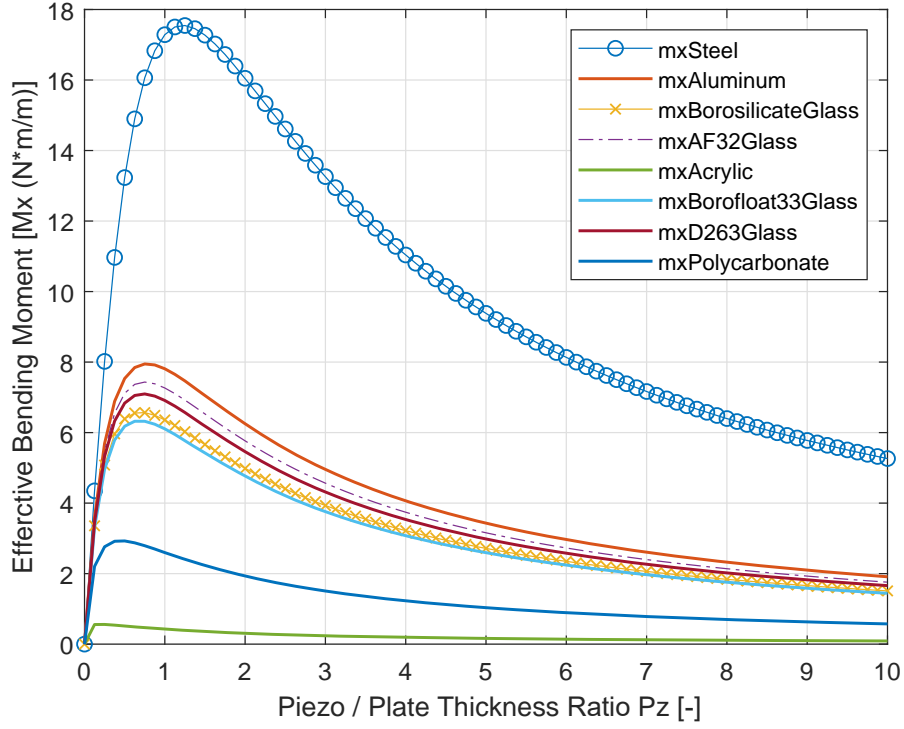


Figure 3.3: Effective bending moment vs. Non-dimensional piezo/plate thickness ratio. The maximum moment value represents the optimal thickness ratio.

slightly higher for the aluminum substructure.

From Fig. 3.3 and Fig. 3.4 it can be concluded that the optimal ρ_z for an aluminum structure is 0.9. Furthermore, the optimal thickness and expected vibration amplitude for aluminum and for the different glass structures are very similar. This makes sense because the moment-to-bending stiffness ratio is in the same order of magnitude.

The optimal thickness relation is more important for the actuation mode than for the sensing mode. This knowledge will be further explored in Chapter 6, during the development of a haptic surface demonstrator.

Minimal Achievable Spatial Resolution

Another important step is to understand the minimal wavelength for the flexural waves that propagate in the plate. This wavelength is directly linked to the highest spatial resolution of a localized peak.

The minimal achievable resolution can be modeled using the Kirchhoff theory of thin plates [96]. Assuming that the shearing of cross sections and the rotational inertia are negligible, and assuming the plate to be isotropic. Then, the flexural wavenumber, k_f , is defined as:

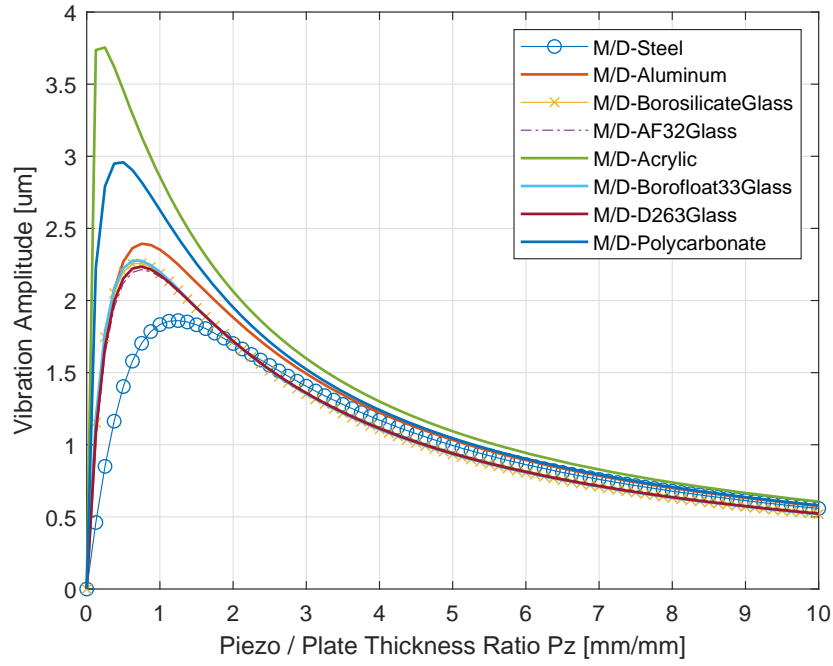


Figure 3.4: Expected vibration amplitude induced by a pair of piezo actuators bonded on a steel or an aluminum plate with thickness $h = 2\text{mm}$. Amplitude for different thickness ratios.

$$k_f = \left(\omega^2 \frac{\rho h}{D} \right)^{\frac{1}{4}}. \quad (3.3)$$

And the phase velocity, C_φ , which represents the relation between the angular velocity and the linear speed of propagation, can be modeled by:

$$C_\varphi = \frac{\omega}{k_f}. \quad (3.4)$$

By replacing the bending stiffness of the plate (Eq. 2.2) into the plate flexural wavenumber (Eq. 3.3) and then the resulting equation into the phase velocity equation (Eq. 3.4), one can obtain the phase velocity in terms of the plate mechanical properties and the circular frequency ω [96]:

$$C_\varphi = \left(\frac{E h^2}{12\rho (1 - \nu^2)} \right)^{\frac{1}{4}} \sqrt{\omega}. \quad (3.5)$$

The phase velocity can also be expressed in terms of the wavelength, λ , and the time period, T :

$$C_\varphi = \frac{\lambda}{T} = \lambda f = \frac{\lambda \omega}{2\pi} . \quad (3.6)$$

Then, the achievable wavelength is calculated by putting Eq. 3.5 into Eq. 3.6:

$$\lambda = \left(\frac{E h^2}{12\rho (1 - \nu^2)} \right)^{\frac{1}{4}} \frac{2\pi}{\sqrt{\omega}} . \quad (3.7)$$

Given that $\omega = 2\pi f$, then:

$$\lambda = \left(\frac{E h^2}{12\rho (1 - \nu^2)} \right)^{\frac{1}{4}} \sqrt{\frac{2\pi}{f_{max}}} . \quad (3.8)$$

The resulting model represents the minimal wavelength λ in terms of, f , the highest frequency mode that is excited on the plate, and the plate's mechanical properties.

When using the piezoelectric transducer as an actuator, half of this wavelength is the highest resolution that can be achieved for a localized peak. This spatial resolution limits the minimal distance between two adjacent fingers.

Optimal Location of the Piezoelectric Transducers

The location of the transducers plays an important role both in the sensing and actuating modes. Depending on the location, the piezo actuator will be able to excite in a better way certain resonant modes of the system. This occurs because the maximal amplitude of the vibration for a given mode occurs at the antinodes, while there is no displacement on the nodes. The location of the nodes and antinodes changes for the different eigenfrequencies.

In [97] different piezo patch shapes and locations were evaluated experimentally. It was confirmed that the location of the actuator strongly influences the ability of the actuator to excite certain modes as well as the excitation of undesired modes (what they call spillover). In [98, 99] an optimization algorithm was implemented in Matlab. Eight piezoelectric patches were located in different positions within a pre-defined area. Using FEA, the time reversal focusing was simulated and the maximum amplitude was validated.

The general recommendation is to distribute the piezoelectric patches avoiding any symmetries with respect to the axis of the plate. In this manner one piezo does not cancel the effect of the other in the actuation mode. Similarly, piezoelectric transducers in non-symmetrical locations will not retrieve redundant information. In the case of a single piezo, the guideline is to keep some space from the borders of the plate. In this way, the effect of reflection and the appearance of virtual sources will increase the quality of the peak reconstruction.

3.2.2 Experimental Setup and Data Acquisition

Machine Learning (ML) models and especially Deep Learning (DL) algorithms require big amounts of data. In particular, labeled datasets are needed to carry out the supervised training process. Seeing that, manual impulse generation processes that are used in [56] and [55] are not viable and need to be automated. This is why, a Linear Impact Generator (LIG) was developed. The goal is to obtain repeatable impulsive force stimuli in different locations of the touch surface.

Linear impactors were originally invented to simulate the head-to-head collisions that occur on American football when testing the protective headgear (i.e. Helmets) [100, 101]. The two principal methods that are used for this test are: The vertical drop tower, where a mass is released from a fixed height. And the pneumatic ram test, where a mass is linearly propelled at a pre-defined speed using a pneumatic actuator. Similar test methods are used in material science [102] to estimate the energy that is absorbed by a material when a fracture occurs, or in modal analysis to find the impulse response of a structure [103].

Inspired by these standardized methods, two versions were developed for the LIG. The first one is an electromagnetic actuator and the second one is based on a pneumatic actuator. Both are described in Section 3.3.

Experimental Setup: Aluminum Beam

To simplify the impact detection problem, the study will be made on a one-dimensional (1D) scenario. In this case an aluminum beam is chosen. A beam is a special case of a plate where the width is much shorter compared to the length ($l \gg w$). In this particular case, the geometry of the plate behaves as a wave-guide where the wave propagation and reflection mainly occur in the length direction. So, it can be assumed that the wave propagation occurs in 1D and the surface motion at the center-line of the plate will represent the wave propagation phenomena.

This experimental setup is also used for the peak generation experiments using Deep Learning (Chapter 4 and Chapter 5). Particularly, in these experiments, several scans of the surface vibration are carried out. Having a 2D surface (regular plate) would have taken more time than studying a single line on a beam.

To develop the experimental setup, a $7 \times 7 \times 0.2 \text{ mm}^3$ piezo-ceramic transducer (Steminc - SMPL7W8T02412WL) was bonded to a $250 \times 16 \times 2 \text{ mm}^3$ ($l \times w \times h$) aluminum beam (Aluminum AW-6082) using epoxy glue. The piezoelectric transducer was placed at position $x_b = 62.5 \text{ mm}$ (in the length direction) and $y_b = 8 \text{ mm}$ (in the width direction). The beam is fixed to a pair of solid aluminum blocks using double-sided tape. To avoid dampening the vibrations and allow wave reflection, the thinnest available tape ($200 \mu\text{m}$ thick) is chosen. The fixing is symmetrical and the bar is supported over a length of 2 mm. The single-piezo aluminum bar experimental setup is illustrated in Fig. 3.5.

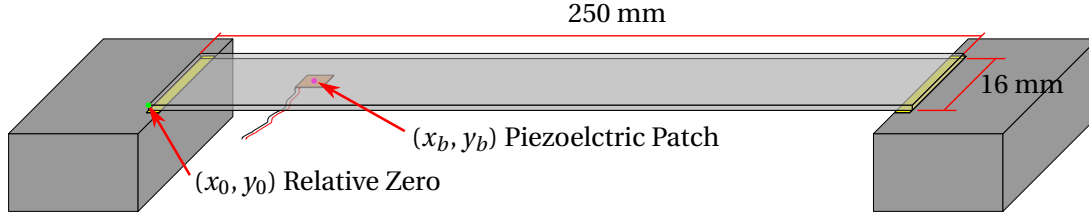


Figure 3.5: One-dimensional (1D) wave propagation experimental setup. A single piezoelectric transducer is bonded to an aluminum beam which supported on two solid aluminum blocks.

Acquisition System Automation

To automate the dataset acquisition, the LIG is mounted on a 3-axis CNC table (Stepcraft D600). The signals from the piezoelectric actuator are acquired using an oscilloscope (Lecroy LT224). Finally, to control the acquisition process, an automation script was implemented in Python. The Python program can trigger the single-impact sequence on the LIG and communicates with the oscilloscope. An overview of the system is presented in Fig. 3.6.

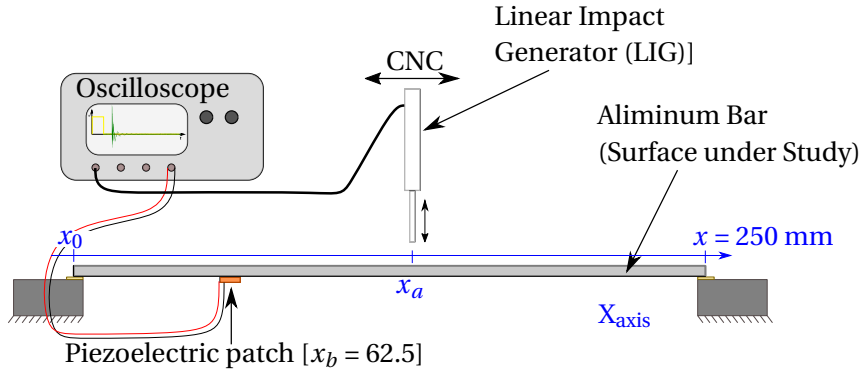


Figure 3.6: Experimental setup for automated mechanical impact signal acquisition.

The system operates as follows: First, the area of interest is defined. Then, the spatial resolution is fixed. The spatial resolution defines the distance between two adjacent impact locations. An $N \times M$ array of coordinates for the measuring points is defined. When the acquisition starts, the system moves the LIG to the first coordinate (x_{start}, y_{start}) . At this point, the computer triggers a single-impact over the surface of the aluminum beam. After each impact, flexural waves propagate on the surface and diverge from the contact point. The piezo patch is used to convert the stress signal into an electric signal. The trigger signal also starts a single acquisition on the oscilloscope, which records the voltage across the electrodes of the piezo-ceramic patch. The software retrieves the signal from the oscilloscope and stores it in an array. Then the system moves to the next location. The acquisition procedure is repeated $N \times M$ times, all across the surface.

The automatic acquisition system is used throughout the rest of the thesis. More details are given on each chapter.

Data Acquisition

The impact data is acquired using the automated LIG. For this reason, the aluminum beam experimental setup is mounted on the CNC table and the relative zero is set at the lower left corner of the aluminum beam, as shown in Fig. 3.5.

The study region was arbitrarily chosen with no significant effect on the results. The impact acquisition starts at position $x_{start} = 63$ mm, $y_{start} = 8$ mm and ends at position $x_{end} = 240$ mm, $y_{end} = 8$ mm. The spatial resolution is set to 1 mm, which leads to 177 unique locations. At every position, 30 samples (i.e. single-impact repetitions) are acquired. As a consequence, the dataset contains 5310 impacts. The position y_a is kept constant during the whole study ($y_a = 8$ mm, which corresponds to the mid-line of the beam). The study region and experimental setup are illustrated in Fig. 3.7.

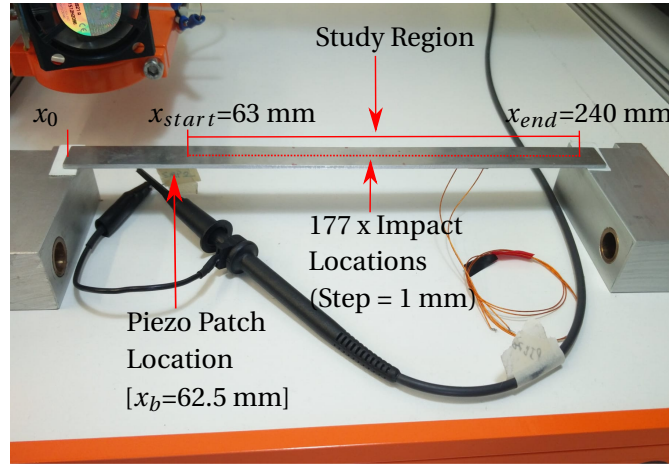


Figure 3.7: The aluminum beam and the aluminum blocks are mounted on the CNC table using double-sided tape. The piezoelectric patch is glued under the beam at position $x_b = 62.5$ mm. The study region goes from $x_{start} = 63$ mm and goes until position $x_{end} = 240$ mm.

Certain variables cannot be controlled in this experimental setup. Such as the temperature of the room and external vibrations. For this reason, the whole acquisition is divided into 5 different datasets. Each of them with a separation of 5 mm between consecutive impact locations and a 1 mm shift with respect to the previous acquisition group. Each subset is acquired at different times of the day. In this way, the variation of these external variables is contained in the data. This approach helps the ML model to generalize, which is further discussed in Section 3.2.4. The conditions for the acquired datasets are as follows:

- Dataset # 1: Starting at $x_{start} = 63$ mm with steps of 5 mm until $x_{end} = 238$ mm.
- Dataset # 2: Starting at $x_{start} = 64$ mm with steps of 5 mm until $x_{end} = 239$ mm.
- Dataset # 3: Starting at $x_{start} = 65$ mm with steps of 5 mm until $x_{end} = 240$ mm.
- Dataset # 4: Starting at $x_{start} = 66$ mm with steps of 5 mm until $x_{end} = 236$ mm.
- Dataset # 5: Starting at $x_{start} = 67$ mm with steps of 5 mm until $x_{end} = 237$ mm.

3.2.3 Data Pre-processing

One of the key steps in the ML pipeline is to prepare the dataset. This refers to pre-processing the data with the goal of finding the best representation of the features contained in the raw signals. Apart from acquiring a good-quality dataset, the representation of the input data has the biggest influence on the ML models' prediction error.

The recorded impact signal is a time series that contains the changes in the voltage of the piezo in time. This time series can be represented as a one-dimensional vector, where the length of the array is equal to the number of voltage samples. The easiest approach to predict the position where an impact occurs would be to feed this vector to the ML model. If that was the case, each time sample will be entered as an input feature for the ML model. Nonetheless, slight variations in the time when the impact has occurred would lead to a wrong prediction. For example, if the impact is delayed by a couple of milliseconds with respect to the trigger signal, the impact signal will shift in time.

An interesting solution to this issue comes from domain knowledge of the impact signals. As discussed in Chapter 2, flexural waves are dispersive when propagating in solid media. This means that the propagation speed changes for the different frequencies. This effect is modeled by the phase velocity equation (Eq. 3.4) presented in Section 3.2.1. In addition, as described in Section 3.2.1, the piezo-transducer and the plate are bonded and the wave propagation can be described by Kirchhoff's plate theory. As a consequence, the resonant modes of the system are related to the frequency content of the signal on the piezo.

Based on this knowledge, there are two signal representations that are interesting.

- The first one is a data transformation that can extract the changes in the frequency in time. A spectrogram is a visual representation of a signal. It makes use of the Fast Fourier Transform (FFT) and window analysis to represent the spectrum of frequencies as it varies in time. For a spectrogram, each pixel contains the intensity of a given frequency in a given window of time.
- The second one is a data transformation that can extract the frequency content of the signal. The magnitude of the FFT is a simpler transformation of the time-domain signal, which contains the intensity of each frequency on the impact signal. In this case, there is no information about the variation in time, which could lead to a more "distilled" representation to inform about the location where the impact occurred.

The effect of these representations on the ML models is studied in Section 3.3, and the prediction results for both representations are compared.

Impact Data Transformation

Depending on the chosen representation, the acquired impact data is transformed into a dataset of spectrograms or a dataset containing the magnitude of the FFT for each impact signal.

Each matrix is reshaped into a single column by stacking each column on top of each other. As a result, a 1D vector is obtained. After all transformations are carried out, the dataset is created by appending, side by side, all the transformed vectors. Each data point is labeled with the position where the impact was generated and the number of the sample.

Data Analysis

The final step of the data pre-processing is to run a preliminary evaluation of the proposed representation. One of the most effective methods to do this is the Principal Component Analysis (PCA). This statistical model transforms the data into a lower-dimensional version with no physical meaning. The final set of orthogonal components (i.e. vectors) is a linear combination of the original data [69]. In other words, PCA is used to find a lower-dimensional representation of the data. When the number of components is set to 2 or 3, it is possible to get a visual representation of the distribution of the data. This intuitive visualization gives an idea of the separability of the data, thus the effectiveness of the chosen representation.

3.2.4 ML and DL Models, Training, and Testing

The impact position detection task is considered a regression problem. In this manner, it is possible to obtain a continuous prediction of the impact position that allows the model to generalize to other locations. In other words, the model can infer the location of impacts that were not observed during the training stage.

ML Models and Neural Networks

To predict the impact location two machine learning algorithms and two deep learning models are selected. This allows to test and compare different approaches to process the data. For machine learning the simplest model and one of the most advanced models are chosen. For deep learning, two common architectures with different processing layers are explored.

First, the linear regression model from Scikit-learn [104]. Second, the XGBoost-Regressor which is an implementation of the gradient-boosted decision trees [105]. Third, a vanilla Neural Network (NN) [69]. This NN has 4 hidden layers with ReLU activation, batch normalization, and Gaussian noise to improve the generalization. The output layer of the NN has a linear activation function. This layer defines the model behavior as a regression problem. Lastly, a 2D Convolutional Neural Network (2D CNN). The 2D CNN has four convolutional layers, the first two layers with a depth of 32, and the second two layers have a depth of 64. The 4 layers have ReLU activation. MaxPooling is used to reduce the dimension of the input and a dropout of 0.5 which helps to improve the generalization of the model. The output of the convolutional layers is flattened and passed to a fully connected NN with one hidden layer. The hidden layer has ReLU activation and an output layer with linear activation. More details on the architecture for the NN are presented in Appendix A.

Data Preparation

To help the models to generalize, one can profit from the separated subsets. One portion of the data is used to train the model and another portion is used to test the models. Thus, the data is separated into two main groups, the train and validation set, and the test set. The first group contains the data to train the model and to validate the error after each training iteration. This group is created by merging datasets # 1, # 2, and # 3. Then randomly shuffling the data and splitting it into two subsets, 80% of this merged set is used for training, and 20% is used for validation. The second group is used to test the performance of the trained models. The test error measures the generalization of the model. In other words, how well the models perform with data that have never been observed before. For this purpose, the datasets # 4 and # 5 are used independently.

Training and Testing

The linear regression and XGBoost-Regressor models are fit using the training portion of the data. Then, the trained model is used to predict the impact location for the validation data and the Test data. The Mean Absolute Error (MAE) is used to measure the error between the real position (i.e. known dataset labels - y) and the predicted value (\hat{y}). The MAE is also known as the L_1 -norm loss function and it is presented in the following equation:

$$\text{MAE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} |y_i - \hat{y}_i| . \quad (3.9)$$

The two NN are implemented using TensorFlow [106]. Both models use the MAE as the loss function. The NN weights are updated using the Adam optimizer [107], which is more powerful and efficient than the classical stochastic gradient descent. To accelerate the training, a high learning rate ($Lr = 0.001$) is used during the first 100 epochs. Then a lower learning rate ($Lr = 0.00001$) is used through 900 epochs.

After every training batch, the validation dataset is used to evaluate the progress of the model. When the training phase is completed, the test datasets (# 4 and # 5) are passed through the model and the MAE is calculated.

3.3 Impact Position Detection Experiments

3.3.1 Impact Position Detection: Iteration 1

The first version of the Linear Impact Generator (LIG v1) is the electromagnetic LIG and it is called LIG v1. It is a relatively simple linear actuator that consists of a dropping mass controlled by an electromagnetic actuator.

Construction and Working Principle of the LIG v1

Let an object with a known mass m be suspended at a defined height h_0 over a surface. When this body is dropped, its motion can be described by the "*Newtonian classical mechanics*" bouncing ball problem. If the effect of air friction is ignored, then the instantaneous speed right before it makes contact with the surface is equal to: $V_f = \sqrt{2gh_0}$, where g is the acceleration due to gravity at the place of the experiment. Now, considering that the impact occurs perpendicular to the surface. The impact energy is equal to $U_i = mgh_0 - mgh_b$, where h_b is the maximum height reached by the body after the first bounce. This height will principally depend on the material properties and geometry of the body and is commonly represented by the coefficient of restitution.

If the parameters are kept constant (m , h_0 , and g constant) and the body is re-grabbed after the first bounce, one can expect that every time, the mass will impact the surface with the same amount of energy. In order to change the amount of energy, thus, the magnitude of the impact force, one can either modify the mass of the object or change the initial drop height.

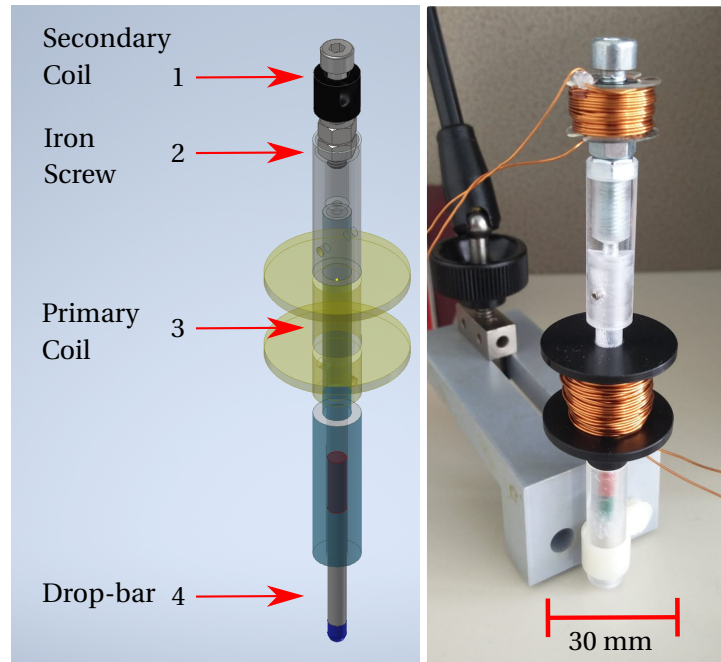


Figure 3.8: Linear Impact Generator (LIG). On the LEFT side the CAD model and on the RIGHT side the first functional prototype. Modified figure from [108] ©2019 IEEE.

The electromagnetic LIG is presented in Fig. 3.8. A magnet (red element) is attached to the upper end of a steel bar (gray element). This body, called the drop-bar, can move freely inside a tube made of acrylic. A pair of coils, operated with MOSFET transistors, are used to control the motion of the drop-bar. The operation principle is as follows: When the drop-bar (4) is at the lower position (i.e. in contact with the surface), the primary coil (3) generates a magnetic field to propel the bar upwards until it is attracted to an iron screw (2).

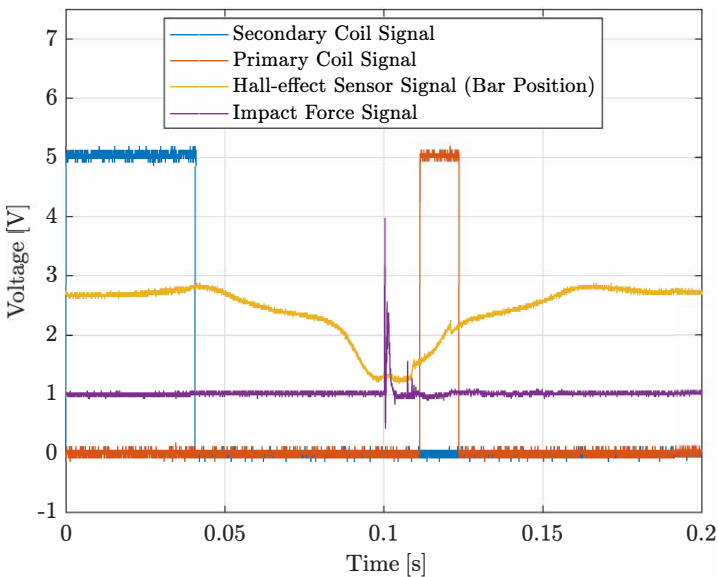
Chapter 3. Machine Learning for Impact Position Detection

At this point, the bar is magnetically stuck to the screw and it will rest in equilibrium until a triggering signal is received. When desired, the secondary coil (1) generates an inverse field in the screw which lets the bar drop in free-fall and impact the surface for a single time. After the first bounce, the primary coil is activated once again. This magnetic field provides the additional energy to return the bar to the upper position until it is attracted to the screw.

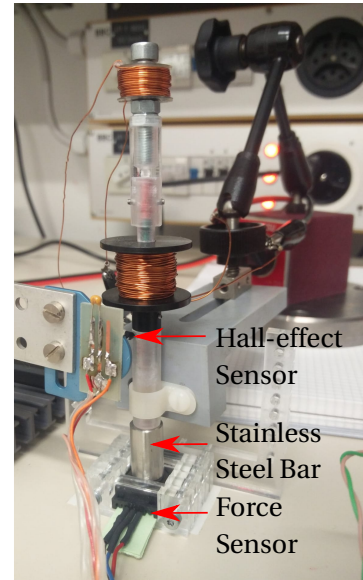
The MOSFETs are driven by a microcontroller that constantly measures the position of the drop-bar. A hall-effect sensor, which is placed in the lower part of the actuator, is used for this purpose. The switching sequence is adjusted to ensure that a single bounce is achieved and that the bar always falls from the same height. Fig. 3.9a, displays a plot of the position of the bar (measured by the hall-effect sensor) and the switching sequence for the primary and secondary coils. The LIG single bounce sequence is triggered with a digital signal coming from the automated acquisition system described in Section 3.2.2.

Validation of the Impact Produced by the LIG v1

To validate the nature of the impact, a force sensor is used (compression load cell from TE Connectivity FS20). Fig. 3.9a presents the force signal that is acquired after a single impact and Fig. 3.9b illustrates the setup to measure the impact force. Note that a non-ferromagnetic stainless steel bar is placed on top of the force sensor. This is done because the surface of the sensor is a ferromagnetic material. By doing so the drop-bar does not remain magnetically attached to the sensor, instead, it bounces on the surface.



(a) Impact force sensing using a load cell in the surface to measure the impact signal after the bar is released.



(b) Side view of the set-up for the impact force signal acquisition.

Figure 3.9: Validation of the impact produced by the LIG v1. Modified figure from [108] ©2019 IEEE.

The force signal reveals that the impact occurs 100 ms after the drop signal is given. The force signal has an *Impulse-like* waveform, where a peak amplitude is achieved in a very short period of time. As expected, the contact duration is minimal. This duration directly affects the frequency content of the excitation induced on the plate (i.e. the quality of the impulse stimulus). If the contact duration is longer, some of the eigenmodes of the bar are damped and the voltage signal on the piezo would change drastically.

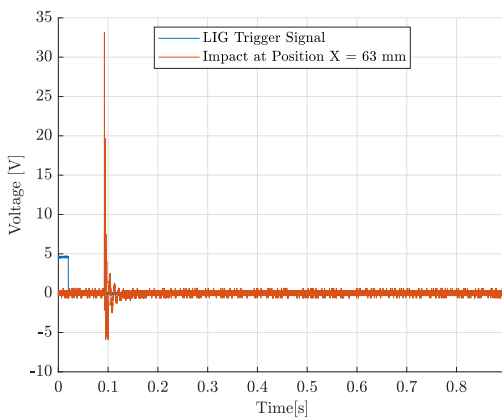
Data Acquisition LIG v1

The LIG v1 is used to acquire an impact dataset. The data is acquired as described in Section 3.2.2.

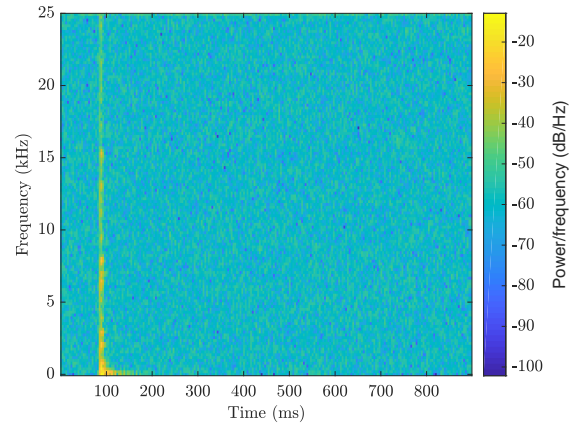
An example of the acquired impact signal is presented in Fig. 3.10a. To synchronize all the acquisitions of the dataset, the triggering signal of the LIG was also acquired. The impact signal appears at the instant $t = 100$ ms, as expected. Nonetheless, some variations were observed. The impact signal could shift in time by 10 ms. This can be explained by intermittent friction between the drop-bar and the acrylic tube.

Data Pre-processing LIG v1

In the first iteration, only the spectrogram transformation is used. The spectrogram function in Matlab is used to generate a spectrogram for each impact signal on the dataset. A window of 256 points, an overlap of 64 points, a fast Fourier transform window of 256 points, and a sampling frequency of 50 kHz are defined. As a result, a matrix with 128×234 "pixels" or features is obtained. An example of the spectrogram transformation for an impact signal is displayed in Fig. 3.10b.



(a) Single impact acquisition for an impulsive force in position $x_a = 63$ mm.



(b) Spectrogram transformation for a single impact with the LIG v1.

Figure 3.10: Impact signal and spectrogram transformation at location $x_a = 63$ mm. Impact created with the LIG v1. Figure from [109] ©2019 IEEE.

Chapter 3. Machine Learning for Impact Position Detection

As expected, at $t = 100$ ms there is a high concentration of all of the frequencies. Because of dispersion, it can be observed that the lower frequencies take longer to dissipate. Each spectrogram matrix is reshaped into a single column, thus a 1D vector with length 29952 is obtained.

The Scikit-learn Python module [104], which implements the randomized truncated singular value decomposition (SVD) method by Halko *et al.* [110], is used to obtain the PCA in 3D. The PCA is fit to reduce Dataset # 1 into the three main components and plotted on a three-dimensional (3D) figure. The results are presented in Fig. 3.11. The results show a promising data separation where the 30 samples from each location agglomerate in a specific region of the 3D plot. Even if the separation is not perfect in 3D, this indicates a high chance of obtaining a good regression model, especially when using all the features on the data representation (29952 components).

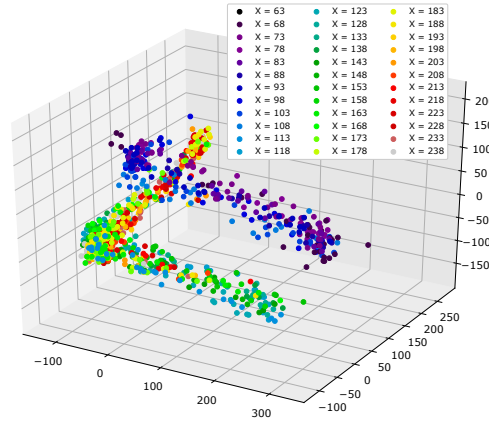


Figure 3.11: PCA 3D for Dataset # 1. Graphical representation of the reduction to the 3 principal vectors (i.e orthogonal components with the highest variance), each set of acquisitions at a specific position is represented by one unique color. Figure from [109] ©2019 IEEE.

Model Training and Results LIG v1

The two ML and two DL models are trained as depicted in Section 3.2.4. Table 3.1 presents the results obtained for the four models during the training/validation phase and the testing phase. For the training step, the validation set of data is used to evaluate the error. And during the testing phase, the test datasets # 4 and # 5 are used. All the error values come from the MAE (in mm) and the percentage of error is calculated considering the length of the working area of 177 mm.

The best models are the XGBoost and the 2D CNN. The XGBoost has the best results for the particular training dataset (merge of data # 1, # 2, and # 3) but it does not have a good generalization when evaluated in the test data (Data # 4 and # 5), which contain unseen data points. Similarly, the 2D CNN presents good results for the training data. Nonetheless, it also presents low and constant error values for both test datasets.

Table 3.1: Validation and test MAE [mm and %] for LIG v1 Data.

| LIG v1 - Spectrogram dataset | | | | |
|------------------------------|--------------|--------------|--------------|-------------|
| Error | LinearReg | XGBoost | NN | 2D CNN |
| Validation | 19.5 / 11.2% | 7.6 / 4.3% | 14.9 / 8.4% | 8.1 / 4.6% |
| Test # 4 | 22.2 / 12.6% | 12.4 / 7.0% | 21.6 / 12.2% | 15.7 / 8.9% |
| Test # 5 | 31.7 / 17.9% | 20.4 / 11.5% | 26.9 / 15.2% | 14.8 / 8.4% |

This means that the 2D CNN has better generalization. It is important to highlight that during the training phase, the 2D CNN presented an MAE of 13 mm while the validation MAE is 8 mm, this shows that the model still needs more exploitation and deepening. The fact that the training error is above the validation error, means that the model needs more data to reduce the error while maintaining a high generalization.

Generally speaking, 2D CNN presents better results because they use kernels to find the 2D relationships of data. The kernels are able to find patterns that are shifted in time. On the other hand, the other models which are more sensitive to time variations, present higher errors and less generalization. Nonetheless, they are predicting the impact position with an error below 20%. This is an early indicator of the robustness of the ML approach for impact position detection. The results are not yet acceptable and do not overpass the state-of-the-art results so a second iteration is required.

3.3.2 Discussion

After the first iteration, there is big room for improvement, especially in the quality of the impact data. When analyzing the spectrogram representation of the impact signal (Fig. 3.10b), it is clear that the drop-bar is mostly stimulating the low-frequency modes of the bar. In addition, the energy dissipates quite fast which also indicates that the LIG is not giving a big amount of energy to the system. The principal reason for this can be the contact duration between the bar and the surface. If the bar remains in contact with the surface, the plate will not reverberate freely and the impact would not resemble a perfect impulse.

To validate this, a high-speed camera (Photron FASTCAM Mini AX100 recording @ 8400 FPS) was used to track the displacement of the drop-bar. After analyzing the frames, it was found that the drop-bar is not bouncing immediately, but it remains in contact with the aluminum beam for 4 ms. This time varied in the order of ± 1 ms. This variation is again explained by the friction between the drop-bar and the acrylic tube.

The LIG v1 does not allow to control the contact duration, $T_{contact}$, so it is not possible to change this value to study its effect on the prediction of the position. In the following sections of this chapter, the contact duration will prove to be an important factor for the precision of the impact position detection.

To fully understand the use of ML on the impact position detection task, it is important to study different types of impacts, especially, if the desired goal is to obtain an impact position detection system that can be used to detect finger taps on a touch surface. In reality, when humans interact with a surface, the contact duration between the fingertip and the surface varies.

For the second version of the LIG with the second iteration of the ML approach, the objective is to improve the quality of the acquired impact signals and to find a better representation of the impact data. The quality of the impact signals and the way that the data is pre-processed can bring important improvements on the training and testing error. At the same time, it is interesting to study if the contact duration has an effect on the precision of the impact position detection.

For these reasons, an improved version of the LIG is developed. The new impact generator (LIG v2) creates an impact that is not dampening the natural modes of the bar and enables the control of $T_{contact}$.

3.3.3 Impact Position Detection: Iteration 2

Based on the knowledge from the first iteration and inspired by the linear impactors mentioned in Section 3.2.2, a pneumatic actuator was chosen to generate the impacts for the second version of the LIG. The functional prototype for LIG v2 and the pneumatic system diagram can be seen in Fig. 3.12.

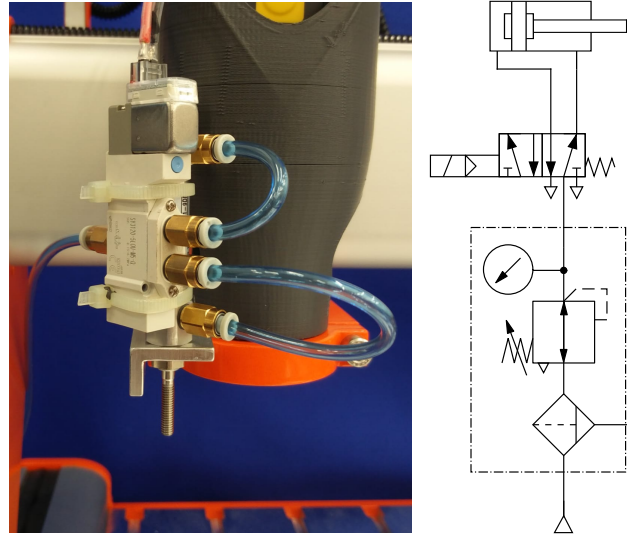


Figure 3.12: The new version of the Linear Impact Generator. (LEFT), functional prototype LIG V3. (RIGHT), Pneumatic system diagram. Figure from [111].

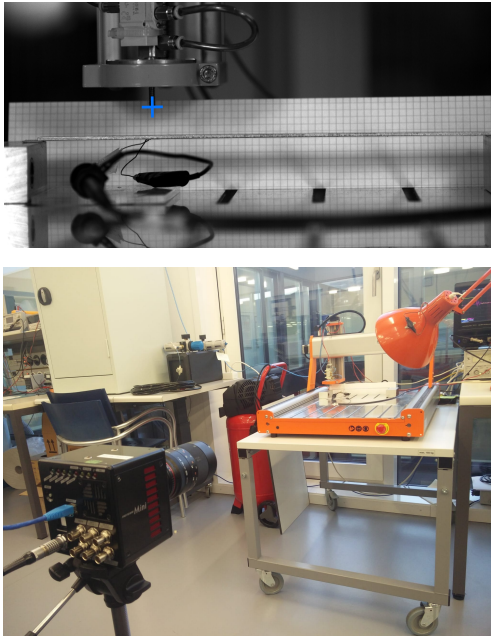
Construction and Working Principle of the LIG v2

An air cylinder (SMC - CDJ2D10-15Z-B) driven by a mono-stable 5/2 solenoid valve (SMC - SY3120-5LOU-M5-Q) is used to impact the surface. An arbitrary signal generator (TG 2512A), which can be remotely controlled (via Ethernet), creates the signal to control the solenoid valve. The duration of this signal sets the period during which the impactor stays in the lower position, thus the contact duration, $T_{contact}$. The signal generator is controlled by the Python automation script described in Section 3.2.2.

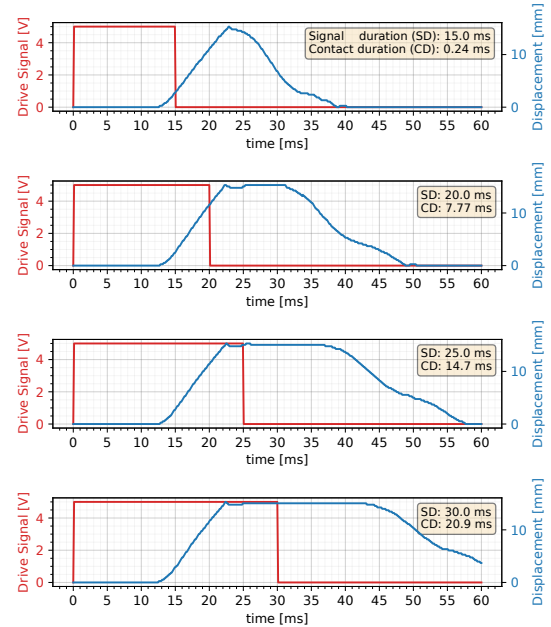
Validation of the Impact Produced by the LIG v2

The contact time is validated using the same high-speed camera, and the validation experiment is displayed in Fig. 3.13a. After analyzing the frames of the video, one can track a reference point in the moving part and obtain the displacement profile for the tip of the impactor.

By changing the length of the square pulse that drives the solenoid valve, it is possible to modify the contact duration, $T_{contact}$. Different types of impacts are presented in Fig. 3.13b. The four graphs present the displacement profile for the LIG after different pulses are used to drive the solenoid valve. As a consequence, four different impacts, with different $T_{contact}$, are created.



(a) High-speed camera experimental set-up for the analysis of the displacement profile for the tip of the LIG v2. The blue cross is tracked.



(b) Displacement profile for the tip of the Pneumatic LIG when 4 different driving pulses are used to drive the solenoid valve.

Figure 3.13: Validation of the impact produced by the LIG v2. Figure from [111].

The LIG v1 achieved a contact time of approximately 4 ms. The new version can go down to a $T_{contact}$ of 0.24 ms. Also, the signals acquired with the new actuator are more consistent among several repetitions.

Data Acquisition LIG v2

The LIG v2 is used to acquire a second impact dataset. The acquisition procedure is described in Section 3.2.2. The signal sampling rate was 250 kHz instead of 50 kHz. This modification allowed to capture a higher frequency bandwidth and to recover important spectral content that was neglected in the first iteration. For example, the frequency components around 42 kHz which present the highest magnitudes on the spectral content (see Fig. 3.17b). An example of a single acquisition, with the new sampling conditions, is presented in Fig. 3.14.

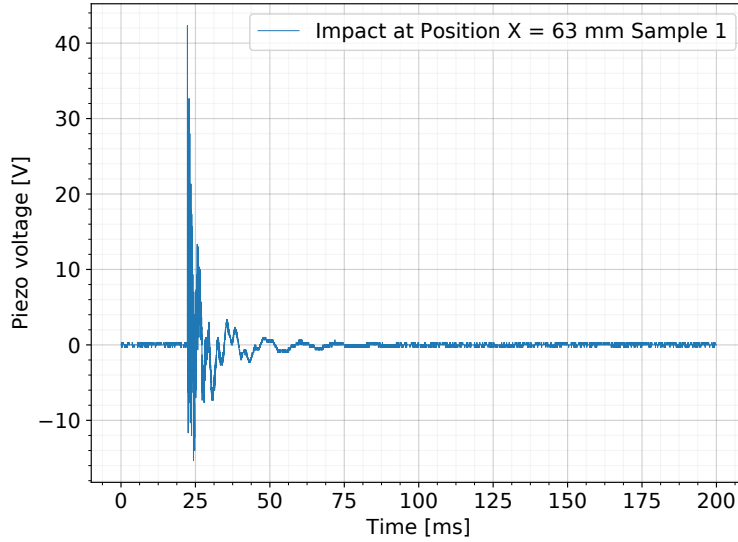


Figure 3.14: Example of a single stress signal acquisition for an impact induced at position $x_a = 63$ mm. Figure from [111].

To train the models and to test the effect of the contact duration four types of impacts are defined. The displacement profile for each type of impact is presented in Fig. 3.13b. The reference $T_{contact}$ was 0.24 ms. This type of impact is closer to a perfect impulse and is obtained when a drive signal of 15 ms is sent into the solenoid valve. The dataset obtained for this type of impact is used to train and test the model (Section 3.3.3).

Three additional $T_{contact}$ are defined, 8 ms, 15 ms, and 21 ms. During the acquisition of each group of 5 datasets $T_{contact}$ remains constant. The three additional datasets are used to test the effect of the contact duration on the impact position detection error. This effect is studied in Section 3.4.

Frequency content study LIG v2

To gain further knowledge on the nature of impact signals, it is interesting to visualize the frequency content for impact signals at different locations. Fig. 3.15 presents the magnitude of the FFT for six different impact locations ($x_a = [63, 68, 73, 78, 83, 88]$).

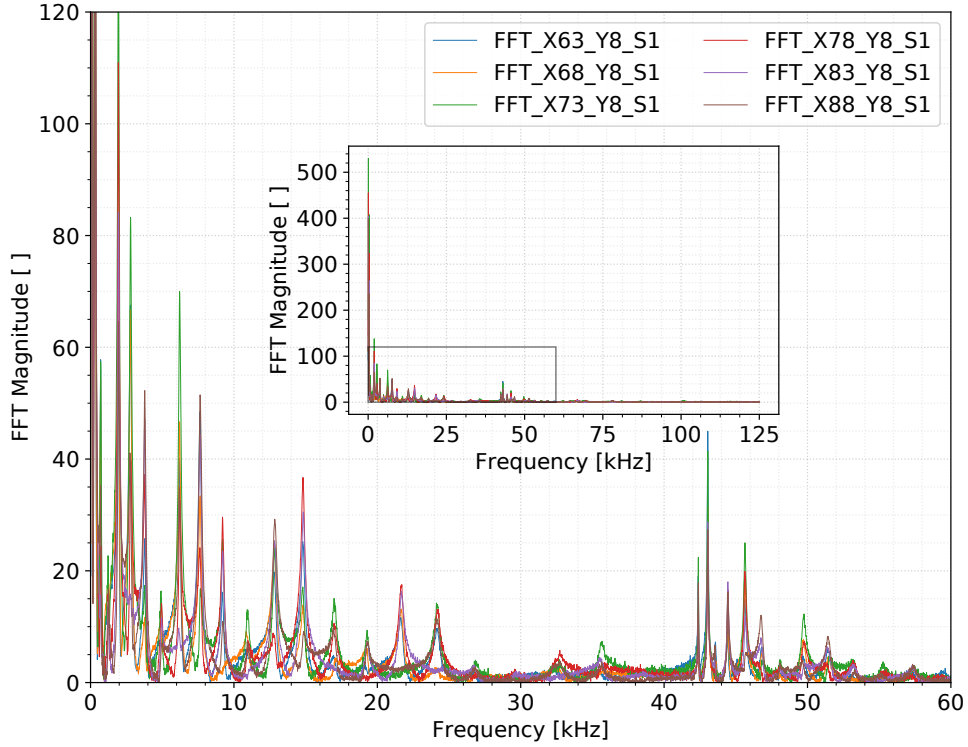


Figure 3.15: Frequency content for different impact locations. The signals are obtained from the magnitude of the FFT for each impact signal.

The main figure contains the most relevant frequency range and a sub-figure presents a zoom-out where the magnitude of the FFT can be seen for all frequencies. It is evident that the frequency values remain constant. Nonetheless, the magnitude of each frequency component varies for every impact location. As discussed in Chapter 2, these frequencies are related to the resonant frequencies of the system. This finding is crucially important since it inspires the use of the magnitude of the FFT as an alternative representation to the spectrogram.

Another key information from the frequency analysis is that the sampling rate plays a key role in signal acquisition. On the first iteration, the signals were sampled at 50 kHz. According to the Nyquist-Shannon sampling theorem, the sampling frequency should be at least two times bigger than the maximal frequency to be acquired.

This is why on the first iteration the highly relevant frequencies around 42 kHz were lost. To understand the origin of these frequency components, an impedance analysis of the system can be carried out. Using a precision impedance analyzer (Agilent 4294A), a swept sine wave is sent into the piezo actuator on the aluminum beam. At the same time, the magnitude and phase for the current and voltage on the piezo are measured. Based on this value, it is possible to find the complex electrical impedance as a function of frequency. Fig. 3.16 presents the magnitude of the impedance for different frequencies. From this analysis, it was found that the highest resonant effect appears around 42 kHz. Thus, this frequency is associated with the electrical resonance of the whole system (i.e. the piezo and bar coupled). Note that the resonant frequency of the piezo patch, when it is not bound to the aluminum bar, is around 200 kHz.

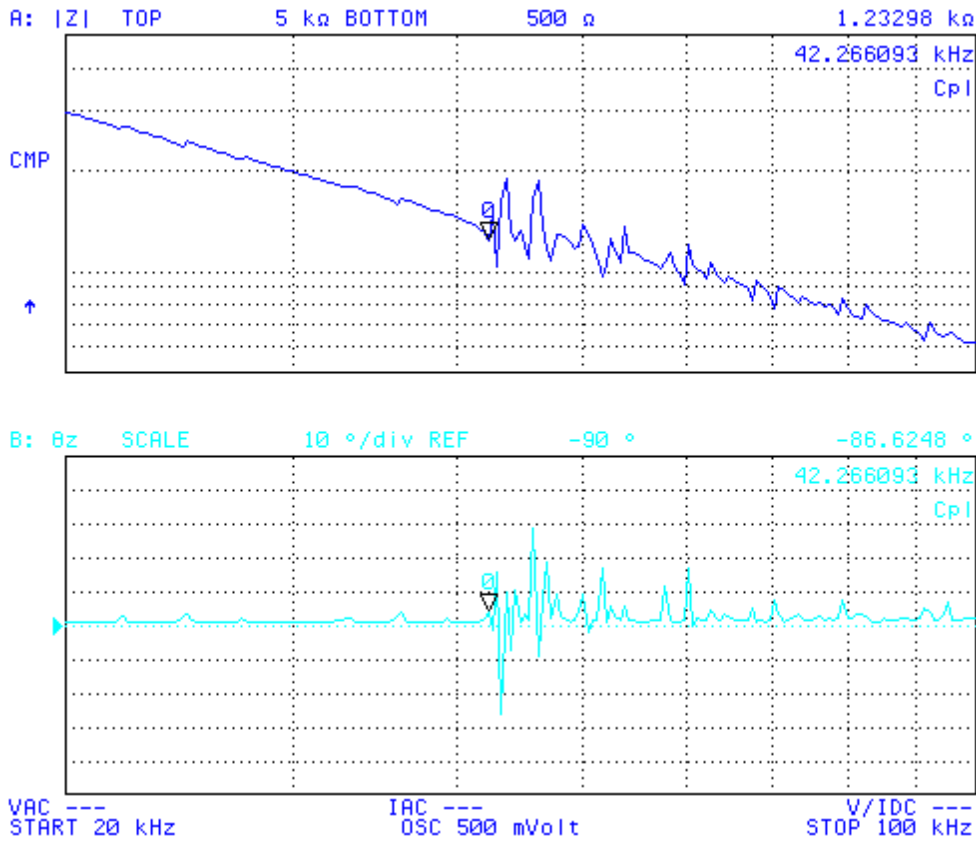


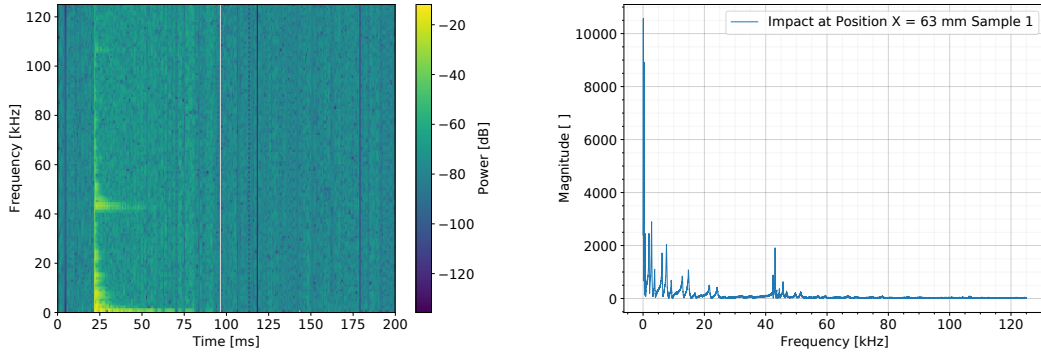
Figure 3.16: Impedance analysis of the single-piezo aluminum beam experimental setup. Sweep from 20 kHz to 100 kHz, the first resonance appears at 42.3 kHz.

Data Pre-processing LIG v2

For this iteration, two different transformations are used and their effect on the precision of the impact detection is evaluated. The results are presented later in this Section.

The first representation is the spectrogram. As in Section 3.3.1, the spectrogram function in Matlab is used. The only difference is the sampling frequency which is 250 kHz. The output of this function is a 128×260 "pixels" matrix, which is transformed into a single column vector of length 32280. As a reference, the spectrogram obtained from a single acquisition is presented in Fig. 3.17a.

The second representation is the magnitude of the FFT. In this case, the Python Numpy real FFT module (numpy.fft.rfft) [112] is used to transform the time-domain impact signal. After that, the magnitude of the complex number is calculated for each impact. The result of this transformation is a single-column vector of length 25000. All the signals are labeled with the position where the impact was induced and the sample number. As a reference, the magnitude of FFT transformation for an impact signal is presented in Fig. 3.17b.



(a) Spectrogram transformation for a single impact with the LIG v2. (b) Magnitude of the FFT for a single impact with the LIG v2.

Figure 3.17: Signal transformation for an impact at location $x_a = 63$ mm. Impact created with LIG v2. Figure from [111].

Once again, the PCA method is used to reduce the number of features from 32280 (for the spectrogram) or 25000 (for the magnitude of the FFT) to 3. The goal is to get a three-dimensional (3D) visualization of the dataset, which provides insight on the distribution of the data. The PCA is fit to reduce both the Spectrogram and FFT Magnitude of LIG v2 dataset # 1 into the three main components and plotted on a 3D figure. The results are presented in Fig. 3.18a and Fig. 3.18b.

As expected, the spectrogram dataset for LIG v2 impacts presents more clustered data (compared to the first iteration Fig. 3.11), which seems easier to separate or classify. What's more, the frequency-domain transformation leads to an even more separated representation in the 3D PCA plot. This confirms that this representation contains richer or more condensed information for the impact position detection task.

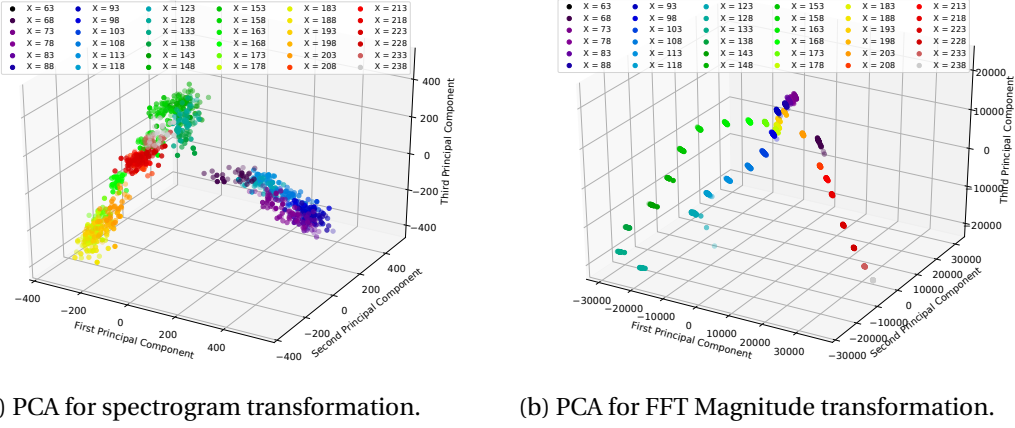


Figure 3.18: Three-dimensional graphic representation of the PCA dimensionality reduction, for dataset LIG v2 # 1. Each set of acquisitions at a specific position is represented by one unique color. Figure from [111].

Model Training and Results LIG v2

Following the approach proposed in Section 3.2.4, the prediction task is treated as a regression problem and the same 4 ML models are trained and evaluated.

For the spectrogram dataset, the four models were trained. Whereas for the magnitude FFT dataset, the 2D CNN was not used due to the structure of the data: The FFT dataset has changes along only one dimension (i.e. it is a single vector containing the magnitude for each frequency component), opposed to the 2D nature of the spectrogram dataset.

The results for the four models that were trained with the spectrogram dataset, are presented in the lower part of Table 3.2. While the results for the three models that were trained using the FFT dataset appear in Table 3.3. For all results, the error term refers to the MAE in millimeters, while the percentage (%) refers to the ratio between the error and the total length of the studied area (i.e. 177 mm). As a reference, the results from the first iteration (Section 3.3.1) are presented in the top part of Table 3.2.

As anticipated, the precision of the impact position detection increased with the new version of the LIG and the increased sampling rate. When observing the results for the models trained with the spectrogram dataset, all four models achieve a test error below 5%. Additionally, the NN which was not very performant in the first iteration, overcame the accuracy and generalization capabilities of the 2D CNN.

What's more, the frequency-domain data transformation led to a better prediction of the position. This was already expected after the PCA 3D visualization of the data. It appears that the simpler ML models (XGBoost-Regressor and Linear Regression) performed better with this transformation of the data.

3.4 Effect of the Impact Contact Duration

Table 3.2: Validation and test MAE for the four models [mm and %].

| Iteration 1 Results (LIG v1) - Spectrogram dataset | | | | |
|--|--------------|--------------|--------------|-------------|
| Error | LinearReg | XGBoost | NN | 2D CNN |
| Validation | 19.5 / 11.2% | 7.6 / 4.3% | 14.9 / 8.4% | 8.1 / 4.6% |
| Test # 4 | 22.2 / 12.6% | 12.4 / 7.0% | 21.6 / 12.2% | 15.7 / 8.9% |
| Test # 5 | 31.7 / 17.9% | 20.4 / 11.5% | 26.9 / 15.2% | 14.8 / 8.4% |

| Iteration 2 Results (LIG v2) - Spectrogram dataset | | | | |
|--|--------------|--------------|--------------|--------------|
| Error | LinearReg | XGBoost | NN | 2D CNN |
| Validation | 6.42 / 3.63% | 0.81 / 0.46% | 2.26 / 1.28% | 3.45 / 1.95% |
| Test # 4 | 9.19 / 5.19% | 2.98 / 1.68% | 3.66 / 2.07% | 5.32 / 3.01% |
| Test # 5 | 8.85 / 5.00% | 4.85 / 2.74% | 4.68 / 2.64% | 7.43 / 4.20% |

Table 3.3: Validation and test MAE for the three relevant models [mm and %].

| Iteration 2 Results (LIG v2) - FFT Magnitude Dataset | | | | |
|--|--------------|--------------|--------------|--------|
| Error | LinearReg | XGBoost | NN | 2D CNN |
| Validation | 0.54 / 0.31% | 0.31 / 0.18% | 0.44 / 0.25% | N/A |
| Test # 4 | 1.73 / 0.98% | 2.01 / 1.14% | 6.50 / 3.67% | N/A |
| Test # 5 | 2.40 / 1.36% | 2.06 / 1.16% | 4.64 / 2.62% | N/A |

However, the three evaluated models achieve a test MAE lower than the average contact size of a fingertip (i.e. 6 mm << 10 mm which is the diameter of the average contact area of a finger with a surface), which proves their potential for finger touch detection.

The precision improvement is associated with the nature of the impact signals. The shorter $T_{contact}$ achieved by the new version of the LIG makes the impacts closer to a perfect impulse. A better impulse stimulus can stimulate a larger number of frequencies on the plate, which provides more precise information about the location of the impact.

As discussed in Section 3.3.3, the excited frequencies are directly related to the natural harmonics of the plate. These frequency values remain constant if the system does not change, only the magnitude for each frequency component changes with the location of the impact. The higher precision on impact position detection for the FFT confirms the hypothesis of the FFT representation being a better representation than the spectrogram.

3.4 Effect of the Impact Contact Duration

To test the effect of the contact duration, three additional spectrogram datasets were obtained. For each dataset $T_{contact}$ was fixed to a given value. The pre-trained ML models with the best results were chosen. In this case, the three groups of five datasets with different contact times were independently fed to the models. The predicted position values were compared with the original labels to obtain the MAE for each test dataset.

Chapter 3. Machine Learning for Impact Position Detection

Later, the average MAE and the percentage were estimated for each contact duration. Table 3.4 presents the error for each dataset when $T_{contact}$ was set to 7.77 ms. For the other contact duration values, only the average value on the 5 datasets is presented. This is done because the error values for the different datasets are very similar.

Table 3.4: Contact Duration Effect - Validation and test MAE for the three relevant models.

| Test of the Contact Duration effect - Spectrogram dataset | | | |
|---|----------------|----------------|----------------|
| Error - $T_{contact}$ [ms] | XGBoost | NN | 2D CNN |
| Validation - 0.24 | 00.81 / 00.46% | 02.26 / 01.28% | 03.45 / 01.95% |
| Test # 1 - 7.77 | 49.29 / 27.85% | 52.55 / 29.69% | 40.44 / 22.85% |
| Test # 2 - 7.77 | 49.10 / 27.74% | 51.10 / 28.87% | 41.53 / 23.46% |
| Test # 3 - 7.77 | 40.41 / 22.83% | 49.08 / 27.73% | 41.96 / 23.71% |
| Test # 4 - 7.77 | 38.82 / 21.93% | 51.25 / 28.95% | 42.52 / 24.02% |
| Test # 5 - 7.77 | 35.72 / 20.18% | 51.60 / 29.15% | 45.48 / 25.69% |
| Average - 7.77 | 42.67 / 24.11% | 51.12 / 28.88% | 42.39 / 23.95% |
| Average - 14.7 | 48.98 / 27.67% | 40.83 / 23.07% | 48.22 / 27.24% |
| Average - 20.9 | 63.32 / 35.77% | 43.26 / 24.44% | 51.46 / 29.07% |

For the XGBoost and 2D CNN, it is possible to observe a proportional relation between $T_{contact}$ and the increase in the error. While for the NN model, the relation is not proportionally consistent. Nevertheless, it is possible to conclude that the fact that there is a longer contact duration affects the precision of the impact position detection. This result was expected, and it could be related to the fact that the impactor is dampening the vibrations on the plate. Thus, affecting the magnitude of the natural frequencies that are excited.

To make the model more robust, $T_{contact}$ can be set as a second predicted value on the ML models. In this case, the system will not only predict the location of the impact but also the contact duration. The developed experimental setup can be used, in a straightforward manner, to obtain different datasets with diverse conditions (e.g. contact duration, force, among others). This approach can also be used for other predicted variables or input variables.

3.5 Conclusion

This chapter presents an iterative methodology, based on a set of building blocks, in this manner, it is possible to develop an impact position detection strategy using Machine learning. This study helps to understand the behavior and performance of machine learning for impact position detection in the particular domains that will be used for haptic feedback (i.e. thin rigid plates). Also, provide an alternative method for detecting the position of the finger in the tactile surface of a Digital Musical Instrument.

An automated impact acquisition system is proposed and two different versions of a Linear Impact Generator (LIG) are studied.

In addition, two representations for the time-domain impact signals were proposed based on domain knowledge, and the effect of both representations on the error of the impact source location task is studied. It was found that even though the FFT magnitude transformation is a simpler representation when compared to the spectrogram (i.e. it does not present the frequency changes in time), it allows the models to obtain better impact position detection results.

The proposed ML pipeline permits to extract the impact position from the stress signals that are recorded using a single piezoelectric transducer. The obtained precision is higher than the requirements for finger taps detection, and the error for impact location is lower than in previous studies in composite materials.

It was found that when an impact occurs, the duration of the contact with the surface has an important effect on the impact location precision. This parameter can be integrated into the inputs of the models to improve the robustness and generality of the detection task.

The proposed methodology and the experimental setup can be used to develop more complex ML systems.

In the future, it could be interesting to study why the error on the NN does not increase proportionally with the contact duration as in the other two models. Also, following the same methodology, one can build an ML system with multiple outputs. It is possible to predict the impact force and the number of contact points. If a limited number of contacts is defined, the model can be designed to extract multiple contact locations from the acquired signal, thus providing a multi-touch detection system with a single transducer.

Publications related to this chapter :

- C. H. Mejia, J. Jayet, P. Germano, A. Thabuis, and Y. Perriard, "Linear Impact Generator for Automated Dataset Acquisition of Elastic Waves in Haptic Surfaces," in *2019 22nd International Conference on Electrical Machines and Systems (ICEMS)*, pp. 1–5, Aug. 2019.
- C. H. Mejia, P. Germano, S. C. Echeverri, and Y. Perriard, "Artificial Neural Networks for Impact Position Detection in Haptic Surfaces," in *2019 IEEE International Ultrasonics Symposium (IUS)*, pp. 1874–1877, Oct. 2019.
- C. Hernandez-Mejia, J. Chavanne, P. Germano, and Y. Perriard, "Effect of the Impact Contact Duration on Machine Learning Models for Impact Position Detection," in *2020 23rd International Conference on Electrical Machines and Systems (ICEMS)*, pp. 2063–2068, Nov. 2020.

4 Reinforcement Learning for Localized Peak Generation

4.1 Introduction

The Time-Reversal Method (TRM) dictates that localized peaks can be obtained using time-reversed signals to drive one or multiple actuators on a surface. These peaks are repeated to obtain a localized vibration and provide haptic feedback. This chapter studies an alternative method to improve the quality of the peaks. This optimization is carried out in an experimental setup with a single piezoelectric transducer.

The research questions for this chapter are *"How can Machine Learning (ML) and Deep Learning (DL) be used to represent the time-reversed signals used to create a localized peak?"* and *"How can DL be used to optimize the contrast ratio and ensure the desired location of the peak?"*. The associated hypothesis is that *"DL can capture the one-to-one mapping between the time-reversed signals and the peak location, and this representation can be improved by imposing some metrics on the generated data points"*.

Deep learning can be used to model and optimize complex systems. Moreover, it has proven to overpass the accuracy and generalization of some mathematical modeling strategies when sufficient data are available [69, 113]. Seeing that, it is interesting to study the ability of Deep Neural Networks to generate time-reversed signals capable of creating localized peaks.

Reinforcement Learning (RL) [114] is an approach for training deep-learning models to map situations of an environment with the actions of an actor aiming to maximize a numerical reward. In this framework, the learner is not told which actions to take but must discover which actions yield the highest reward by examining them (i.e., a trial-error approach).

In this chapter, we explore the potential of RL to optimize the time-reversed signals used in an experimental setup to obtain localized peaks. The aim is to increase the contrast ratio (ratio between the peak amplitude and the displacement elsewhere at the instant of focalization, Section 2.7.2) and ensure that the localized peak occurs at the desired spot.

The first step is to select the time-reversed signal acquisition strategy and explain how these signals are used to generate a localized peak. The second step is to introduce the reinforcement learning framework and select the most suited algorithm that will be used for the optimization experiments. The last step is to run the RL optimization and evaluate the generated peaks using an experimental setup and a set of metrics.

4.2 Domain Knowledge on Peak Generation with TRM

Before training and optimizing a Neural Network (NN) to generate localized peaks, let us start by understanding how to acquire a time-reversed signal and create a localized peak. Furthermore, two methods to acquire the Tr-signal are compared.

As mentioned in Chapter 2, there are two experimental methods to obtain such a signal. One is to create mechanical impacts and record the flexural waves using piezoelectric transducers, which is described in Chapter 3. The other is to use the Impulse Response (IR) between the actuator and a given location on the surface of the plate.

4.2.1 Impulse Response (IR) Acquisition

An alternative to the mechanical impacts comes from the system control theory. Different input stimuli are used in system identification to understand the system's response. For example, the Step Response can provide information on the system's dynamics, and the Impulse Response can give information on the stability and frequency response.

Instead of using the piezoelectric transducer as a sensor, one can feed a voltage signal (V_{piezo}) into the piezo (at location x_b) and measure the displacement $u_a(t)$ at a given location (x_a) on the surface of the plate. Hence, the Transfer Function can be obtained by linking the input and the output of the system. This transfer function relates the voltage input on the actuator to the displacement at a particular location. Then, applying the inverse Fourier transform to the transfer function, one can obtain the IR of the system. The IR represents the signal on the piezoelectric transducer when an impulsive stimulus is given at the focusing location (x_a).

This approach not only simplifies the acquisition, but also helps to reduce the error coming from mechanical delays (e.g., the friction of the mechanical system and delays in pneumatic actuation). Moreover, creating a repeatable electrical signal is much simpler than a repeatable mechanical stimulus.

The common approach to acquiring the IR is to feed a broadband signal, like white noise, to the piezo actuator. While the signal is emitted, a Compact Laser Vibrometer (CLV) is used to measure the displacement (CLV_{out}) at the focusing location (x_a). Both signals are used to calculate the transfer function. Finally, the IR is obtained using the inverse Fast Fourier Transform (inverse FFT). This approach has been used by several authors [57, 58, 62].

Signal processing strategies can inspire an important simplification to this procedure. Instead of using white noise, one can use a swept-sine wave. Since the linear sine sweep is a white excitation signal, the impulse response can be directly calculated by performing the convolution between the measured displacement $CLV_{out}(t)$ and the time-reversed excitation signal $V_{piezo}(-t)$ [115]:

$$IR(t) = CLV_{out}(t) * V_{piezo}(-t) . \quad (4.1)$$

Experimental Setup for IR Data Acquisition

To create an IR dataset, a considerable amount of linear sine sweeps must be acquired. Thus, it is necessary to automate the acquisition of the swept-sine response. For this purpose, the automated experimental setup used in Chapter 3 is transformed into an automated surface vibration scanning system.

To do this, the LIG is replaced with a Compact Laser Vibrometer (Polytec CLV 1000), which allows measuring the speed and displacement of a given spot on a vibrating surface. The CLV is mounted on the CNC table, so it is possible to focus the laser spot at different locations over the surface of the aluminum beam. Then, an arbitrary signal generator (TG5012A) is coupled with a voltage broadband amplifier (TOE7607) to reproduce the desired signal. This voltage signal is fed to the piezo on the aluminum beam experimental setup (described in Section 3.2.2). Lastly, an oscilloscope (Lecroy Wavesurfer 3074) is used to acquire a) the surface-displacement signal (i.e., the CLV output), b) the piezo input voltage (e.g., the amplified swept-sine signal), and c) a trigger signal that synchronizes all the devices. The entire system is controlled from a remote computer that runs the automation script in Python. The automated vibration scanning system is depicted in Fig. 4.1.

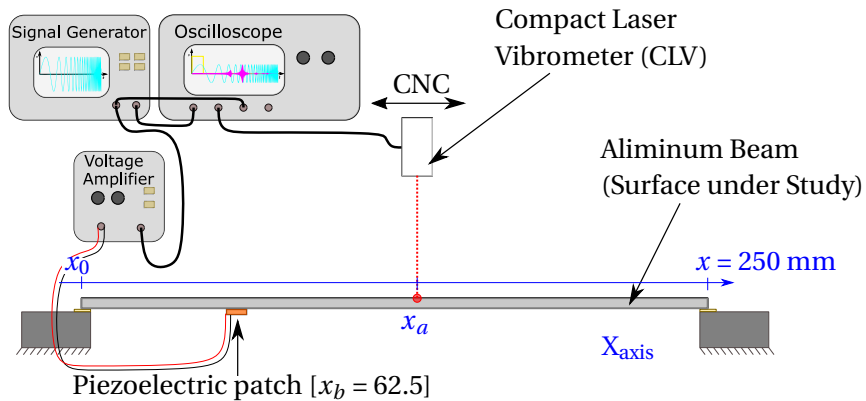


Figure 4.1: Experimental setup for automated swept-sine acquisition.

Since the CLV is very sensitive, any external vibrations (e.g., noise from the building) would be captured on the response of the system (CLV_{out}). To ensure a clean displacement acquisition, the CNC system is mounted on top of a vibration-dampening table (TMC's CleanBench Table).

IR Data Acquisition

The acquisition process is conducted independently for every position. First, the CNC table moves the CLV laser spot to the starting point $x_{start} = 63$ mm. Later, the signal generator is configured to generate a linear sine sweep with a starting frequency, $f_{start} = 1$ kHz and an ending frequency, $f_{stop} = 70$ kHz. The duration of the sine sweep is set to $T_{sweep} = 500$ ms.

Once the signal is sent to the piezo, the oscilloscope acquires both the piezo input signal $V_{piezo}(t)$ and the response of the system $V_{CLV}(t)$ (i.e., the displacement of the aluminum surface at location x_a). The acquisition sampling rate is $f_s = 200$ kHz. The acquisition is repeated 30 times at every location. Finally, the laser spot is moved to the next position, and the process is repeated.

Following the approach from Chapter 3, the sine-sweep acquisition is divided into five groups. This approach helps to consider the effect of temperature variations in the dataset. Each group is acquired at different times of the day. The conditions for the acquired datasets are:

- Dataset # 1: Starting at $x_{start} = 63$ mm with steps of 5 mm until $x_{end} = 238$ mm.
- Dataset # 2: Starting at $x_{start} = 64$ mm with steps of 5 mm until $x_{end} = 239$ mm.
- Dataset # 3: Starting at $x_{start} = 65$ mm with steps of 5 mm until $x_{end} = 240$ mm.
- Dataset # 4: Starting at $x_{start} = 66$ mm with steps of 5 mm until $x_{end} = 236$ mm.
- Dataset # 5: Starting at $x_{start} = 67$ mm with steps of 5 mm until $x_{end} = 237$ mm.

An example of a swept-sine response acquisition at location $x_a = 93$ mm is displayed in Fig. 4.2. One can observe the linear sine sweep signal (V_{piezo}) in blue and the response of the system (CLV_{out}) in orange. The spikes on CLV_{out} appear at the system's resonant frequencies for that particular location. To help visualize the linear sine sweep, a sub-figure displays V_{piezo} with the time axis in the logarithmic scale.

Finally, the IR is calculated for every location using the approach presented in Eq. 4.1. An example of the calculated IR for location $x_a = 93$ mm is presented in Fig. 4.3. Because of the convolution operation, the length of the signal is twice the duration of the swept-sine. The first half of the signal can be discarded since it does not contribute to the localized peak.

The last steps to obtain the Time-reversed Impulse Response Dataset (TrIRDs) are: to flip the signal in time (time-reverse), to normalize the signal (between -1 and 1), and to crop the region of interest. The signal is cropped because only a portion of the IR is required to recreate the peak. This portion is defined by the time-reversal window (the duration of the driving signal T_s). Then, for the experiments in this chapter, the length of the signal is set to $T_s = 20$ ms. The TrIRDs has a total of 5310 TrIR signals. The TrIR for location $x_a = 93$ mm is presented in Fig. 4.4. It is possible to see the amplitude increasing until the instant $t = 20$ ms when the converging wavefront is reconstructed, and the localized peak is created.

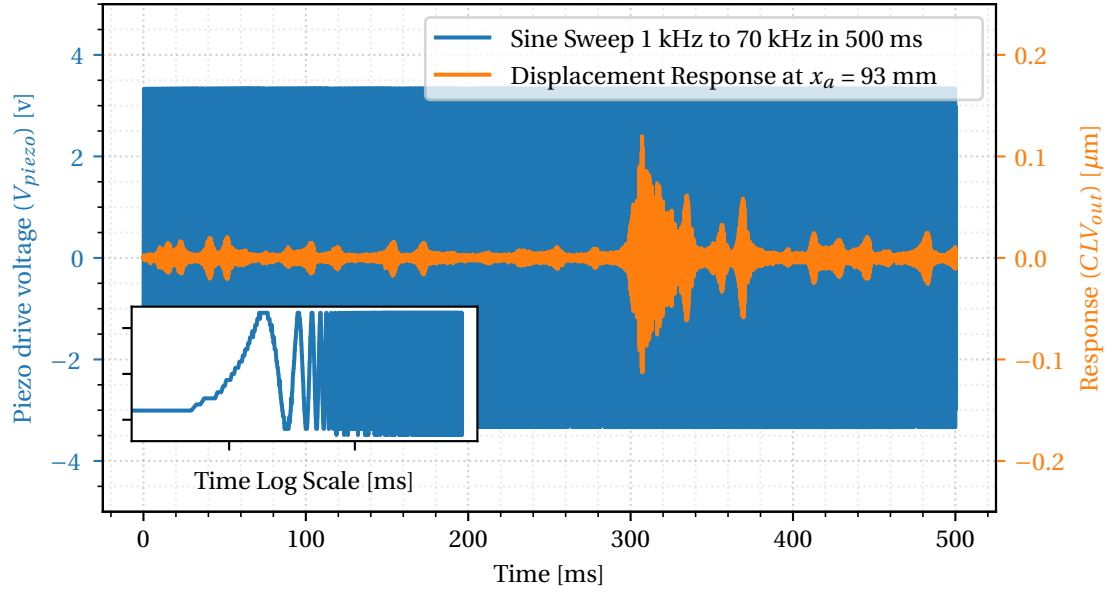


Figure 4.2: Swept-sine acquisition for location $x_a = 93$ mm. A subplot displays the linear sine-sweep (V_{piezo}) in the logarithmic scale.

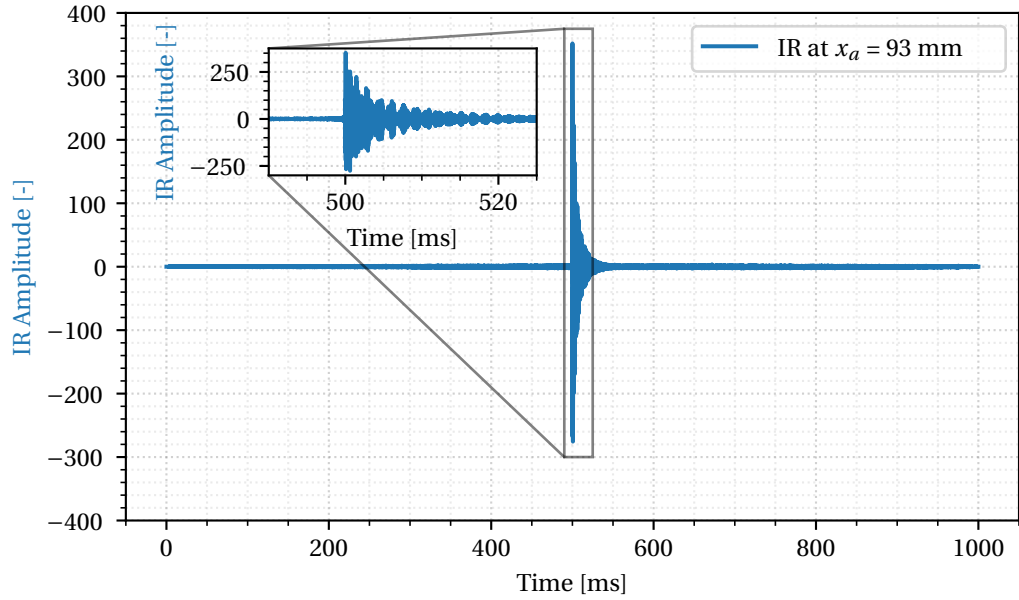


Figure 4.3: IR for location $x_a = 93$ mm. The IR is obtained by applying Eq. 4.1 to the swept-sine acquisition (presented in Fig. 4.2).

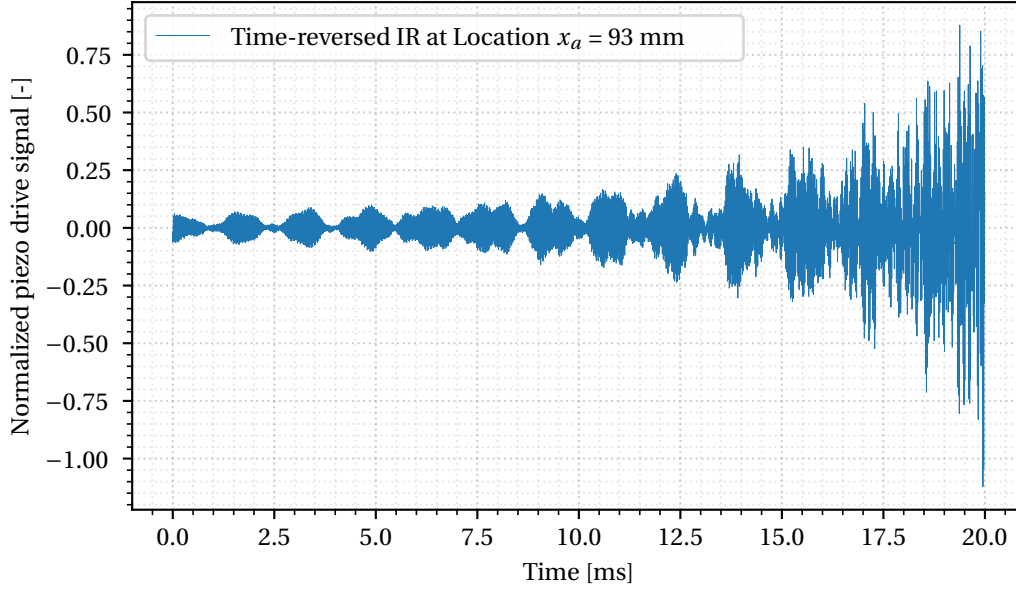


Figure 4.4: Time-reversed Impulse Response for location $x_a = 93$ mm. The signal is cropped to $T_s = 20$ ms.

4.2.2 Peak Generation and Comparison

As described in Chapter 2, to create a localized peak at location x_a it suffices to feed the time-reversed impact or IR signal into the actuator in location x_b .

To evaluate the properties of a localized peak, it is critical to measure the displacement of the surface over time. This evaluation allows us to compare a peak created by a time-reversed impact with a peak created by a time-reversed IR.

Experimental Setup for Surface Vibration Acquisition and Localized-peak Evaluation

The previous experimental setup was initially conceived to acquire the swept-sine response. However, it can also be used to study the wave propagation by scanning the entire surface of the aluminum beam. After this scan, one can evaluate the quality of a localized peak. The only change is that the swept-sine signal is replaced with the time-reversed impact or IR. The experimental setup for the surface vibration acquisition is presented in Fig. 4.5.

The surface vibration scan goes as follows: first, the time-reversed signal is transferred to the signal generator. This signal is amplified and fed to the piezoelectric actuator. Then, the CLV and CNC acquire the vibrations at several locations. Since the CLV can only record the displacement in one position at a time, the acquisition is repeated at several points over the surface. An $N \times M$ matrix of points defines the area of interest. N is the number of points in

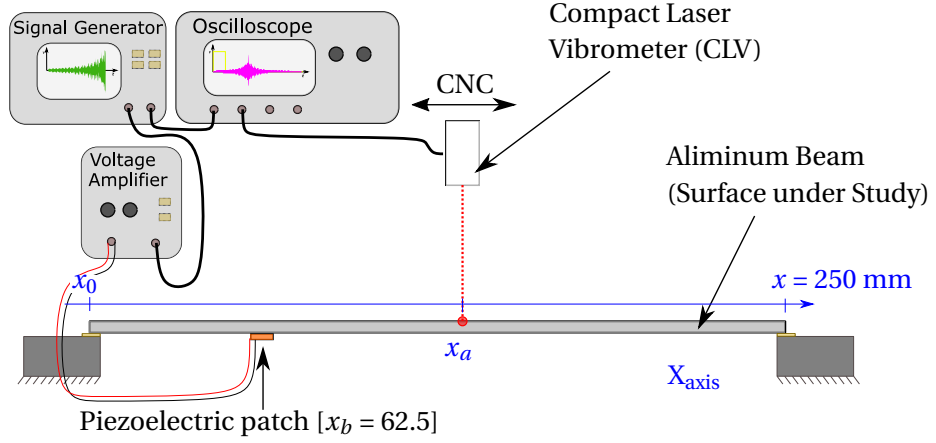


Figure 4.5: Experimental setup for automated surface vibration acquisition (surface vibration scan).

the X direction, and M is the number of points in the Y direction. The CNC moves the laser spot to the first point where the acquisition starts. At each location, the laser remains static while the same signal is fed to the piezo actuator. This signal also triggers the acquisition of the oscilloscope. After each acquisition, the computer stores the displacement signal. Then, the CNC moves to the next location, where the acquisition is repeated. Fig. 4.6 presents the displacement of the surface for a single location, $x_a = 93$ mm, when the time-reversed IR for location $x_a = 93$ mm is supplied to the piezoelectric transducer. The signal generator is set to $2.5 V_{pp}$, and the voltage amplifier multiplies the amplitude by 20 (i.e., the drive signal has an amplitude of $50 V_{pp}$).

When the whole surface has been scanned, the displacement signals can be superposed in space and explored in the time dimension to study wave propagation. Fig. 4.7 presents an example of a 3D plot with the displacement of the surface at the instant of the focalization ($t_{peak} = 20$ ms). The peak is created when the time-reversed IR for location $x_a = 93$ mm (presented in Fig. 4.4) is used to drive the piezo actuator. For this displacement acquisition, the surface was scanned from $x_{start} = 0$ mm to $x_{end} = 248$ mm. The scan spatial resolution is set to $R_{scan} = 2$ mm, which leads to an $N = 125$. In the Y direction, eight rows were scanned, thus $M = 8$. A cutting plane is displayed at the center line of the 3D plot, $y = 8$ mm, and the displacement profile is the plot in black.

Because of the geometrical properties of the beam ($l \gg w$), this study can be considered a one-dimensional wave propagation problem. Thus, it is only necessary to study the center line or middle plane of the aluminum beam (plane located at $y = 8$ mm). The displacement profile for the center line at the instant of the focalization is presented in Fig. 4.8. The 2D scan of the surface takes 55 minutes, while the 1D acquisition (scan of the center line) takes only 8 minutes.

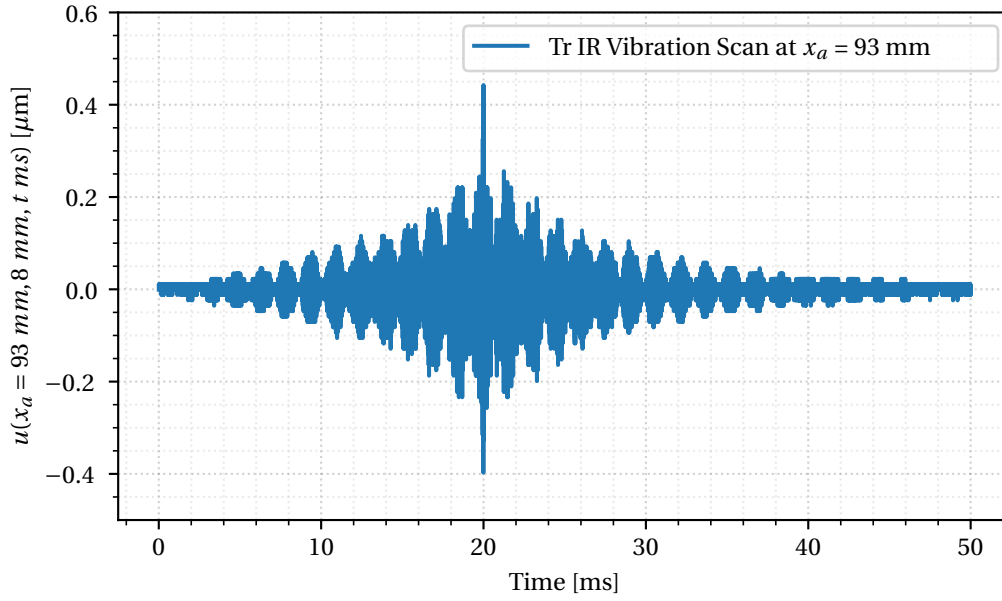


Figure 4.6: Displacement of the surface over time ($CLV_{out}(t)$) for the focusing location, $x_a = 93 \text{ mm}$.

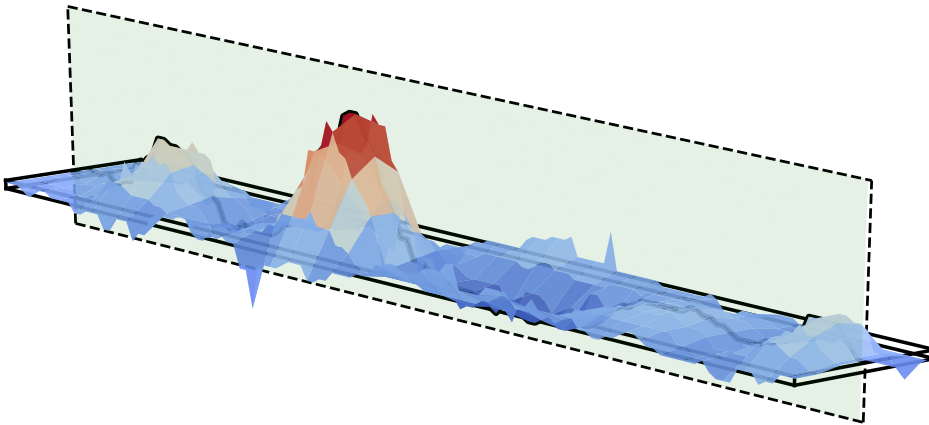


Figure 4.7: Three-dimensional plot (3D) for a surface vibration acquisition at the instant when the peak occurs, $t_{peak} = 20 \text{ ms}$. The thickness of the aluminum beam is not at scale.

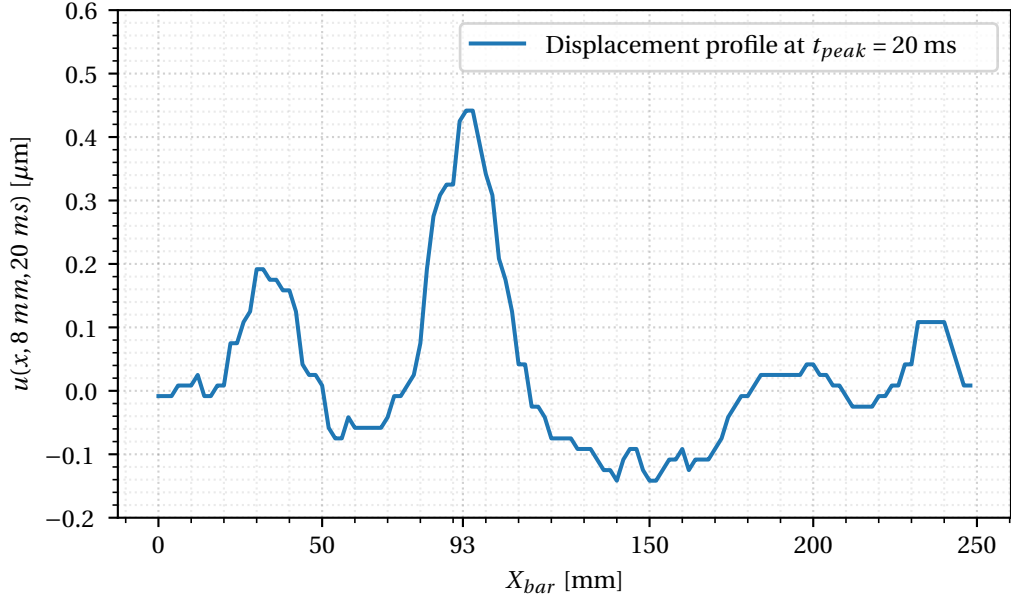


Figure 4.8: Displacement profile of the beam center line for the Time-Reversed IR at $x_a = 93$.

After the vibration scan, the quality of the localized peak can be evaluated. For this purpose, four metrics are defined. x_{peak} , t_{peak} , u_{peak} , and C'_r respectively, the location of the maximal displacement peak, the instant when the localized peak appears, the amplitude of the localized peak, and the modified contrast ratio.

This chapter proposes a modified version of the contrast ratio, C'_r . In the proposed metric, the peak amplitude (u_{peak}) is compared to the average maximal amplitude at all other locations and during the whole acquisition (i.e., the maximum displacement at each location x_a during the acquisition period, $T_{acquisition}$). This metric differs from the traditional contrast ratio C_r , where only the instant of the maximum peak is evaluated (Section 2.7.2). This modification ensures that the localized peak is bigger than any other peaks during the reconstruction period (i.e., the signal duration T_s). The modified contrast ratio, C'_r , is presented in Eq. 4.2:

$$C'_r = \frac{u_{peak}}{\frac{1}{n_{scan}} \sum_{x_a=x_{start}}^{x_{end}} \max(u(x_a))}, \quad (4.2)$$

where n_{scan} is the number of locations x_a for the vibration scan, $x_{start} = 0$ mm, and $x_{end} = 248$ mm. For this study, $n_{scan} = 125$ (i.e., length of the bar $l = 250$ mm divided by $R_{scan} = 2$ mm).

Localized Peak and Signal Comparison

As mentioned before, both the Time-reversed Impact (Tr Impact) and the Time-reversed IR (Tr IR) signals can be used to obtain a localized peak. Nevertheless, it is essential to understand the quality of the localized peak obtained for both signals and choose the most suitable signal.

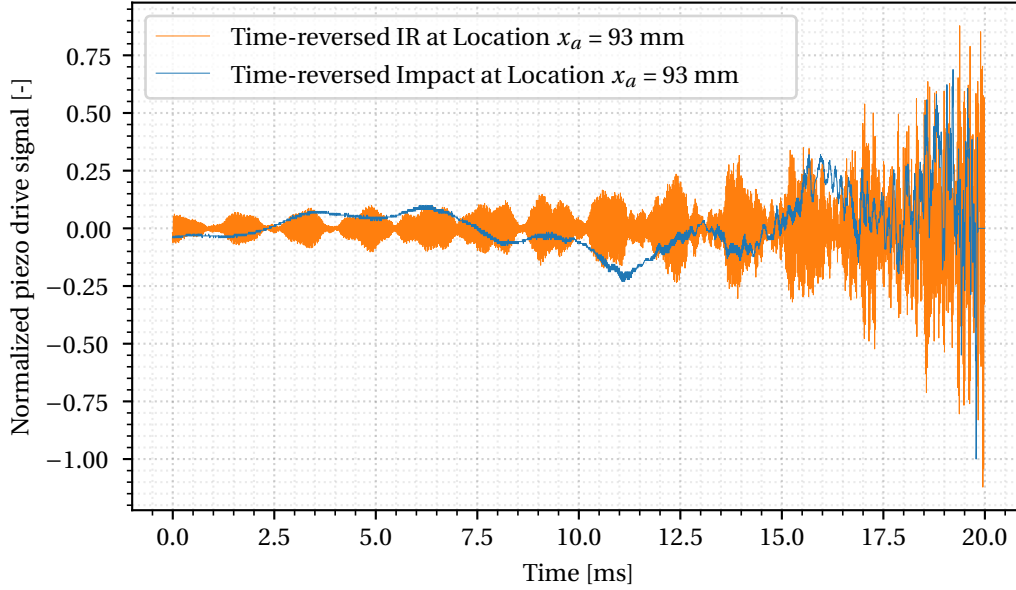


Figure 4.9: Time-reversed Impact and time-reversed IR in the time domain.

Fig. 4.9 presents the Tr Impact and the Tr IR signals. It is not very easy to compare them in the time domain, but it is already possible to identify that the IR contains much higher frequencies than the impact signal. Moreover, the IR provides more energy to the system from the beginning of the driving period. Lastly, for the Tr Impact signal, it is possible to see that the signal is not ending at exactly 20 ms but a couple of milliseconds before. The mechanical system delays explain this difference (e.g., the delay between the actuation and displacement of the LIG).

Comparing both signals in the frequency domain is easier. Fig. 4.10 presents the magnitude of the FFT for both normalized signals. Note that the magnitude of the FFT is nondimensional due to the signal normalization. As predicted, the Tr IR contains much more energy on the higher frequencies. Most of its energy concentrates around the 42 kHz range, which was identified as the resonant frequency of the electromechanical system in Chapter 2.

For the Tr Impact signal, the energy is concentrated in the lower frequencies, which can lead to lower amplitudes of displacement on the localized peak. The limitations of the LIG can explain the frequency content of the Tr Impact. In the LIG, the maximal speed and contact duration are limited. As a consequence, the LIG is not able to stimulate the higher resonant

modes of the beam. In contrast, the electrical signal has virtually no limitations, so it can recreate an impulsive stimulus closer to a perfect impulse or at least stimulate the frequencies within the limitations of the swept-sine input, V_{piezo} . i.e., the mechanical impact is not as perfect as the electrically induced impulse.

The frequency spectrum of both signals has a similar shape between 2 kHz and 40 kHz. In this range, the magnitude peaks appear at the same frequencies. These similarities are expected because the IR and the mechanical impact are provided at the exact location. Thus, they stimulate similar natural modes on the beam.

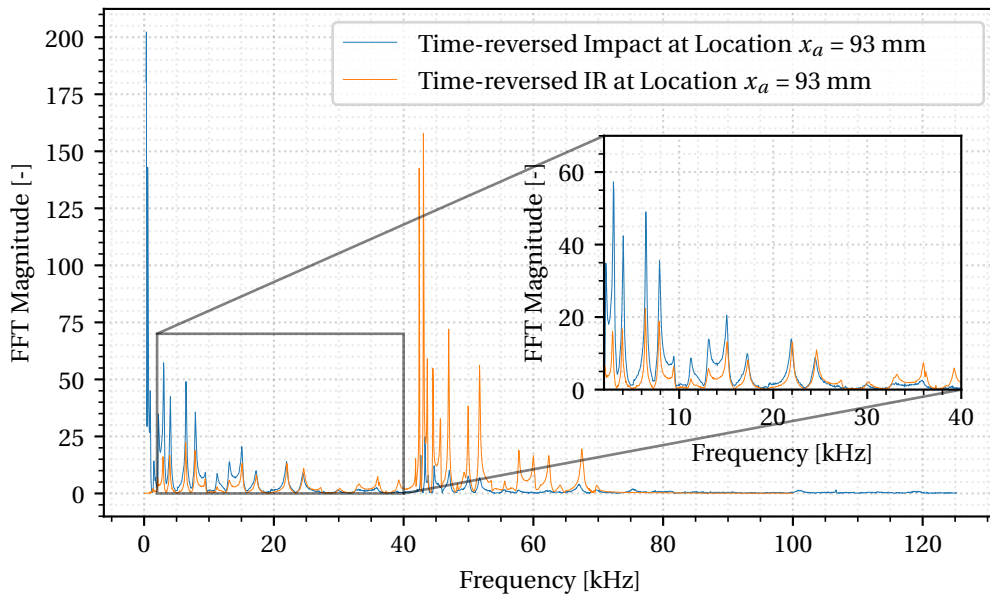


Figure 4.10: Magnitude of the FFT for the time-reversed Impact and time-reversed IR.

The next step is to compare the properties of the localized peak for both signals. To do this, the center line of the aluminum beam is scanned. Fig. 4.11 presents two localized peaks. Both peaks are obtained when a time-reversed impact and an IR are used to drive the piezo actuator. Both acquisitions are made independently.

The peak amplitude appears at the desired location for both signals, $x_{peak} = 93$ mm. As expected, the Tr IR achieves a higher amplitude, approximately six times bigger than the Tr Impact signal. Moreover, the peak for the TrIR appears at $t_{peak} = 20$ ms, precisely after the whole signal has been reproduced. In contrast, the peak for the Tr Impact appears at $t_{peak} = 19.81$ ms. This is expected since the maximum peak of the voltage signals for the impact has a delay. This delay can affect the control over the appearance of the localized peak, especially when simultaneous peaks are desired at different locations over the surface.

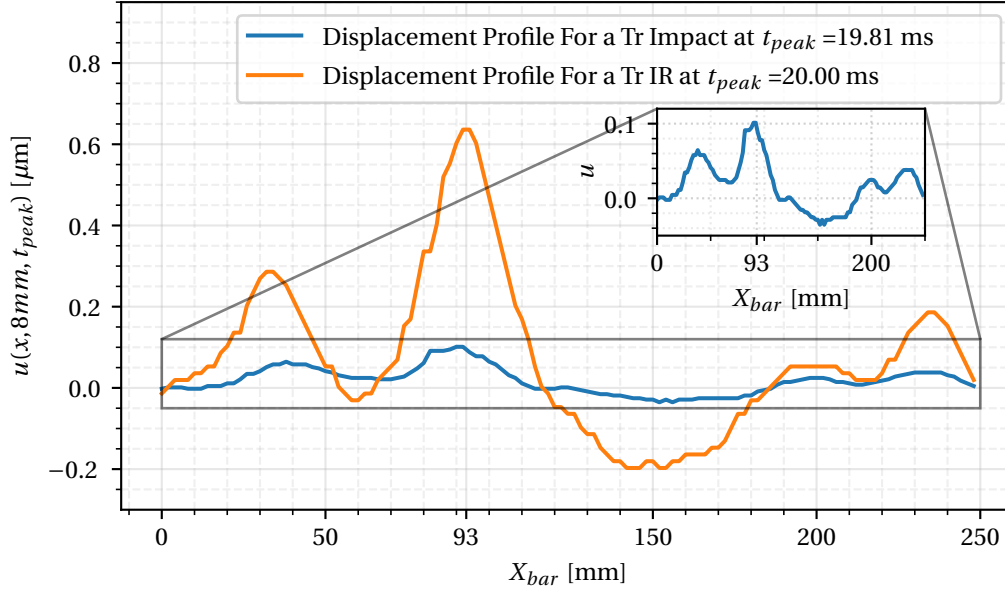


Figure 4.11: Displacement profile for two localized peaks. The peaks are created with a Tr Impact and a Tr IR at location $x_a = 93$ mm.

Simultaneous Peak Creation at Different Locations

To provide multi-touch feedback, one can create localized peaks in different locations by combining the time-reversed signals for both locations. After adding both signals, the resulting signal can be fed to the actuator. As a demonstration, the time-reversed signals for an impact and an IR at location $x_a = 93$ mm and $x_a = 208$ mm are combined (respectively). Fig. 4.12 displays the displacement profile for the localized peaks that appear after a Tr IR is fed to the piezo. It is observed that both peaks appear at the same instant of time, $t_{peak} = 20.00$ ms.

Fig. 4.13 displays the displacement profile after a vibration scan using the combined Tr Impact signals. In this case, the peak at location $x_{peak} = 208$ mm appears $90 \mu s$ after the peak at location $x_{peak} = 93$ mm is recreated. This time difference occurs because the Tr Impact signals are not perfectly synchronized (i.e., with the Tr Impact, it is hard to control the instant when the peaks appear because of mechanical delays on the LIG).

Tr-Signal Selection

In conclusion, acquiring a Tr IR signal is much simpler than acquiring a mechanical impact. Electrical signals are easier to control and more repeatable than mechanical impacts. Besides, when using a mechanical impact for the TRM, the whole system's transfer function differs during the acquisition and focusing steps (i.e., delays associated with different instruments). In contrast, the piezo is used as an actuator for the Tr IR signal in both steps. This consistency

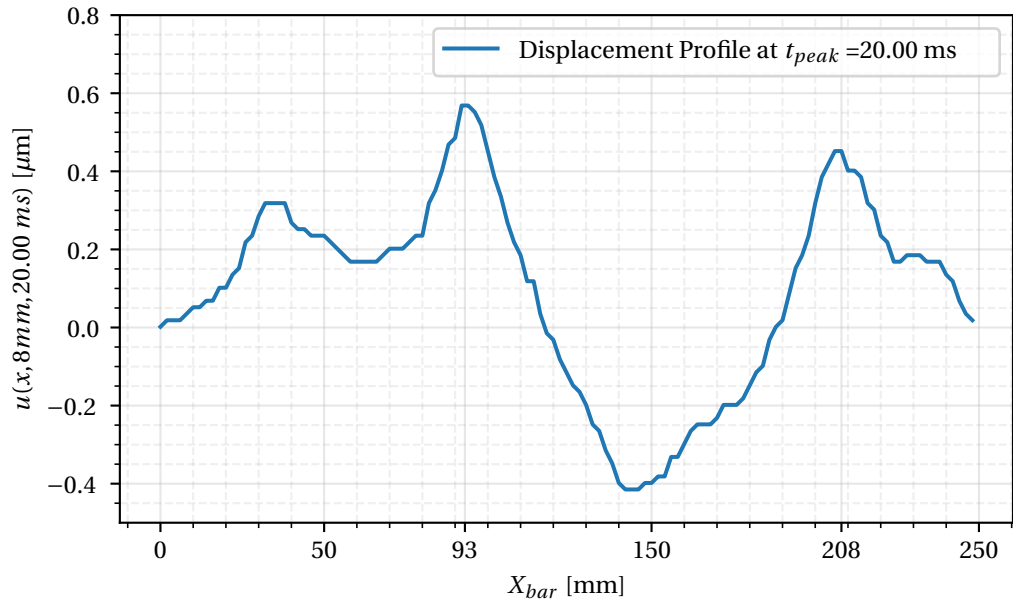


Figure 4.12: Displacement profile for the center line of the beam after a vibration scan for the combined Tr IR signal ($x_a = 93\ mm$ and $x_a = 208\ mm$).

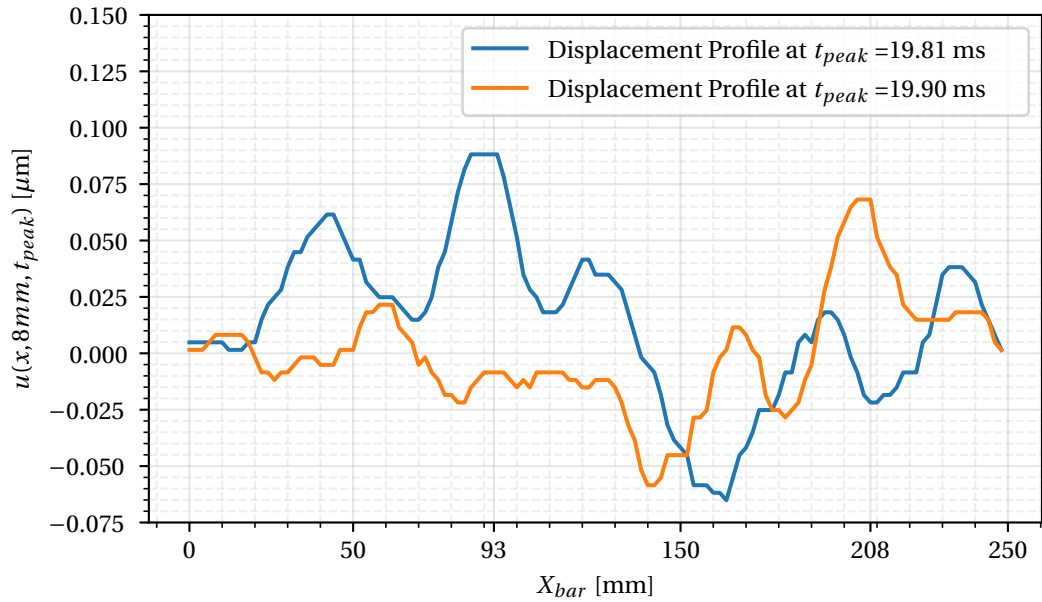


Figure 4.13: Displacement profile for the center line of the beam after a vibration scan for the combined Tr Impact signal ($x_a = 93\ mm$ and $x_a = 208\ mm$).

can also justify why the IR has a better reconstruction quality and higher amplitude on the localized peak. For these reasons, the Tr IR signal is preferred over the Tr Impact and will be used in the rest of the thesis for localized peak creation.

4.3 Methodology: Localized Peak Optimization

The alternative approach to optimizing the localized peak is to train a Deep Neural Network (DNN) to generate the time-reversed signal. The task of the DNN is to learn the one-to-one mapping between a single input, $x_{desired}$, and a higher dimensional output, V_{piezo} . Where, $x_{desired}$ is the desired peak location (e.g., $x_{desired} = 114$ mm) and V_{piezo} is the Time-reversed IR (TrIR) signal that can generate a localized peak at that location. Since the model converts a low-dimensional input into a higher-dimensional output, it can be called a decoder or upsampling Neural Network.

Once the DNN is trained for this task, the vibration scanning experimental setup is integrated into the training process to optimize the localized peak. The signal optimization is carried out using the Reinforcement Learning framework which can explore different modifications to the DNN to improve the properties of the peak. Since the experimental setup is part of the training process, it will be called the Hardware-in-the-Loop (HiL) environment.

4.3.1 Reinforcement Learning (RL)

Reinforcement Learning (RL) [114] is an approach to train deep-learning models to map situations of an environment to the actions of an agent aiming to maximize a numerical reward. In this framework, the learner is not told which actions to take but must discover which actions yield the highest reward by examining them (i.e., a trial-error approach).

The RL setup comprises two parts, an agent and an environment. The agent outputs an action that is fed to the environment. The environment then returns a reward that is used to train the agent. There are different types of agents, but generally, the agent is responsible for the learning process. Its objective is to find a balance between exploration and exploitation. The agent explores (discovers) new actions to learn new approaches and exploits (specializes or fine-tunes) the knowledge it has to ensure that it reaches the desired objective.

The RL framework is usually trained during several episodes. An episode starts with the reset function, followed by the step function that can be repeated once or several times until the environment reaches the final goal.

RL Model Selection

The selection of the RL agent model is a crucial step since several algorithms are available [116].

The first criterion to consider is the action space. In the peak generation problem, the action is a continuous signal composed of a series of continuous values. This action differs from a discrete action in a video game, such as up, down, left or right. Considering this, one can discard all the discrete models, such as Q-learning, Deep Q-learning, and SARSA, which are designed for a finite and small number of actions.

The second criterion is the possibility to control and fine-tune the exploration of the model. On the one hand, stochastic models tend to have a more aggressive exploration, which accelerates the learning phase but can lead to noisy outputs. On the other hand, deterministic models control the exploration/exploitation ratio, which allows for fine-tuning or optimizing the agent's actions. Deterministic models such as Deep Deterministic Policy Gradient (DDPG) or more complex versions are preferred. In contrast, stochastic models such as Proximal Policy Optimization and Policy gradient can be discarded.

The DDPG is the most straightforward deterministic agent with continuous action space, which makes it a reasonable choice. In DDPG RL, the agent is composed of two independent NN. The first one is the *actor*, the agent's core part. The actor takes the desired target and outputs the action used to actuate the environment. The second network is the *critic*, which takes both the action and the target and predicts the expected reward. This network is used to improve the actor's learning capacity. One can think of it as a normalization function. The critic outputs a mean reward or a baseline to reduce the variance of the reward during the learning process.

Since this model is deterministic, the actor's output is the actual signal, not the probabilities of an action. This characteristic, and the separation between the actor and critic, allows us to pre-train the actor using a supervised regression model (this step is explained later in this chapter). This strategy dramatically accelerates the learning process because the goal is not to learn the signals from scratch but to fine-tune the action. Seeing that, the possibility to freely choose and control the exploration becomes much more important than letting a stochastic model choose the randomness of its action. In conclusion, with this approach, exploitation is preferred over exploration, which allows for fine-tuning existing and efficient Time-reversed IR signals. Nonetheless, a controlled exploration is still required to let the model learn new signals.

RL Exploration strategy

In DDPG, the new behavior is explored by adding white noise to the action. This noise has zero mean and variance, σ^2 , which can be tuned.

A high σ^2 value leads to a more aggressive exploration but leads to noisier signals, so it is not possible to fine-tune an efficient signal. In contrast, the agent cannot discover new behaviors if the σ^2 value is too low.

RL for Localized Peak Optimization

In this thesis, the different RL elements are as follows. The actor is a NN that takes the desired location for the peak as an input and provides the driving signal as an output. The environment is the HiL experimental setup, composed of the aluminum beam and the automated scanning system. In our case, there is only one step per episode because each time the Tr-signal is sent into the HiL environment, it provides a reward. At each episode, the reset function defines the target location for the localized peak. Then the step function gives this input to the actor, which generates the driving signal. This signal is then sent into the aluminum bar experiment, which is scanned to study the generated peak. After the vibration scan, the step function calculates the reward function, which is fed back to the agent. During the different experiments, the agent is trained through several episodes (at least 300).

4.3.2 Actor: Signal Generation Strategies

To create a localized peak, the actor must create a voltage signal that can drive the piezoelectric actuator. As defined in Section 4.2.1, the TrIR signals from the dataset contain 4000 samples ($T_s = 20$ ms sampled at 200 kHz). Thus, it is vital to determine an efficient strategy to generate this signal since it has important repercussions on the number of actions and the complexity of the RL model.

Time-domain Signal Generation

The most obvious solution is to train the actor to generate the time series directly. One can implement a NN with as many output neurons as there are values in the desired signal. Then, this actor can generate the signal in one forward pass. One can use transposed convolutions layers to take advantage of the temporal continuity and relation with neighboring time samples. Nonetheless, this implies a Reinforcement Learning episode with a high dimensional action space (4000 actions). Then, although the reward immediately follows the choice of the action, it is hard for the agent to progress efficiently during learning. By definition, the RL algorithm is designed for long-term rewards and small action space, so this signal-generation strategy should not be considered.

Frequency-domain Signal Generation

Based on the knowledge acquired in Chapter 3, the FFT transformation is a powerful yet compact representation of the time-domain signal. One can consider working with the Fourier transform. As a result, the action would be in the frequency domain. Then, the action can be transformed back to the time domain before feeding it to the environment. The first approach is to output several (unknown number) tuples of three elements: 1) a frequency value, 2) the associated amplitude, and 3) the phase. There are several questions and considerations. How many of these tuples are needed to generate an efficient time-domain signal? Is it possible

to eliminate one of these three elements (frequency, amplitude, or phase)? In this sense, is the phase not critical, or only some limited frequency values are enough to reconstruct the time-domain signal?

To answer those questions, we propose to plot these quantities across different samples of the TrIRDs dataset. To facilitate the computation and visualization, only 50 samples are studied (approximately one sample for every 100 samples).

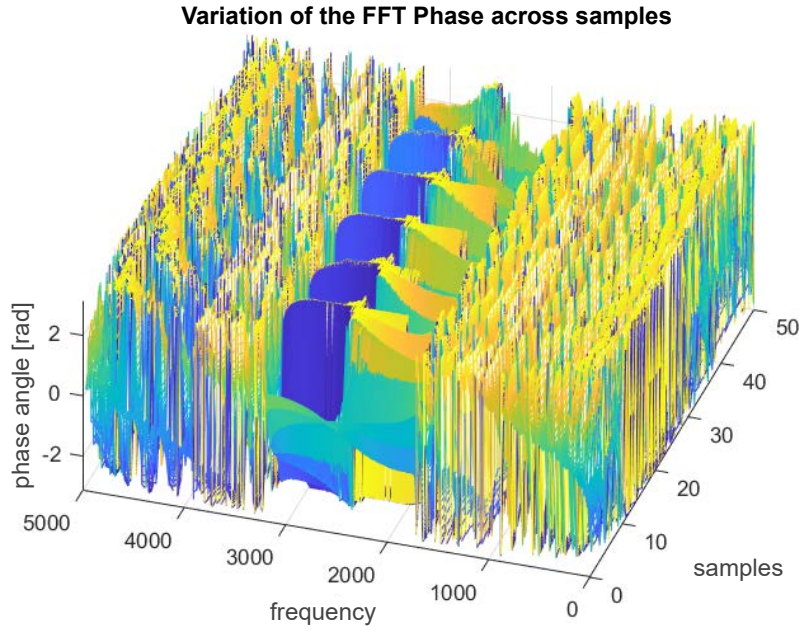


Figure 4.14: FFT Phase for 50 TrIR signals at different locations on the aluminum beam.

Fig. 4.14 displays the phase for the FFT of multiple TrIR signals at 50 different locations. It can be seen that the phase is very complex and hard to model. It behaves similarly to white noise for more than half of the signal. In general, for impulsive signals like an impact, an impulse response, or a drum sound recording, the phase is chaotic and tends to be random [83]. For this reason, when frequency transformations are carried out, the phase is generally omitted or, in some cases, replaced by a white-noise array [84]. Then, it seems reasonable to ignore this parameter, which helps to reduce the action space.

Fig. 4.15 shows the magnitude of the Fourier transform for the same 50 TrIR signals. It even seems smoother and easier to model than the time-domain signal, which is represented in Fig. 4.16. Furthermore, most frequencies have a null (or very low) magnitude, and the ones with the most significant magnitude are within the same frequency values. This behavior confirms the relationship between the beam's natural frequencies and the variation in the FFT magnitude for different locations. This trend is significant: if we consider only the amplitude of the FFT (instead of an unknown number of three-valued tuples), we can output a few amplitude values at predefined frequencies. These frequencies can be chosen based on the

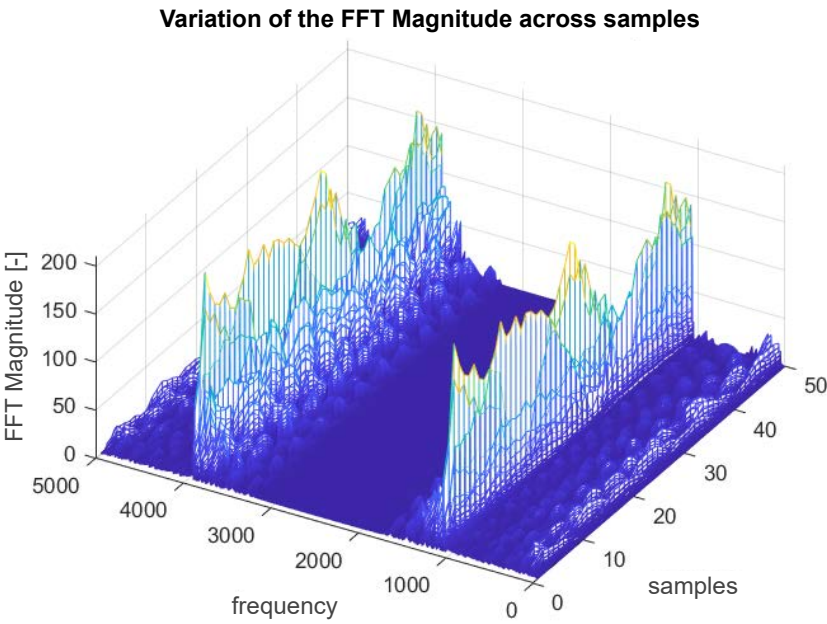


Figure 4.15: FFT Magnitude for 50 TrIR signals at different locations on the aluminum beam.

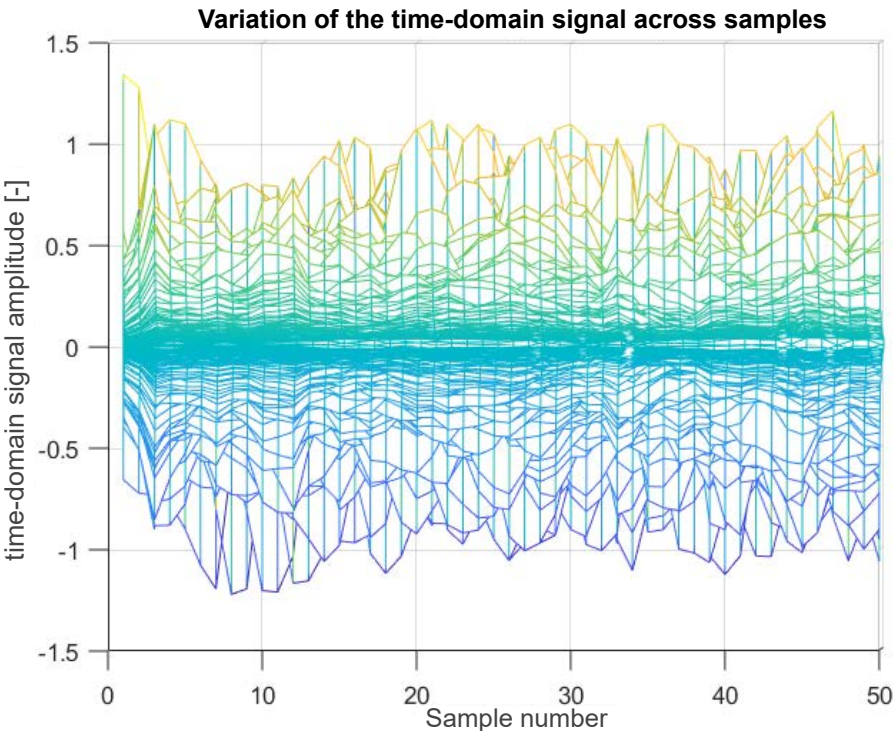


Figure 4.16: Time-domain TrIR signals at 50 different locations on the aluminum beam.

most relevant natural modes of the system.

This approach has some limitations. Because the phase information is omitted, there is no guarantee that the inverse FFT will adequately reconstruct the original time-domain signal. Nevertheless, there are some alternatives to deal with this problem. In [117], the authors demonstrate that by adding a delay and zero-padding at the end of the signal, it is possible to reconstruct any time-domain signal from the real part of the discrete Fourier transform.

Based on this solution, one can zero-pad the time-domain signal with one value on the left side (delay) and its original size on the right side. This trick allows for a good reconstruction. Several peak localization experiments were carried out. The reconstructed signal was observed to perform similarly to the original TrIR signal on the experimental setup (similar amplitude and modified contrast ratio).

In order to choose the most relevant frequencies for the entire TrIRDs, a magnitude threshold is defined. Any frequency component with a magnitude under that value is set to zero. The reconstruction experiments determined that at least the 20 most relevant frequency components are required to get similar performance on the peak generation. It was observed that the main frequency values for the different locations could be slightly different (frequency variance of 20 Hz), and some components do not appear in all signals. Lastly, only the frequencies that have a value higher than the threshold in at least one of the locations (x_a) are chosen. This selection strategy leads to 88 unique values. This signal generation method requires a reliable dataset and is appropriate only to improve existing signals.

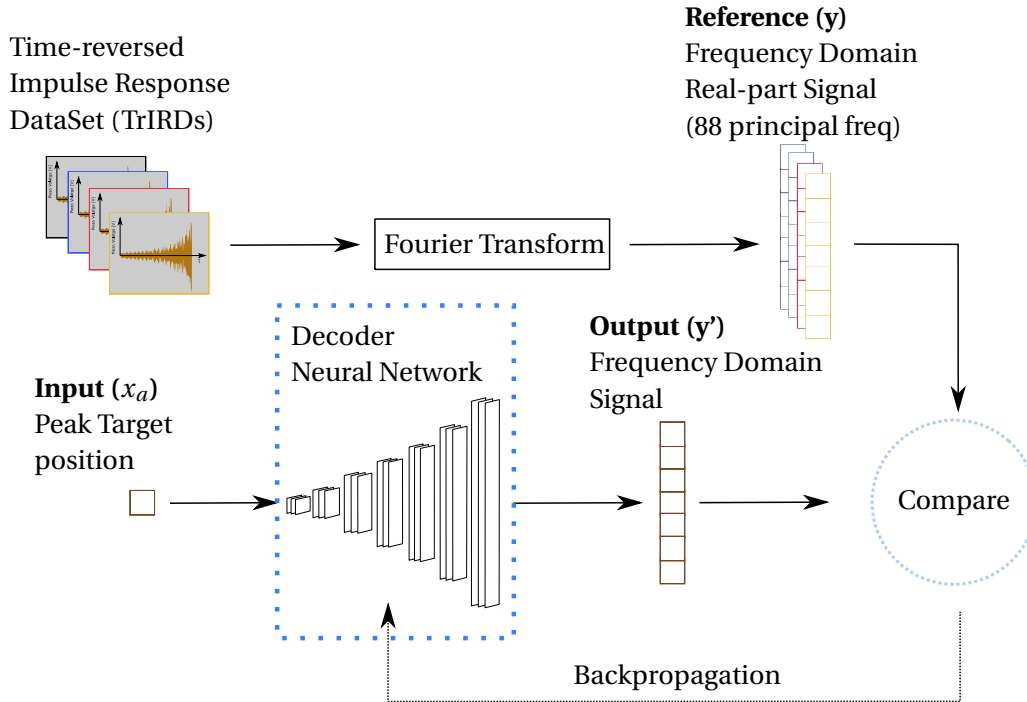
For this chapter, the frequency domain signal generation is preferred. This signal generation strategy reduces the number of outputs on the agent (number of actions), simplifying the RL process.

The complete RL methodology is summarized in Fig. 4.17. The proposed methodology has two main stages. In the first stage, the actor NN is pre-trained in a supervised manner using the Time-reversed Impulse Response Dataset (TrIRDs). Then the knowledge of this NN is transferred into the RL framework. This knowledge transfer is done by integrating the Pre-trained actor into the RL system. The second stage is the optimization of the actor using the RL framework. For this stage, the RL system is coupled with the Hardware-in-the-Loop experimental setup.

4.4 Pre-train the Decoder NN

An advantage of DDPG policy is its dual implementation using separate actor and critic neural networks. This characteristic allows the implementation of the actor in an independent Matlab script without taking care of the critic. Moreover, it allows training this actor in a classic supervised learning approach, which speeds up the RL training process by avoiding long exploration of the action space.

(a) Upsampling Deep Neural Network pre-Training



(b) Reinforcement Learning (RL) Optimization

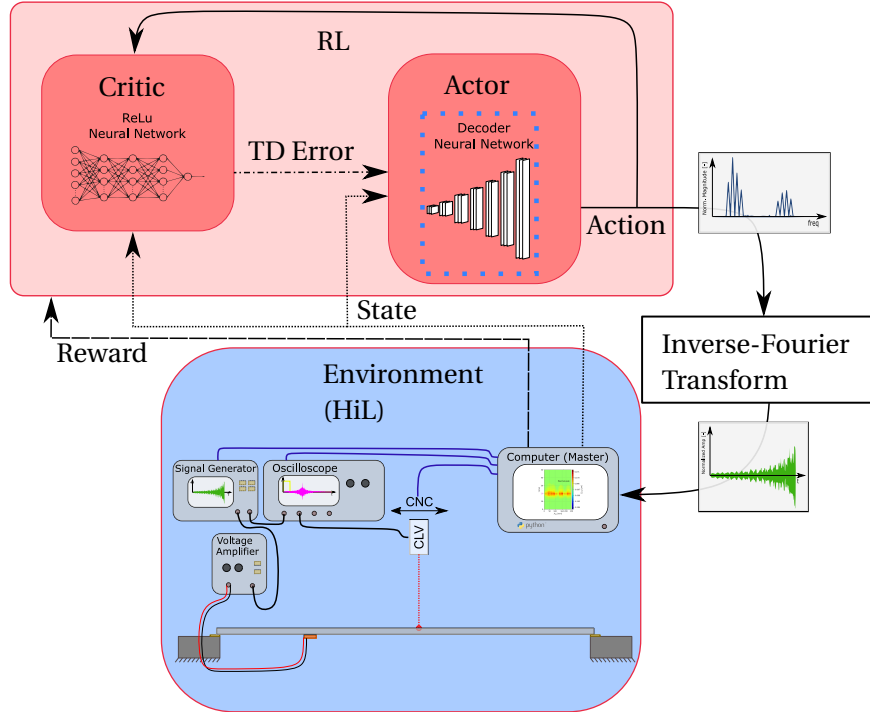


Figure 4.17: Optimization methodology and Deep learning pipeline. a) Supervised pre-training of the Decoder Neural Network and b) Reinforcement Learning (RL) optimization with Hardware-in-the-Loop (HiL).

4.4.1 Supervised Training of the Actor

The Matlab supervised-regression framework is used to train a DNN using the TrIRDs. The input of the Neural Network is the coordinate where the localized peak should appear, $x_{desired}$, and the output is a vector containing the 88 values of the magnitude of the FFT. It is a simple and fast process. The agent pre-train is treated as a regression problem. The DL pre-training pipeline is presented in Fig. 4.17 a.

There are several considerations for the training. First, the network size is arbitrarily limited to keep further optimization on the RL framework more efficient. Second, the weights of the NN are initialized with Gaussian noise. This trick helps reduce the learning time and avoid divergence during the NN training. The Gaussian noise has a standard deviation equal to $\frac{2}{\sqrt{IN_{dim}+OUT_{dim}}}$, where IN_{dim} is the dimension of the input layer and OUT_{dim} is the dimension of the output layer. Third, the input is normalized. As the desired coordinate, $x_{desired}$ can go from 63 mm to 240 mm, the mean value is equal to 155.5 mm, and the standard deviation is 130.8 mm. This normalization does not have an important effect on the supervised training step, but it radically helps the learning in the RL setup.

There is one additional advantage of pre-training the agent NN. Since the supervised training process takes less than 1 hour, it is possible to choose the best architecture of the NN by experimentation. The quick learning process makes the performance comparison between several models much faster than in classical reinforcement learning. For these reasons, the architecture of the NN is chosen during this stage.

4.4.2 NN Architecture Selection

There are many options when deciding on the architecture, but the focus will be on two general NN structures. The first agent is a fully-connected NN (commonly called Vanilla NN). This NN is a simple model which can simplify the training process. The second agent is based on a one-dimensional Convolutional Neural Network (1D CNN). This model has more parameters but can exploit the existing relation between adjacent frequencies.

The architecture of both models is described in Section A.2 in Appendix A. The two agents use the Rectified Linear Unit (ReLU) as the non-linear activation function. The final regression layer is added to both models since the Matlab Deep-Reinforcement-Learning framework requires it.

Both models were trained using the magnitude of the FFT for the TrIRDs (Simplified representation discussed in Section 4.3.2). The training dataset (TrIRDS) covers the entire work area, i.e., all the locations from $x_a = 63$ mm to $x_a = 240$ mm with a one-millimeter step. The selected loss function is the Root-Mean-Square Error (RMSE). Fig. 4.18 presents the RMSE along the training iterations.

Actor 2, based on a 1D CNN, provides the lower RMSE. For this actor, the RMSE can go below

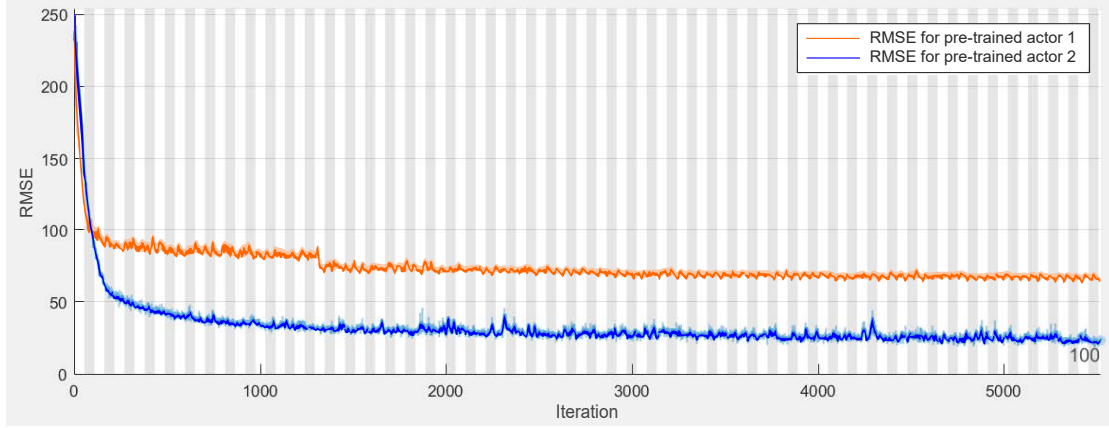


Figure 4.18: Plot of the RMSE during the supervised pre-training of both actor NNs.

50 and continues with a descending trend. While actor 1, based on the Vanilla NN, stagnates at an RMSE of 60. Following the same approach, different dimensions were evaluated for the layers of both architectures. Increasing the network size does not increase the performance of the generated signal. Nonetheless, a more extensive network is more complex and slower to train. Therefore, the Actor 2 (1D CNN) architecture is used for the following experiments.

Fig. 4.19 presents a TrIR signal generated with the pre-trained actor 2 and the equivalent signal from the TrIRDs. In the Fourier domain, the output of the supervised trained actor is very close to the original signal from the database. In the time domain, the signals are different. The big difference in the time domain comes from the fact that the original signal (Reference sample from the TrIRDs) contains all the frequency components. In contrast, the reconstructed signal only contains frequencies above the defined FFT Magnitude threshold.

At this stage, the resulting network brings no improvement compared to the peak generated with a signal from the dataset. In fact, there are minor losses in the quality of the localized peak. The performance of the generated signals (C'_r) is smaller than that of the Tr IR signals, and $x_{desired}$ slightly differs from x_{peak} . These differences will be corrected by the RL optimization in the next stage. The comparison of all results is presented in Section 4.6.

Once the actor NN is trained, the knowledge can be transferred into the reinforcement learning setup. i.e., the pre-trained NN becomes the actor in the RL framework, and it is the starting point to improve the signal.

4.5 RL Experiments

The Matlab RL framework is used to optimize the localized peaks that can be generated with a signal from the agent. In this stage, the pre-trained network is integrated into the RL loop. In addition, the HiL experimental setup is integrated into the RL system to evaluate the quality of the localized peak at each episode.

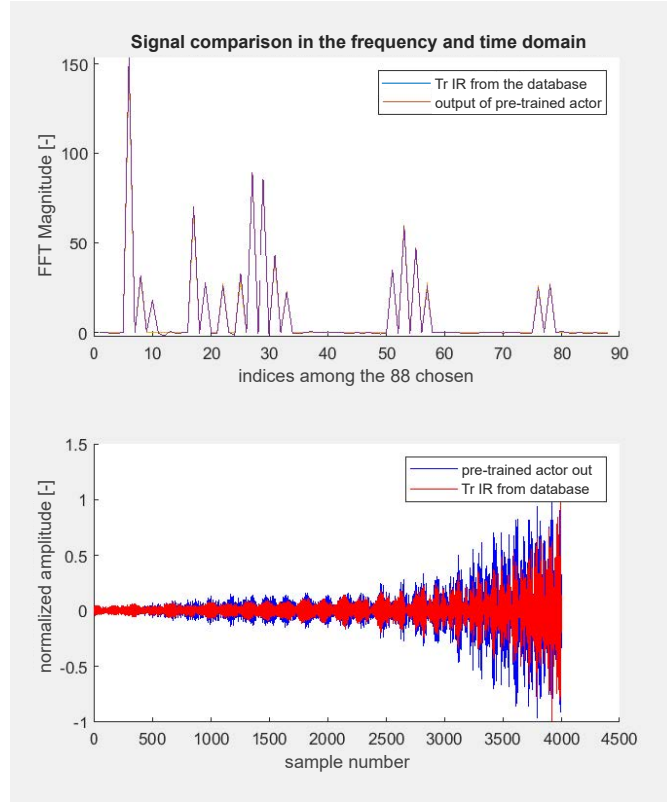


Figure 4.19: IR from the dataset compared with the generated signal for the pre-trained actor 2.

The optimization goal is to improve the modified contrast ratio and the maximum amplitude of the generated peak while respecting the desired location. For this purpose, the most critical element of the RL framework is the reward function.

4.5.1 Reward Function

We proposed the reward function, R , which is composed by three metrics from the vibration scan. R is defined in the following equation:

$$R = w_1 \times C_r' + w_2 \times u_{peak} - w_3 \times \sqrt{(x_{desired} - x_{peak})^2}, \quad (4.3)$$

where w_i is the weight for each metric, C_r' is the modified contrast ratio, u_{peak} is the amplitude of the localized peak, and $(x_{desired} - x_{peak})^2$ is the squared difference between the desired location and the measured location of the peak.

Note that after the supervised pre-training of the actor, the desired peak location $x_{desired}$ and the measured peak location x_{peak} already present a good matching (at some coordinates, the

location varies in the order of ± 2 mm). So it could be assumed that the third term of the reward function is unnecessary. Nonetheless, RL algorithms tend to look for the easiest way to optimize the reward [114].

In the different optimization experiments that were executed, it was observed that the model prefers to increase the amplitude of the localized peak by concentrating the energy of the signal in the main natural frequencies of the aluminum bar. As a consequence, the location of the peak diverges from the desired position. Thus, if this portion of the reward function is removed, x_{peak} becomes quite distant from the desired peak location. To counteract this preferential behavior, a weight (w_i) is added to each term of the reward function and its value is set accordingly. i.e., low values (between 0.5 to 2) are set for w_2 (peak amplitude) and higher values (between 5 and 10) are set for w_1 (modified contrast) and w_3 (location error).

The most exciting aspect of the optimization proposed in this chapter is that it is not necessary to deal with sparse rewards. This advantage comes from the fact that it is possible to evaluate the reward function in a single step.

4.5.2 RL Training

Initial RL Training Experiment

An initial optimization is carried out using the whole dataset. The model starts at location $x_{desired} = 63$ mm and increments the location in steps of 1 mm. $x_{desired}$ changes when the reward is higher than a given threshold. This optimization aims to check that the pre-trained actor can create a peak at all locations. This step was done without significant trouble. Nevertheless, when a reward is too small (below 0), the agent does not improve, and sometimes the reward goes down.

The training results are presented in Fig. 4.20. The performances are pretty constant, which confirms that the pre-trained actor provides homogeneous results. The critic (who has not been pre-trained) fits well and very quickly the expected reward. After evaluating the metrics for different locations, the average of the modified contrast ratio is 1.5, and the max amplitude is $0.33 \mu\text{m}$. The u_{peak} values are bigger than the TrIRDs performances, but the C_r' continues to be smaller. From the learning point of view, it just means that this set of parameters does not make the networks diverge, which can quickly happen in RL.

Training on a Subset of the Beam Locations

After several RL experiments, it was found that the learning is still too slow even with the reduced action space (FFT generation instead of time-domain generation). It means that if the optimization was done over all the locations on the aluminum beam, it is likely that the signals would not be significantly improved. Moreover, when exploring different locations, the following position to optimize is chosen randomly, making it unlikely that the same location

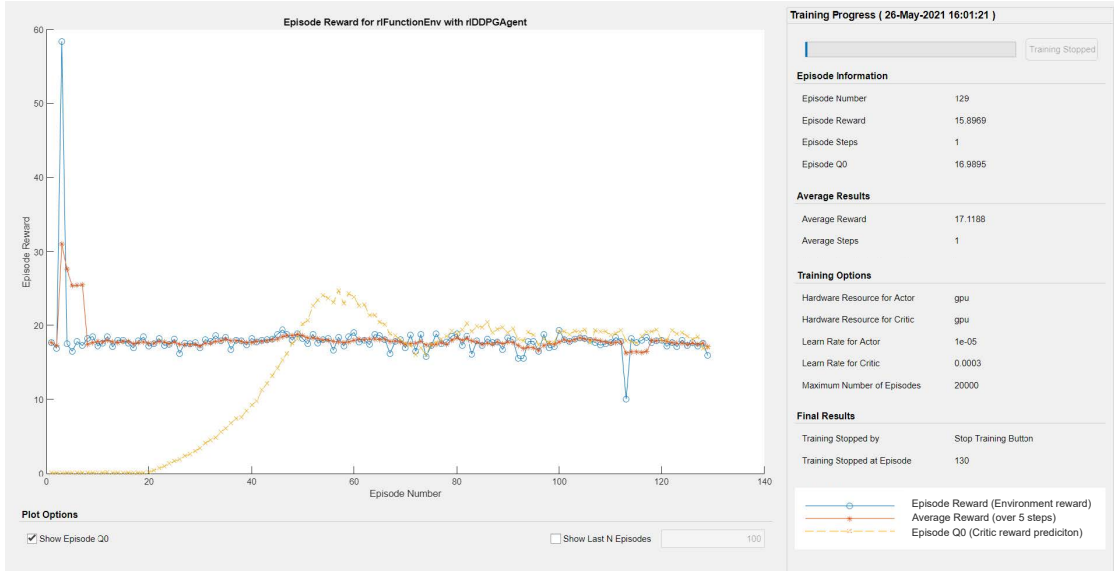


Figure 4.20: Evolution of the reward for the initial RL optimization.

is chosen more than once in a limited time. This behavior is unsuitable for improving the localized peak performance, as different locations may not improve by the same weight.

Since the goal is to improve the signals and not to develop a controller, the following step concentrates on a small subset of locations over the aluminum beam. The RL model is run over five coordinates spaced 5 mm from each other. The experiment starts at $x_{desired} = 100$ mm and randomly moves through the other 4 locations.

Typically, over-fitting is an undesired effect in ML problems. However, in this case, the agent NN will be optimized on a smaller portion of the beam (i.e., the DNN overfits those locations). This strategy reduces the length of the RL experiments and allows us to study the improvement that can be achieved on the localized peak.

It is preferred to choose five different locations and not only one, as it was observed that the reward diverges when training is focused on only one coordinate.

Hyper-parameters: Noise Variance and Learning Rate

The last important element of RL optimization is to find the best hyper-parameters for the optimization. In this case, the hyper-parameters are the noise variance, σ^2 , and the Learning rate for the actor, L_r . The critic learning rate is kept constant at a value of 0.001.

To find the operating range for the hyper-parameters, several experiments were executed. At each experiment, the RL framework was trained for 300 episodes (approximately 2.5 days on a single Nvidia RTX2080Ti GPU). For illustration purposes, two extreme scenarios are presented.

Fig. 4.21 presents the output of the agent for location $x_{desired} = 115$ mm when low values for the hyper-parameters are set during the training ($\sigma^2 = 0.5$ and $L_r = 1 \times 10^{-5}$). And in Fig. 4.22 the hyper-parameters were fixed at higher values ($\sigma^2 = 20$ and $L_r = 1 \times 10^{-2}$).

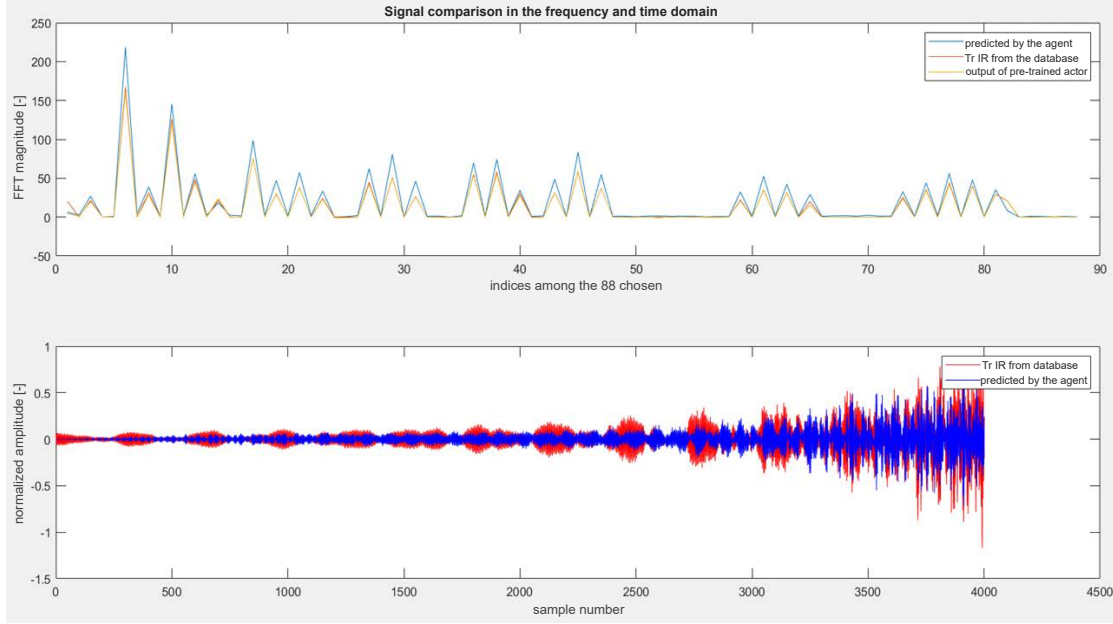


Figure 4.21: Signal generated after an RL optimization with the lowest values for the hyper-parameters. This signal is compared to a reference signal from the TrIRDs ($x_a = 115$ mm).

In both cases, the Frequency-domain signal (the magnitude of the 88 frequencies) and the time-domain signal are presented. In the first case (lowest values), the exploration was very weak, and the performance of the generated peak was similar to the peak obtained with the pre-trained actor. In the second case (highest values), σ^2 and L_r were too high. The reward decreased during learning, going from 16 down to 5. The performance of the generated peak was deficient. The desired coordinate presented a difference of 7 mm, the modified contrast ratio was 20% lower, and the maximal amplitude was $0.05 \mu\text{m}$ smaller when compared to the signal from the pre-trained actor.

During the training, it was observed that the optimization starting point with the pre-trained actor might be a strong local minimum. As a result, much exploration is required to get out of the local solution. Thus, several episodes passed before the reward started to increase.

When exploring different values for L_r and σ^2 , it is observed that: With a small L_r and σ^2 , the model tends to stagnate. For higher values, the model starts to diverge. This means that the optimal value for the hyper-parameters is between these extreme values. For these reasons, the following ranges are defined for the hyper-parameters: A σ^2 between 2 and 10, a L_r between 5×10^{-5} and 5×10^{-3} . The critic stabilizes the learning process with its learning rate of 1×10^{-4} .

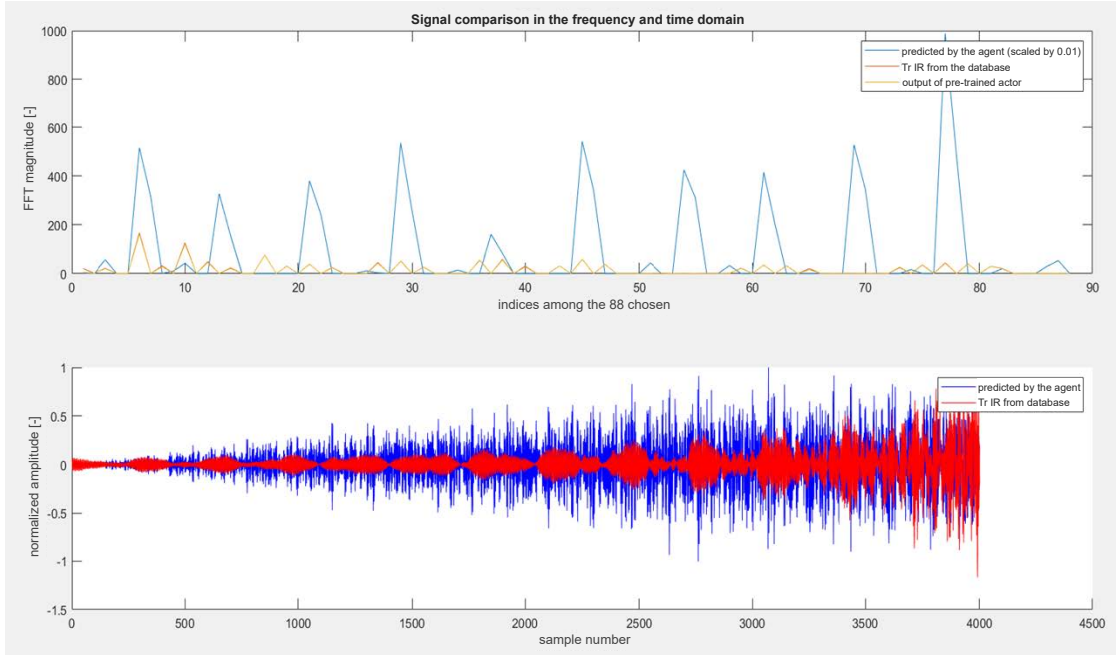


Figure 4.22: Signal generated after an RL optimization with the highest values for the hyper-parameters. This signal is compared to a reference signal from the TrIRDs ($x_a = 115$ mm).

RL Optimization Experiment

After several experiments, it was possible to find a very good set of parameters. The best optimization results, in terms of peak properties and training stability, were obtained for the following parameters: $\sigma^2 = 5$, actor $L_r = 5 \times 10^{-4}$, critic $L_r = 1 \times 10^{-3}$. Using the optimal hyper-parameters, the optimization is focused on coordinates $x_{desired} = [100, 105, 110, 115, 120]$ mm. After running the RL model for nearly 500 episodes (Approximately four days of training on a single RTX 2080Ti GPU system), it was possible to improve the agent and overcome the performance from the localized peak obtained with the original IR signal. In a similar manner, the RL model is trained for locations $x_{desired} = [158, 163, 168, 173, 178]$ mm, where an interesting effect is observed in the optimization of the pre-trained actor. The results for both optimizations are presented in the following section.

4.6 Results and Discussion

After the RL training is finished, the HiL is used for one final time. In this case, the goal is to evaluate the quality of the localized peaks in two desired locations $x_{desired} = 115$ mm and $x_{desired} = 163$ mm. For each position, three different signals are used: the IR from the dataset, the signal from the pre-trained NN, and the signal from the optimized model (i.e., the output of the agent at the end of the Reinforcement learning session).

Chapter 4. Reinforcement Learning for Localized Peak Generation

The complete vibration scan for the desired location, $x_{desired} = 163$ mm, is presented as a contour plot in Fig. 4.23.

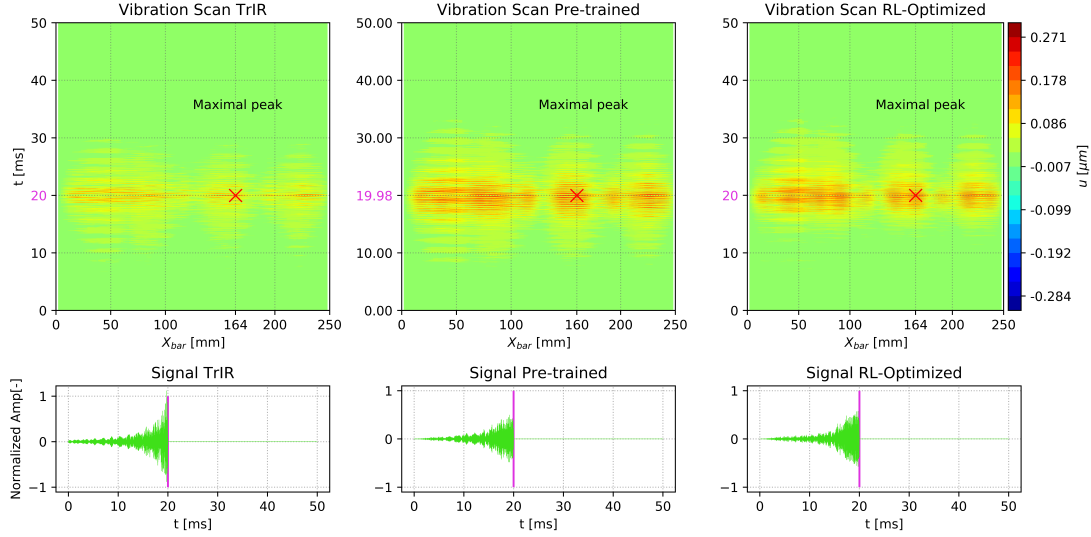


Figure 4.23: Vibration scan using the experimental setup (HiL) to evaluate the quality of three localized peaks at location $x_{desired} = 163$ mm.

Moreover, the measured metrics for both locations are presented in the following tables (Results for $x_{desired} = 115$ mm in Table 4.1 and results for $x_{desired} = 163$ mm in Table 4.2):

Table 4.1: Vibration scan results for the desired coordinate $x_{desired} = 115$ mm.

| Signal | x_{peak} [mm] | u_{peak} [μm] | C'_r |
|--------------|-----------------|------------------------------|--------|
| TrIR | 115 | 0.3333 | 1.7519 |
| Pre-trained | 116 | 0.3667 | 1.5617 |
| RL optimized | 115 | 0.4002 | 1.7838 |

Table 4.2: Vibration scan results for the desired coordinate $x_{desired} = 163$ mm.

| Signal | x_{peak} [mm] | u_{peak} [μm] | C'_r |
|--------------|-----------------|------------------------------|--------|
| TrIR | 164 | 0.2833 | 1.7812 |
| Pre-trained | 160 | 0.3001 | 1.4544 |
| RL optimized | 164 | 0.3167 | 1.5823 |

As expected, the most significant effect of the optimization occurs on the maximum peak amplitude (u_{peak}). Furthermore, the RL optimization ensures that the signal from the optimized actor achieves a peak at the desired location. In some cases, the modified contrast ratio for optimized signals was higher than for dataset signals. In general, the modified contrast ratio increased compared to the C'_r for the pre-trained actor.

For instance, the performances are better for the actor's output at $x_{desired} = 115$ mm. u_{peak} increased from $0.33 \mu\text{m}$ to $0.4 \mu\text{m}$. Moreover, the C'_r improved from 1.75 to 1.78 (2% increase

compared to the traditional approach with the TrIR). These results are obtained for the same driving voltage (the inverse FFT of the generated signal is normalized between -1 and 1, then the signal is amplified to 50 V_{pp}).

At $x_{desired} = 163$ mm, the optimized actor does not over-pass the performance of the original TrIR signal. However, an interesting effect can be observed in the vibration scan. Fig. 4.23 presents the vibration scan for the three signals that are compared. The three evaluated signals are the TrIR from the dataset, the signal from the pre-trained actor, and the signal from the RL-optimized model. One can observe that the RL optimization reduces the amplitude of the vibrations before the instant when the localized peak appears. This modification improves the modified contrast ratio by concentrating the signal's energy on the main peak.

An improvement of 2% might seem like an insignificant amount. Nevertheless, it is crucial to consider that comparing the peak generated by the reference signal (TrIR) with the peak generated by the signal from the optimized actor is not entirely fair. The signals are not equivalent: the TrIR signal contains a broad frequency content. In contrast, the simplified signal (from the agent) contains a limited number of resonant modes (maximum 88 frequencies). One should also consider the information losses associated with the inverse FFT (due to the lack of Phase information).

Then, the most important result is the improvement in the modified contrast ratio when comparing the Pre-trained actor with the RL-optimized actor. In this case C'_r improves by 14.22% for location $x_{desired} = 115$ mm and by 8.79 % for location $x_{desired} = 163$ mm. This result depicts the effect of the RL optimization on the actor NN. It is important to remark that the RL optimization was executed for a limited amount of episodes because of the long duration of each episode. The bottleneck in the training is the vibration scan. Alternative methods to evaluate the peak quality at each episode could allow for better optimizations.

The RL optimization is stopped at this point because the alternative path with deep-generative models seems more interesting in terms of perceptual effects. Nonetheless, the study from this chapter opens the doors to future optimizations. For instance, it would be interesting to experiment with: alternative optimization strategies, different reward functions, and diverse signal representations.

4.7 Conclusion

This chapter presents a new approach to learning and optimizing the Time-reversed Impulse Response (TrIR) signals that can be used to obtain a localized peak on a rigid surface. Two types of Time-reversed (Tr) signals are compared. The IR is preferred to the mechanical impact due to the higher frequency content and the higher quality on the localized peak. A Time-reversed Impulse Response Dataset (TrIRDs) is acquired using an experimental setup with a single piezoelectric transducer. During the acquisition, a simplified strategy to calculate the IR is proposed based on signal processing strategies. The TrIRDs is used to pre-train a

decoder Neural Network. Later, the RL framework is used to optimize the pre-trained NN. During the optimization, an experimental setup is included in the learning loop, and different metrics are defined to evaluate the localized peak in every iteration.

Two desired peak locations are studied. The generated peaks are evaluated using the experimental setup. The chosen metrics show that the most significant improvement appears in the amplitude and localization of the peak. Nevertheless, studying the vibration scan makes it possible to observe that the RL Optimization helps reduce the noise before the instant when the localized peak occurs. Moreover, it was found that in some particular locations, the modified contrast ratio of the optimized NN was improved by 2% compared to the TrIR (traditional approach) and by 14.22% compared to the pre-trained neural network. This improvement was guided by the proposed metrics, especially C'_r that was proposed to evaluate the amplitude of the peak compared to the average maximum displacement at all other locations and during the entire focusing period.

Given the long interval, it takes for one episode (8 min to scan the middle line of the bar) and the ample action space, RL might not be the most efficient approach to optimize the localized peak. Nonetheless, this method made it possible to improve the quality of the localized peaks. Moreover, the proposed approach demonstrated that the TrIR is not the only solution to the peak generation problem and that other combinations of natural modes can lead to good-quality peaks.

This chapter confirmed that it is possible to exploit the FFT representation to simplify the action space. Also, it was demonstrated that NN can learn to generate the TrIR signals and that RL can optimize a pre-trained NN to ensure that the localized peaks appear at the desired location.

In the future, alternative optimization models, signal representations, reward functions, and NN architectures could be explored. In addition, it appears attractive to replace the physical experimental setup with a simplified simulation environment. This modification could lead to a shorter episode duration, allowing for a more efficient and practical optimization framework.

Publications related to this chapter :

- C. H. Mejia, J. Jayet, P. Germano, A. Thabuis, and Y. Perriard, "Linear Impact Generator for Automated Dataset Acquisition of Elastic Waves in Haptic Surfaces," in *2019 22nd International Conference on Electrical Machines and Systems (ICEMS)*, pp. 1–5, Aug. 2019.
- C. Hernandez-Mejia, M. Favier, X. Ren, P. Germano, and Y. Perriard, "Reinforcement Learning and Hardware in the Loop for Localized Vibrotactile Feedback in Haptic Surfaces," in *2021 IEEE International Ultrasonics Symposium (IUS)*, pp. 1–5, Sep. 2021.

5 Generative Adversarial Networks for Vibrotactile Feedback

5.1 Introduction

As discussed in Chapter 2, the time-reversal method and other wave-focusing strategies have been used to render vibrotactile feedback on rigid surfaces. The interaction between actuators and surface response has also been studied extensively and successfully. Nevertheless, the power consumption and the number of actuators needed to obtain a localized peak can still be decreased.

The research questions for this chapter are *"How can Deep-Generative Models be used to generate time-reversed signals that can create a localized displacement peak?"* and *"What is the effect of the diversity of GANs on the properties of the localized peaks?"*. The associated hypothesis is that *"Deep learning can capture the distribution of an Impulse Response (IR) dataset and generate diverse signals that can have a positive effect on the human perception of localized vibrations"*.

After showing that Deep Neural Networks (DNNs) are able to model a simplified version of the Time-reversed Impulse Response (TrIR) signal and that this signal can generate a localized peak with similar or slightly better quality compared to the original TrIR signal, it is interesting to experiment with more advanced Deep-Generative models. Generative Adversarial Networks (GANs) are chosen over other state-of-the-art generative models because they offer the highest quality-to-diversity ratio. Nonetheless, they still provide some diversity, meaning that the generated signal is not always the same.

We are considering profiting from the diversity GANs provide on the generated signals to reduce the power consumption or the number of actuators. The main novelty of our approach is that the localized vibrotactile feedback would be created with diverse signals that the GAN generates in contrast with the traditional approach of repeating the same TrIR signal to obtain a localized vibration. Thus, the expectation is that the peaks are not generated in the exact same position but instead randomly distributed around the desired location, which could better catch the user's attention for smaller displacements.

This chapter studies how GANs can capture the distribution of a dataset containing time-reversed impulse response signals in a Gaussian-like "latent space" (Section 2.9.3). Once the model is trained, one can randomly generate signals to make the target point vibrate locally at the desired rate.

The study is divided into two main stages. In the first stage, the ability of GANs to capture the probabilistic distribution of the Time-reversed Impulse Response Dataset (TriRDs) is studied. After that, one can study the viability of generating new signals in real-time so that these signals can be used to obtain a localized vibration. In the second stage, a condition will be added to the GAN to decide which type of signal is synthesized, thus, where the localized peak appears. Finally, the Hardware-in-the-Loop (HiL) experimental setup from Chapter 4 is used to evaluate the generated peaks. This evaluation allows us to validate the effect of diversity on the location and characteristics of the generated localized peak.

The potential effect of diversity on the human perception of the generated peaks is further discussed at the end of this chapter. This effect is validated in Chapter 6.

5.2 Generative Adversarial Networks (GANs)

Before starting to experiment with the GANs, it is essential to introduce the different elements and the working principle of the GAN framework. Furthermore, it is key to discuss the recent GAN models that are being used for time-domain data and to motivate the selection of the GAN architecture that is used in this chapter.

5.2.1 GAN Definition and Working Principle

The GAN framework is a deep learning strategy to train generative neural networks. It was first introduced by Ian Goodfellow *et al.* in 2014 [78]. This framework allows training DNNs to capture the probabilistic distribution of a given dataset and then generate new data points that look like the ones in the dataset.

The GAN framework aims at simultaneously training two models, typically two multilayer perceptrons (i.e., a "Vanilla" Neural Network), which interact in an *adversarial* manner. The first model, defined as the generator G , learns to represent the distribution of a real dataset p_{data} . In essence, it learns how to up-sample from a lower-dimensional vector, z , that is drawn from a Gaussian noise distribution, Z ($z \sim p_Z$), into a higher-dimensional vector, $G_{out} = G(z)$, that looks like a real data sample. The second model, called the Discriminator, D , takes an input D_{in} and estimates the probability that D_{in} comes from either the real data distribution p_{data} or from the Generator $G(z)$. In other words, D learns to classify examples as either real (from the dataset) or fake (from the generator).

Both models are trained simultaneously in a two-player minimax game. The parameters of G are optimized to minimize the value function $V(D, G)$, while the parameters of D are

optimized to maximize it:

$$V(D, G) = \mathbb{E}_{x \sim p_{data}} [\log D(x)] + \mathbb{E}_{z \sim p_z} [\log (1 - D(G(z)))], \quad (5.1)$$

where \mathbb{E} represents the expectation of the equation, x is the real data sample drawn from the probability distribution of the data set p_{data} , and z represents the random noise vector drawn from a Gaussian noise distribution p_z . $D(x)$ indicates the probability that D discriminates x as real data, and $D(G(z))$ indicates the probability that D determines the data generated by G . The goal of D is to determine the data source correctly, so it wants $D(G(z))$ to approach 0, while the goal of G is to bring it closer to 1.

As the learning progresses, the generator and the discriminator improve their performance. At this stage, the generator can synthesize data samples $G_{out} \sim p_G$ that become indistinguishable from the dataset $x \sim p_{data}$ in the view of the discriminator. Furthermore, new samples can be continuously drawn from the model because G models p_{data} by up-sampling from a continuous latent space p_z . Therefore, GANs can be trained in an unsupervised manner where new data can be generated by inputting a latent vector z to the trained $G(z)$. The general approach to training the GAN is illustrated later in Fig. 5.1(a) (Section 5.3).

With the traditional GAN framework, there is no control over the type of sample generated. Nevertheless, when a condition c is given at the input of G and D , and the model is trained in a semi-supervised fashion, then it is possible to obtain a conditional generative model [78, 118]. For the conditional GAN (cGAN), the value function becomes:

$$V(D, G) = \mathbb{E}_{x \sim p_{data}} [\log D(x|c)] + \mathbb{E}_{z \sim p_z} [\log (1 - D(G(z|c)))]. \quad (5.2)$$

The conditional GAN allows us to decide the exact subset of the probability distribution that has to be sampled. In other words, it is possible to control the generation process. For example, with the cGAN trained on the TrIRDs, it is possible to determine the location (x_a) where the peak should appear.

5.2.2 GANs for Time-domain Signals and Model Selection

The GAN framework has become increasingly popular since it was presented in 2014. It has been especially used for image processing applications where it yields a high resolution, and high-quality results [119, 120, 121]. Gradually it has shown potential in the raw signal domain [122]. In 2019, Donahue *et al.* first introduced the use of GANs for audio generation and processing [82]. They proposed some adjustments to the Deep Convolutional GAN (DCGAN) model [123] to capture the long-term and short-term relations present in raw audio signals, where the main changes occur along the temporal axis. In [82], Donahue *et al.* presented two approaches for capturing the unconditioned distribution of raw-audio data. The first

approach, Spectrogram GAN (SpecGAN), uses the DCGAN capabilities to generate images. The raw audio signals are represented as images in a pre-processing phase by converting the time-domain signal into an amplitude spectrogram using the Short-Time Fourier Transform (STFT). The inverse STFT is then applied to extract the time-domain signal after the generating stage. The phase information is recovered using the iterative Griffin-Lim algorithm. The other approach, called Waveform GAN (WaveGAN), works directly on the raw audio representation by converting the 2D convolution from the DCGAN model into a 1D convolution. WaveGAN is able to generate the time-domain audio signal without using an intermediate representation.

Alternative techniques employing state-of-the-art architectures and training algorithms have been proposed, further improving the quality of raw audio generation. For instance, Dieleman *et al.* developed the GANSynth model [124]. In this model, they adopt the Progressive Growing Wasserstein GAN (PGGAN) and propose an alternative data representation. In essence, they use the instantaneous frequency to capture the phase of signals, and the model produces excellent results for highly harmonic signals (e.g., speech and musical instrument sounds). Moreover, Nistal *et al.* [83, 84] proposed the DrumGAN model. In this case, they exploit the same PGGAN architecture. However, they discover that for impact-like signals (drum beats) where the phase is extremely chaotic, representations such as the waveform and FFT complex representation yield the best results.

Both WaveGAN and DrumGAN have been successfully applied in modeling percussive instrument recordings. There is a significant similarity between the nature of a time-reversed impact signal and a drum-beat audio signal (i.e., Drum sound effects). The time-reversed impact signals that have been used to create localized peaks on a rigid surface [51, 56], and a drum beat are, in essence, the same phenomenon. In both cases, impulsive stimuli are applied to a solid medium, initiating the wave propagation that will stimulate the resonant frequencies of the object. From the acquisition point of view, the stress or pressure waves are recorded after the stimulus is applied.

After several experiments with different GANs models and knowing that one of the goals of this chapter is to evaluate the ability of GANs to model TrIR signals and real-time data generation, the WaveGAN model from [82] is chosen. WaveGAN is the simplest model and provides the best results for our dataset. The simplicity of its architecture allows for an easy transformation into a conditional model. For these reasons, it is the base architecture for our experiments. In future work, more complex models can be studied to improve the generated signals' quality.

5.2.3 Generated Signal Evaluation Metrics

Among deep learning researchers, the quantitative and qualitative metrics used to evaluate the quality of GAN-generated data are still in development. Typically, qualitative metrics are evaluated by expert reviewers (i.e., human evaluators who are familiar with the data being modeled). Recently, many quantitative metrics have been proposed [121, 125, 126], and they have been compared with human perception. Some metrics involve statistical methods to

measure the distance between the generated and real distribution (e.g., 2-Wasserstein or Fréchet distance). An interesting metric is known as the "inception score" [127]. This metric evaluates the accuracy and diversity of the generated data. Using a pre-trained classifier, the inception score can penalize models that do not generate relevant data or lack diversity. In other words, the inception score penalizes models that generate samples that are not easily classified into a set of known categories (lack of accuracy) and models that generate samples that belong to only a few known categories (lack of diversity).

Due to the absence of human expertise in evaluating time-reversed IR signals, it is crucial to develop alternative methods and metrics. In this work, we propose low-level feature evaluation metrics. The proposed metrics are defined on the foundation of specific domain knowledge that was gathered in the previous chapters and using information acquired with the experimental setup. The metrics are:

- The Location of the main peak after a vibration scan.
- The Contrast ratio (i.e., signal-to-noise ratio) of the peak after a vibration scan.
- The Frequency content related to the natural frequencies of the bar.
- The Cross-correlation between the generated signal and a reference signal from the TrIRDs.

5.3 Experiments with GANs

5.3.1 Methodology

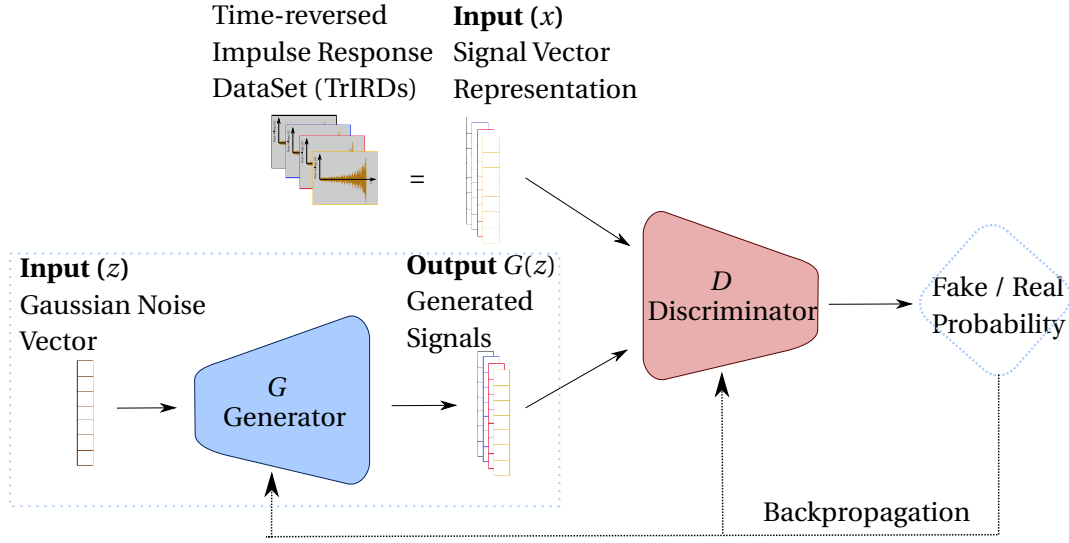
To validate the hypothesis and answer our research questions, this study is carried out in two stages. The first stage concentrates on evaluating the ability of WaveGAN to capture the TrIRDs, as well as the feasibility of real-time signal generation. The second stage focuses on developing the conditional WaveGAN (cWaveGAN) and evaluating the accuracy of the generated peaks. More importantly, it studies the effect of the diversity of the GAN on the localized peaks.

In both stages, the GAN is first trained, and then the generator is used to synthesize a signal. Then, this signal is transferred to the HiL experimental setup to evaluate the obtained localized peak. The general approach is illustrated in Fig. 5.1.

5.3.2 Stage 1: Unconditional GAN

For the first stage, an existing GAN architecture is trained using the TrIRDs. In this case, a cropped version of the raw signal is used, i.e., the model is trained to create the voltage signals that drive the piezoelectric actuator. At this point, the generation is unconditional, meaning there is no control over the location where the peak will appear. For this reason, the generated signals are sent to the HiL experimental setup to find the peak's location and evaluate the localized peak's quality (contrast ratio). The generated signal is compared to a reference signal from the TrIRDs that can obtain a peak in the exact location. This comparison is made in the

(a) GAN Training



(b) Signal Generation (Inference)

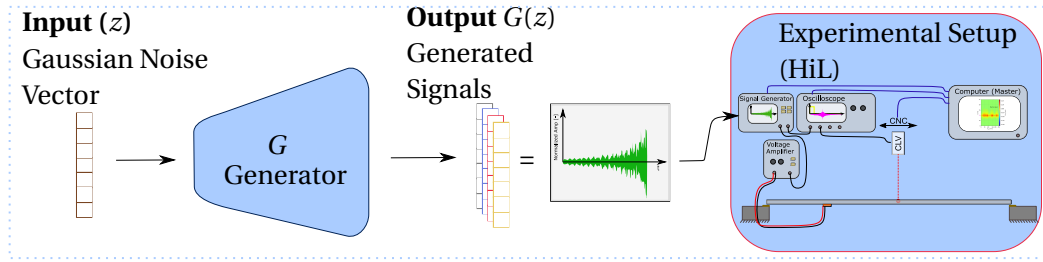


Figure 5.1: General approach to train and evaluate the GAN model. (a) GAN model training with the TrIRDs. (b) Signal generation and peak evaluation using the HiL experimental setup.

time and frequency domain. Furthermore, the characteristics (i.e., contrast ratio, amplitude, and location) of the localized peaks for both signals are compared.

Afterward, the experimental setup is used to validate that the trained model can generate, in real-time, the signals to obtain a localized vibration within the human perception range (200 Hz - 300 Hz [30]). To do this, the total time that it takes to generate the signal, to transfer the signal, and to output the voltage into the piezo is measured.

WaveGAN Architecture

The WaveGAN model is based on the DCGAN Architecture, which uses transposed convolutions to upsample from a latent space into a realistic data sample (e.g., image, waveform). The main differences between WaveGAN and DCGAN are: First, the 2D convolutions with size $[5 \times 5]$ and stride $[2 \times 2]$ are flattened into 1D convolutions with length 25 and stride 4. Second, the training of the GAN is done using the Adam optimizer [107] and the Wasserstein

GAN with Gradient Penalty (WGAN-GP) loss function proposed by Gulrajani *et al.* [128]. The main difference with the original GAN strategy is that in WGAN-GP, the discriminator is not exactly classifying the samples as real or fake. Instead, it measures the Wasserstein-1 distance between the dataset and a batch of generated samples probability distributions. This strategy is more stable and was demonstrated to successfully train a variety of model configurations where other GAN losses fail.

In addition, an operation called "phase shuffle" was added to permit fair competition between the G and D . This operation is needed because the generated signals present some "artifacts" (i.e., unwanted patterns created by the upsampling convolution) in the pitch of certain frequencies. This behavior is similar to the "checkerboard" effect (i.e., an artifact with the appearance of a chess board) of GANs for image generation. The problem with these artifacts is that the discriminator can easily detect them, so it would inhibit the optimization during the adversarial training. The phase shuffle is only applied to the discriminator, and it works by perturbing the phase of the signal inside the different convolutional layers. The phase is perturbed by a factor of $-n_{phase}$ and n_{phase} and the operator can be activated and deactivated (i.e., n_{phase} and the active or inactive status are training hyper-parameters).

The WaveGAN Generator starts with a latent vector, z , of dimension n_z . These values are passed to a fully connected layer and then re-shaped. Later, the re-shaped tensor is passed through five upsampling-convolutional layers until an output, G_{out} with $n_{G_{out}} = 16384$ samples is obtained (i.e., the output of G is a time-domain signal with $n_s = 16384$ samples). The ReLU activation function is used through all the convolutional layers of the model except for the last one, where the Hyperbolic tangent (TanH) activation function is used.

The WaveGAN Discriminator is a mirror of G . The input of the model is a signal from G or a signal (x) from the dataset with length $n_s = n_{G_{out}}$. This data is passed through 5 downsampling-convolutional layers with the Leaky ReLU (LReLU) activation function. Each layer contains the phase shuffle operation with parameter n_{phase} . Finally, the data is resized and passed through a fully connected layer with a single output (i.e., the calculation of the Wasserstein distance).

The complete architecture of G and D for the WaveGAN model are described in Appendix A.

Data Preparation and WaveGAN Model Training

The TrIRDs, described in Section 4.2.1, is reused in this chapter. A total of 5310 Impulse Response (IR) signals were acquired with a sampling rate of $f_s = 200$ kHz. In this case, the sampling period is $T_s = 81.92$ ms which is equivalent to a signal length, $n_s = 16384$ samples for each signal. The size n_s is chosen to match the original output dimension of the WaveGAN architecture ($n_{G_{out}}$) [82].

Tensorflow [106] is used to implement and train the model. The training data is directly supplied from the TrIRDs dataset folder. In this case, the data labels are not used (i.e., x_a is not used), so the samples are drawn randomly from the dataset in batches of 64 data examples

(i.e., the batch size is $n_{batch} = 64$). The model is trained with a latent vector dimension, $n_z = 100$, and the phase shuffle parameter, $n_{phase} = 2$. The latent vector is drawn from a Gaussian distribution with mean, $\mu_z = 0$, and standard deviation, $\sigma_z = 1$. The complete list of hyper-parameters is presented in Appendix A. The training goes through 250k iterations, which takes about 4.5 days on a single GPU system. It should be mentioned that after 100k iterations (1.5 days), the model could produce signals with a similar waveform shape and frequency content as the real dataset. The training stops when the losses for G and D are close to zero, and the training error has been constant for several iterations.

WaveGAN Signal Generation

After the model is trained, one or n examples of time-reversed signals can be synthesized by feeding an array of n latent vectors z drawn from the "continuous uniform distribution" (i.e., the same distribution that was used during training with the same μ_z and σ_z). The generation process was illustrated in Fig. 5.1(b). To provide an idea of the WaveGAN output, two signals are randomly selected from a batch of 100 generated examples (WaveGAN S0 and S2). These signals are compared with a reference signal from the TrIRDs and presented in Fig. 5.2.

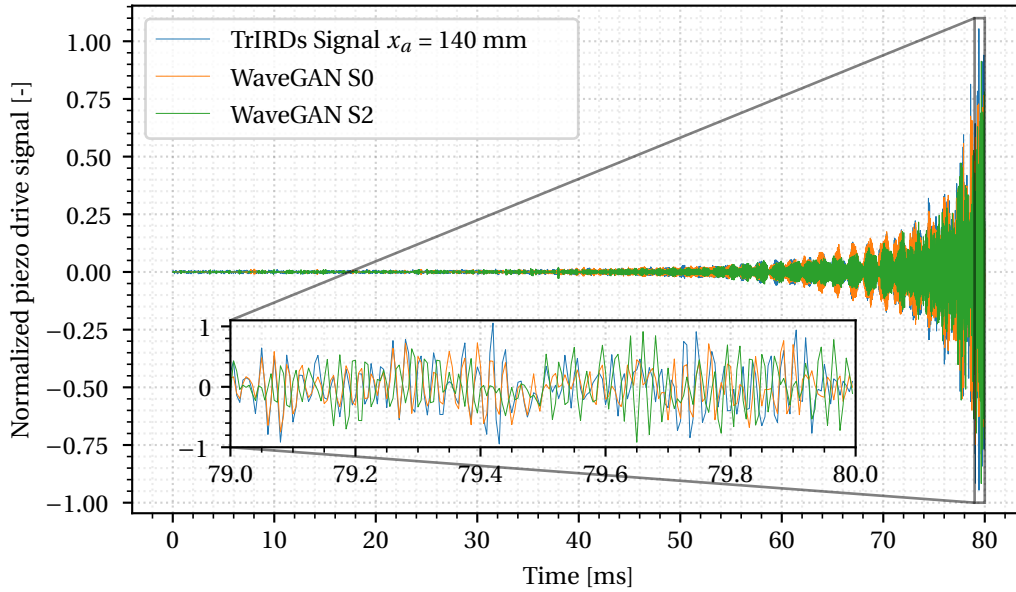


Figure 5.2: One reference signal from the TrIRDs ($x_a = 140$ mm) and two randomly picked signals (S0 and S2) from a batch generated by the WaveGAN model trained on the TrIRDs.

Note that all the signals are cropped to 80 ms (16000 Samples) to facilitate the visualization. The first 384 time samples of the signal can be removed since their contributions to the peak generation are negligible.

The signal from the TrIRDs with peak location, $x_a = 140$ mm, was chosen after carrying out a vibration scan for signal S0 from the WaveGAN. This selection allows two relevant signals to be compared and one different signal to be contrasted. This contrast permits the visualization of the GAN's diversity (i.e., to confirm that the GAN is not only generating one type of signal).

WaveGAN Generated Signal Evaluation

The signals appear to be very similar in the time domain (Fig. 5.2). Looking closely, one can see that the signal from the TrIRDs and WaveGAN signal S0 match quite well, while WaveGAN signal S2 is different. The differences between the signals are more notorious after $t = 60$ ms. The temporal analysis is not enough to conclude the similarity of the two signals, but different metrics can measure it.

In signal processing, the most common metric is cross-correlation: "the measure of similarity between two series as a function of the displacement of one relative to the other." The cross-correlation is calculated as the sum of the dot product between the two compared signals, $x(t)$ and $y(t)$: $corr(x, y) = \sum_{i=0}^{n_s-1} x[i] * y[i]$, where, $x[i]$ and $y[i]$ are the i^{th} sample of signals x and y respectively. In cross-correlation, the higher the scalar value is, the higher the similarity between the two compared signals.

Another interesting metric is the L^2 norm. This metric calculates the distance of the vector coordinate from the origin of the vector space and is generally related to the signal's energy. The L^2 norm is calculated as the Square Root of the Sum of Squares (SRSS): $L^2(x) = \sqrt{\sum_{i=0}^{n_s-1} x[i]^2}$. We propose to use the difference between the L^2 norm of two signals to compare how close they are to each other in the vector space. Table 5.1 presents the result for both metrics in the three evaluated signals.

Table 5.1: Cross-correlation, L^2 norm, and L^2 norm difference for the reference signal from the TrIRDs ($x_a = 140$ mm) and WaveGAN signals S0 and S2.

| Signal | $corr(\text{Ref}, \text{Signal})$ | L^2 norm | L^2 norm difference |
|------------|-----------------------------------|------------|-----------------------|
| Ref TrIR | 72.06 | 8.49 | - - - |
| WaveGAN S0 | 54.96 | 7.76 | 0.73 |
| WaveGAN S2 | -20.30 | 7.52 | 0.97 |

The L^2 norm shows that the reference TrIR signal (Ref) and WaveGAN signal S0 are closer in the vector space, compared to signal S2. Moreover, the cross-correlation between Ref and S0 is higher than the cross-correlation between Ref and S2. These results confirm the higher similarity between the reference TrIR signal (Ref) and WaveGAN signal S0 (the signals correlate). The lower similarity between Ref and S2 is a good indicator of the difference (the signal does not correlate).

The graphical comparison is easier in the frequency domain. The magnitude of the FFT for the three compared signals are presented in Fig. 5.3.

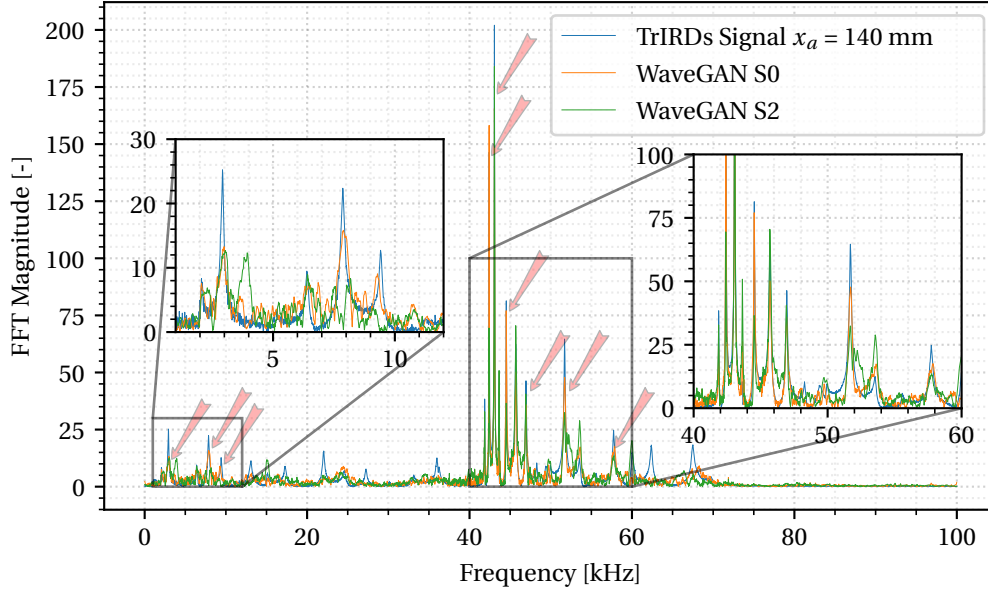


Figure 5.3: Magnitude of the FFT for the reference TrIR signal and WaveGAN-generated signals S0 and S2. The red arrows highlight the prominent peaks of magnitude where the variation of the FFT magnitude for the different signals is observed.

On the one hand, it is possible to observe that both the reference signal and WaveGAN S0 signals contain prominent peaks of magnitude at the same frequencies (red arrows highlight the most relevant peaks). The magnitude values are similar but not the same. This difference indicates that the GAN is not learning the exact same signal but is capturing the dataset's distribution. On the other hand, it can be seen that the magnitude values for the WaveGAN S2 signal differ from the other two signals and some peaks appear at different frequencies. Following the same approach as in the time domain, Table 5.2 presents the result for the cross-correlation and L^2 norm for the magnitude of the FFT of the three evaluated signals.

Table 5.2: Cross-correlation, L^2 norm, and L^2 norm difference for the FFT magnitude of the reference signal from TrIRDs ($x_a = 140$ mm), Ref, and WaveGAN signals S0 and S2.

| FFT Magnitude | $corr(FFTRef, FFTMag)$ | L^2 norm | L^2 norm difference |
|---------------|------------------------|------------|-----------------------|
| Ref TrIR | 57.65×10^4 | 759.27 | 0.00 |
| WaveGAN S0 | 49.99×10^4 | 694.10 | 65.17 |
| WaveGAN S2 | 44.43×10^4 | 672.48 | 86.79 |

The results for the L^2 norm and the cross-correlation in the frequency domain are very similar to those in the time domain. The similarity of the Ref signal and WaveGAN S0 signals is higher than Ref vs. S2. Nonetheless, the correlation of Ref vs. S2 in the frequency domain is higher compared to the time domain, which makes sense because in the frequency domain is easier

to see that all IR signals stimulate the natural modes of the bar. The cross-correlation of a signal with itself is known as auto-correlation. The difference between the auto-correlation of Ref and the cross-correlation of Ref with signals S0 and S2 is a good indicator to show that the GAN model is not only generating one type of signal but is covering the whole dataset (i.e., there is no "mode collapse" during the training).

The next step is to compare the localized peaks obtained with the three signals. To do this, the HiL experimental setup introduced in Chapter 4 is used to obtain the vibration scan. Generation and evaluation procedures were illustrated in Fig. 5.1(b). The vibration scan is carried out from $x_{start} = 0$ mm until $x_{end} = 250$ mm, with a spatial resolution of $R_{scan} = 2$ mm.

After the vibration scan, the signals are superposed and analyzed in the time domain. The maximal displacements (i.e., the main peak) are located, and the wave propagation on the surface of the aluminum bar can be studied.

To evaluate the proposed metrics (Section 5.2.3), the data from the vibration scan is organized as a 3D matrix $M_{t,x_a,u}$ where time (t), scanned location on the bar (x_a) and displacement amplitude u are the main axes. The measured peak location x_{peak} and the peak instant t_{peak} can be located by projecting the maximal displacement across the t and x_a axes.

The vibration scan for WaveGAN signal S0 is presented in Fig. 5.4. One can observe the displacement at the middle-line of the aluminum bar over time. The sub-plot on the right shows the vibration scan contour visualization (i.e., the top view of the displacement at every scanned bar position over time). The maximum displacement (localized peak) is marked with an X. The Upper-Left plot shows the normalized input signal, and two colored bars indicate two different instants, t_1 and t_{peak} . The two Lower-Left plots present the displacements occurring in the aluminum bar at t_1 and t_{peak} . The peak appears at location $x_{peak} = 140$ mm right after the generated signal has been reproduced ($t_{peak} = 79.98$ ms).

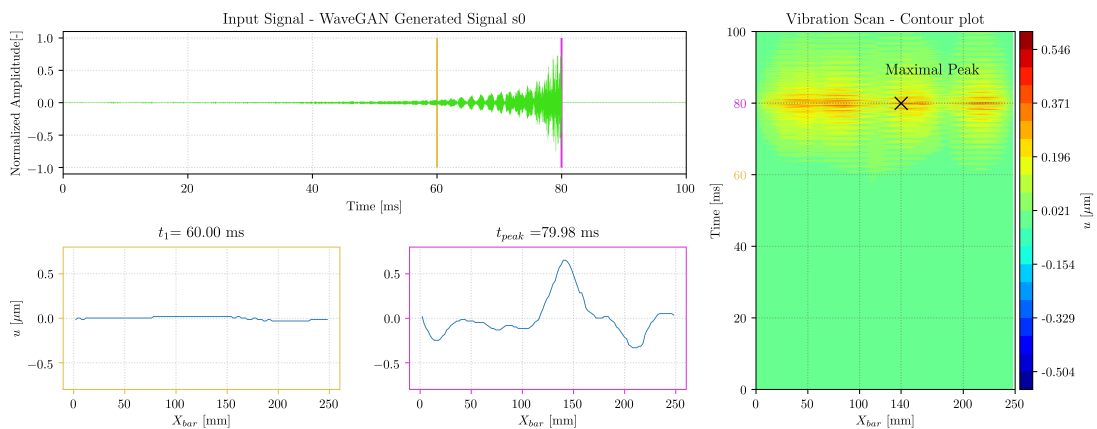


Figure 5.4: Vibration scan for WaveGAN signal S0. The localized peak is displayed with a black X.

To validate the quality of the generated peak, the original contrast ratio, C_r , is used. As discussed in Chapters 2 and 4, C_r is defined as the ratio between the maximum displacement u_{peak} and the Root Mean Square (RMS) displacement at all locations x_a at instant t_{peak} . C_r is calculated as follows:

$$C_r(t = t_{peak}) = \frac{u_{peak}}{\sqrt{\frac{1}{n_{scan}} \sum_{x_a=x_{start}}^{x_{end}} (u(x_a, t))^2}}, \quad (5.3)$$

where n_{scan} is the number of locations x_a for the vibration scan (for this study $n_{scan} = 125$, i.e., the length of the bar, $l = 250$ mm divided by $R_{scan} = 2$ mm), $x_{start} = 0$ mm and $x_{end} = 248$ mm.

The results for the metrics after the vibration scan are presented in Table 5.3. This table contains the evaluated metrics for the reference signal from the TrIRDs and three signals from the WaveGAN model (an additional WaveGAN signal is studied, signal S1).

Table 5.3: Vibration scan metrics for three signals from WaveGAN and the reference TrIR signal ($x_a = 140$ mm).

| Signal | x_{peak} [mm] | u_{peak} [μ m] | t_{peak} [ms] | C_r |
|-----------------------|-----------------|-----------------------|-----------------|--------|
| TrIRDs $x_a = 140$ mm | 140 | 0.5667 | 79.99 | 2.8585 |
| WaveGAN S0 | 140 | 0.61667 | 79.98 | 2.7680 |
| WaveGAN S1 | 148 | 0.55 | 79.99 | 2.8218 |
| WaveGAN S2 | 220 | 0.51667 | 79.99 | 2.7511 |

The peak obtained with signal S0 is comparable to that obtained from the TrIRDs with $x_a = 140$ mm. These results confirm that the WaveGAN model can capture the distribution of the TrIRDs and that the generated signals can be used to obtain a localized peak. The peak amplitude for signal S0 is slightly higher (10 %), but the contrast ratio is reduced by 3.2 %. In addition, the peak appears 10 μ s earlier compared to the other signals, but this is not very critical at this stage. In contrast, the vibration scan for the generated signal S1 and S2 present peaks with similar properties to the reference signal. However, they appear at different locations, which confirms that the GAN is not constantly generating the same type of signal (no "mode collapse").

Real-time Generation and Localized Vibration Feasibility

Up to this point, it has been demonstrated that GANs can capture the TrIRDs distribution and generate time-reversed IR signals. When the generated signals are fed into the system, localized peaks occur.

The next important step is to evaluate the potential of WaveGAN for real-time signal generation. Furthermore, to validate the potential to generate localized vibrations within the range of human perception thresholds (200 - 300 Hz [30]). In order to get a 200 Hz localized vibration,

the Tr signal's generation and reproduction should occur in less than 5 ms (i.e., one peak should be obtained every 5 ms).

The experimental setup and the trained GAN model are used to validate this. A single signal can be generated in 3.94 ms using the Trained Generator from Section 5.3.2. Then, the reproduction time, i.e., the time it takes to transfer and play back the signal, takes slightly less than 2 ms. Adding both periods results in a total time of 5.94 ms, slightly longer than the desired 5 ms. An alternative is to generate the signals asynchronously. This alternative can be done because generation with GANs is fully parallel [82] (thanks to the parallelization of GPU processing). Thus, generating a batch of signals is much faster than iteratively generating them one by one. For the trained WaveGAN, it takes 340 ms to generate 1000 signals (i.e., 0.34 ms per sample). In this manner, obtaining a localized vibration in the desired frequency range is possible.

Further improvements can be obtained in the future by optimizing the current evaluation setup. For instance, the size and number of parameters of the GAN model can be reduced so that the generation times can be shortened. Alternatively, it is possible to simplify the signals by decreasing the signal length or by generating a "1-bit quantized" version as in [51] (1-bit quantization is described in Chapter 6). Thirdly, signal generation hardware can be directly integrated into the system where the generation takes place.

5.3.3 Stage 2: Conditional GAN

As observed in the previous section, one of the limitations of an unconditional GAN model is that there is no control over the generated signal. Thus it is impossible to decide where the peak will appear (i.e., G may generate a peak on any location within the modeled domain). For this reason, the GAN model is modified for conditional generation in the second stage. In this case, the desired location $x_{desired}$ is given as an additional input to the model. Once the model is trained, several signals can be generated for a given location, and the effect of diversity on the properties of the peaks can be evaluated.

Development of the Conditional WaveGAN

There are different strategies to control the data generated by the GAN. One option is to find the relationship between points in the latent space (i.e., the different values of the input vector z) and the generated signals. However, this relationship is complex and hard to map. Another option is to present the Generator and Discriminator with a condition. This condition guides the model to generate data samples in the desired domain (e.g., images of a given type [118], sounds of a given nature [129] or TrIR signals for a given location). This condition is more explicit than manipulating the latent space.

There are two primary motivations for using a condition in the GAN framework. The first and most obvious one is to get control over the generation. The second one is to improve the GAN,

thus the quality of the generated samples. Additional information correlated to the dataset examples can help make the training more stable, reduce the training time, and get higher quality on the generated samples.

As mentioned in Chapter 2, the condition can have different natures. It can be a continuous or a discrete label (e.g., peak desired location, $x_{desired}$), or it can be another signal or an image (e.g., image-to-image translation).

The simplest conditioning method is the "class label" or "discrete condition". It implies selecting a discrete number, n_{labels} , of desired locations, $x_{desired}$, and using them as a condition to get time-reversed signals of that type. In our case, a continuous condition would make more sense than a discrete condition given that space is a continuous variable. Nonetheless, it requires much more data; since the model needs to see as many samples on the continuous range as possible (Several attempts were carried out with the continuous label, but the results were not as successful as for the discrete label).

The best practice to include the class label in the GAN involves using an embedding layer followed by a fully connected layer with a linear activation. This layer scales the embedding dimension to the size of the target data sample. The scaled feature map can then be concatenated to the model as an additional channel of the convolutional layer [130, 131].

Based on this conditional approach, we modified the original WaveGAN architecture to include the class conditioning input.

For the generator, the peak desired location $x_{desired}$ is given as an additional input (i.e., the conditioning label, $c = x_{desired}$). At this point, the NN for G has two parallel input branches that will be merged before passing through the upsampling-convolutional layers. In the first branch, the latent vector z is fed into a fully connected layer with an output of shape $[n_{G_{out}}, 1]$ ([16384, 1]) and then reshaped to size $[16, n_{G_{out}}/16]$ ([16, 1024]), same as in the WaveGAN model. In the second branch, c is mapped into a one-hot encoder. A one-hot encoder is a binary vector with a length equal to $n_{labels} = 177$ (number of unique desired locations on the aluminum bar), where one value of the vector is equal to 1 in the desired location and 0 elsewhere. This one-hot encoder is mapped to a higher dimensional vector with shape $[n_{labels} \times 20, 1]$ ([3540, 1]) (i.e., embedding layer). Lastly, the obtained tensor is passed through a fully connected layer and re-shaped to size $[16, 1]$. At this point, the second branch can be merged into the first one, and the resulting output has a shape $[16, 2049]$, which is passed to the upsampling-convolutional layers. From this point on, the architecture remains the same as in the original WaveGAN implementation.

For the discriminator, the integration of the label is much simpler. The conditioning label, c , is mapped into a one-hot encoder. Then, it is mapped to a higher dimensional vector (i.e., embedding layer) and passed through a fully connected layer. Finally, this tensor is re-shaped to size $[n_{G_{out}}, 1]$ ([16384, 1]). The embedded condition is concatenated with the audio input, which has a shape of $[n_s, 1]$ ([16384, 1]). From this point onward, the architecture remains

the same as in the WaveGAN model. The detailed architecture for the conditional WaveGAN (cWaveGAN) is presented in Appendix A.

Data Preparation and Conditional WaveGAN Training

The same dataset from the first stage is used to train the cWaveGAN model. Nonetheless, the dataset is classified by desired peak location and reorganized in folders with x_a as the name. The model loads this data, and each folder's name is fed to the GAN as c . Following the same approach from Chapter 4, several experiments are executed to find the best hyper-parameters. Some hyper-parameters were modified to ensure the stability of the GAN during training and improve the quality of the generated samples (Compared with WaveGAN training). First, the learning rate for the Adam optimizer on both G and D is set to $L_r = 2.5 \times 10^{-4}$ (On WaveGAN $L_r = 1 \times 10^{-4}$). This change accelerates the training process and avoids getting stuck in local minima during the optimization. Second, n_{batch} , is set to 128 samples (on WaveGAN $n_{batch} = 64$). This change ensures that the model can see different signals during each iteration. Third, the phase shuffle is removed, $n_{phase} = 0$ (on WaveGAN n_{phase} was 2). The phase shuffle was deactivated because it was inducing noise in the signals. And fourth, the batch normalization is activated, $batch_{norm} = \text{True}$ (on WaveGAN was $batch_{norm} = \text{False}$). The batch normalization helps to accelerate the training and to ensure stability. The rest of the hyper-parameters remain constant. For reference, the complete list of hyper-parameters is presented in Appendix A.

The model is trained for 30k iterations, which lasted for 1.5 days on a single GPU machine (Nvidia RTX2080Ti GPU). As expected, thanks to the conditioning label, the training is much faster than the WaveGAN model, which took 250k iterations and 4.5 days. Moreover, after 1000 iterations (around 1.5 h), the cWaveGAN model is already generating signals coherent with the original dataset (similar waveform). This confirms the expected positive effect of the condition, c , on the training of the GAN.

Generation with Conditional WaveGAN

Once the model is trained, it can synthesize one or many signals by feeding G with n latent vectors z and the associated condition c . Something worth exploring is the standard deviation of the latent vector, σ_z . The default value used during the cWaveGAN model training is $\sigma_z = 1$. When this value is changed, there is a higher probability of getting values farther from the mean value (in this case, $\mu_z = 0$), which provides higher diversity in the generated samples.

To provide an idea of the generated signals, a batch of 1000 samples is generated with the condition $c = x_{desired} = 208$ mm and $\sigma_z = 1$. The generation process for one signal takes 140 ms, while generating 1000 signals takes 1140 ms (i.e., 1.14 ms per signal). To illustrate the generation quality, nine examples are randomly selected from the batch. These signals are compared with one signal from the TrIRDs for the same location $x_a = 208$ mm. This

comparison is shown in Fig. 5.5. The signals are cropped to 25 ms (5000 samples length) to facilitate comparing them. When cropping the signals, the first part of the signal is removed (similar to Stage 1).

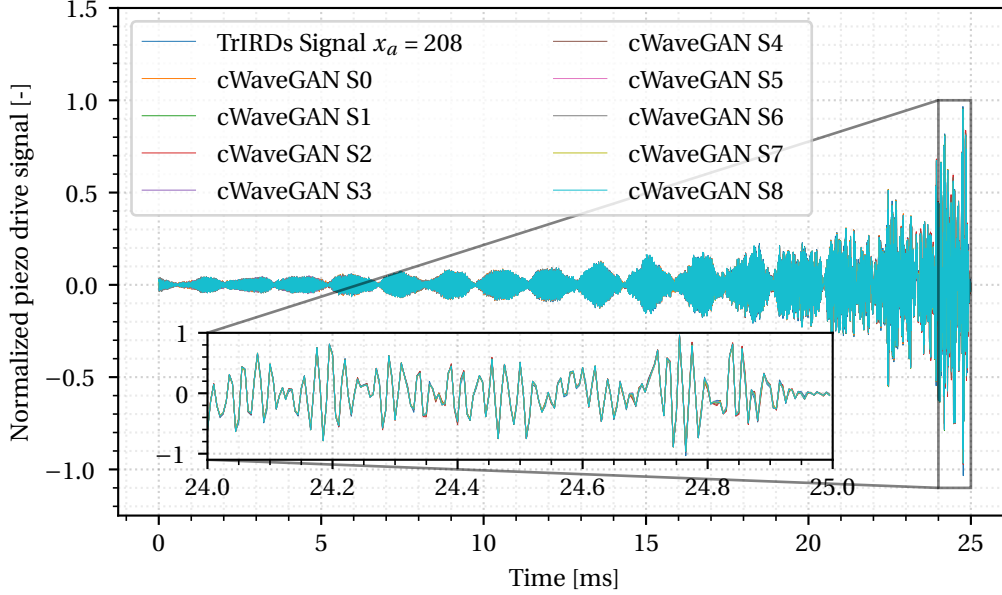


Figure 5.5: The reference TrIR at location $x_a = 208$ mm and and nine signals from the cWaveGAN model when $\sigma_z = 1$ and $c = 208$ mm.

In the time domain, the generated and the TrIRDs signals appear to be very similar or even the same. In this case, the graphical comparison is harder and the cross-correlation metric becomes even more relevant. This metric is evaluated between the reference signal (Ref TrIR for $x_a = 208$ mm) and each of the samples from cWaveGAN [S0, S1, ..., S8]. The results are displayed in Table 5.4. The scalar values are much closer to each other, compared to the samples from WaveGAN. This result indicates that there is much more similarity among the samples. Nonetheless, the results are not the same and some wanted diversity is still present.

The frequency-domain representation for the ten signals is presented in Fig. 5.6. In the frequency domain, the signals present the main peaks of magnitude in the same frequencies. The magnitude has slight variations among the different signals which is the expected effect of the diversity of the GAN. The results for the frequency-domain cross-correlation are presented in Table 5.5.

Like the time domain, the frequency-domain cross-correlation results confirm that the cWaveGAN signals are much closer than the WaveGAN examples. This degree of similarity is an expected effect of the condition on the generation. The results for the cross-correlation in the frequency domain have the same trend as in the time domain, so only one of the two evaluations is sufficient to study the similarity.

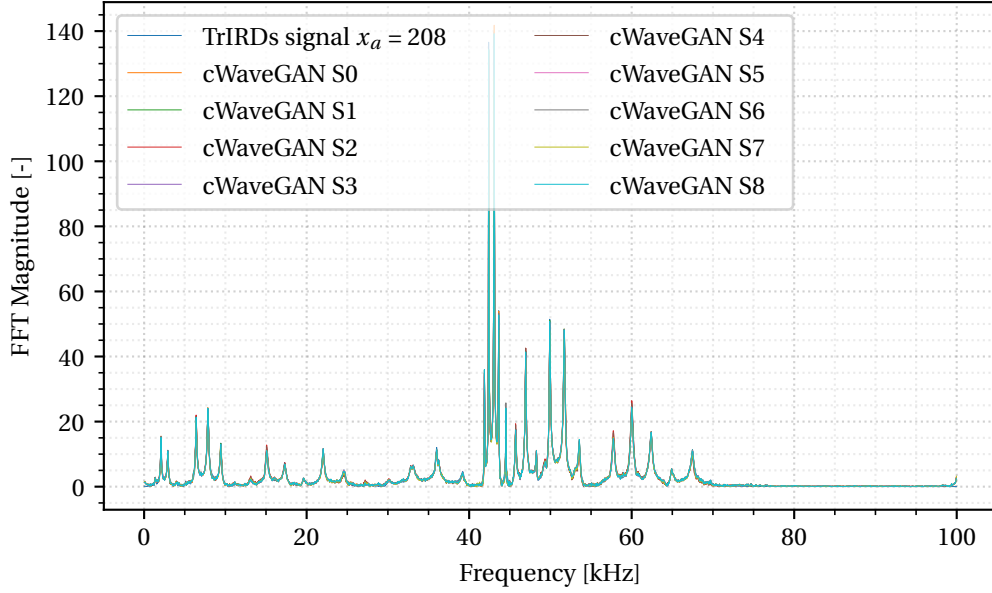


Figure 5.6: Magnitude of the FFT for the reference TrIR at location $x_a = 208$ mm and nine signals from the cWaveGAN model when $\sigma_z = 1$ and $c = 208$ mm.

Table 5.4: Time-domain cross-correlation between the reference signal (Ref TrIR) and the nine evaluated signals.

| Signal | cross-correlation |
|------------|-------------------|
| Ref TrIR | 58.4773 |
| WaveGAN S0 | 58.3998 |
| WaveGAN S1 | 58.0038 |
| WaveGAN S2 | 56.9311 |
| WaveGAN S3 | 56.1964 |
| WaveGAN S4 | 56.4638 |
| WaveGAN S5 | 57.3905 |
| WaveGAN S6 | 57.9665 |
| WaveGAN S7 | 54.5053 |
| WaveGAN S8 | 58.1150 |

Table 5.5: Frequency-domain cross-correlation between the reference signal (Ref TrIR) and the nine evaluated signals.

| FFT magnitude | cross-correlation |
|---------------|-----------------------|
| Ref TrIR | 14.6193×10^4 |
| WaveGAN S0 | 14.6213×10^4 |
| WaveGAN S1 | 14.5203×10^4 |
| WaveGAN S2 | 14.2622×10^4 |
| WaveGAN S3 | 14.0699×10^4 |
| WaveGAN S4 | 14.1375×10^4 |
| WaveGAN S5 | 14.3668×10^4 |
| WaveGAN S6 | 14.5142×10^4 |
| WaveGAN S7 | 13.6714×10^4 |
| WaveGAN S8 | 14.5491×10^4 |

Increasing the Diversity of the Generated Signals

The next step is to validate the effect of σ_z on the generated signals' diversity. Fig. 5.7 presents the reference signal from the TrIRs and nine signals from the cWaveGAN generated with $\sigma_z = 3$ and $c = 208$ mm. Fig. 5.8 presents the magnitude of the FFT for the same signals.

When the σ_z is increased there is more diversity in the generated signals compared to $\sigma_z = 1$.

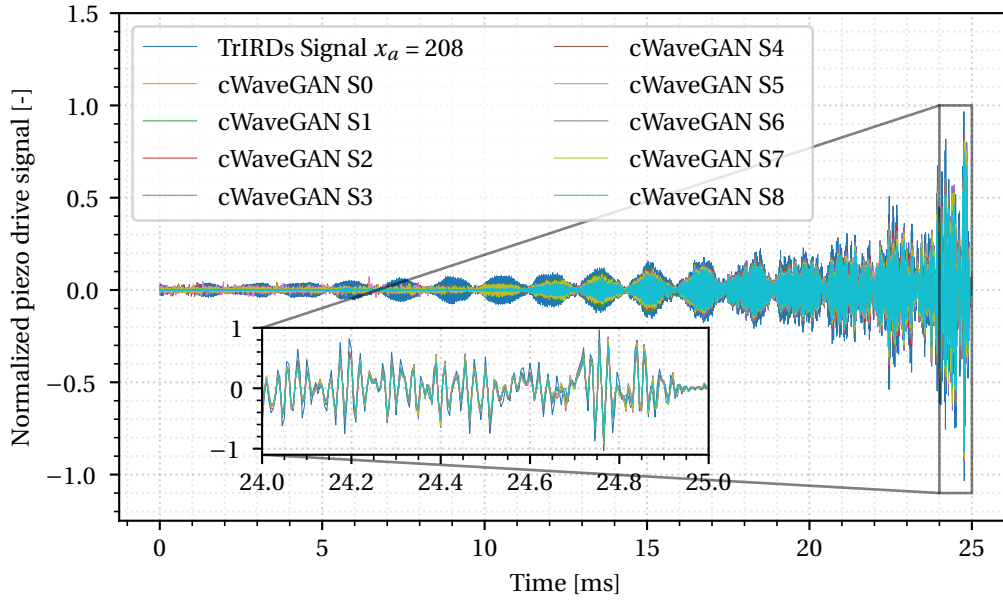


Figure 5.7: The reference TrIR at location $x_a = 208$ mm and nine signals from the cWaveGAN model when $\sigma_z = 3$ and $c = 208$ mm.

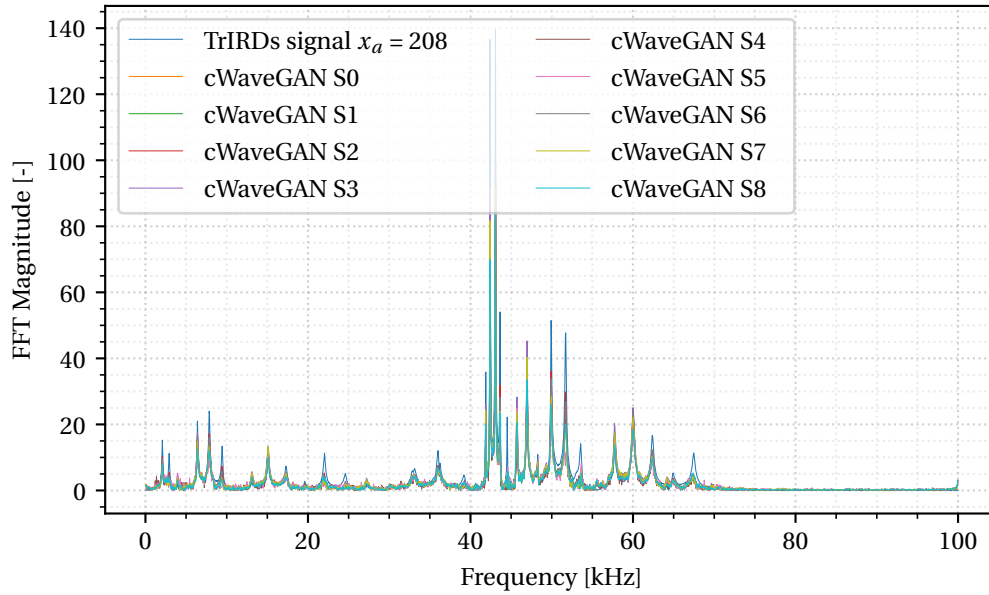


Figure 5.8: MMagnitude of the FFT for the reference TrIR at location $x_a = 208$ mm and nine signals from the cWaveGAN model when $\sigma_z = 3$ and $c = 208$ mm.

The additional diversity can already be observed in the time domain, where it is possible to see the variation of the waveform among the different signals. Nonetheless, in the frequency domain, the difference is more clear. By analyzing the magnitude of the FFT for the 10 signals, it is possible to observe that there is some energy going into frequencies outside of the main magnitude peaks (resonant modes). Also, it can be noted that the signals are noisier. This distortion is especially seen at the beginning of the signal. Finally, the cross-correlation is calculated in the frequency domain and the results are presented in Table 5.6.

Table 5.6: Frequency-domain cross-correlation for the reference signal (Ref TrIR) and nine signals generated by cWaveGAN when $c = 208$ mm and $\sigma_z = 3$.

| FFT Magnitude | $corr(FFTRef, FFTMag)$ |
|---------------|------------------------|
| Ref TrIR | 146.1933×10^3 |
| WaveGAN S0 | 96.0245×10^3 |
| WaveGAN S1 | 103.9179×10^3 |
| WaveGAN S2 | 108.8389×10^3 |
| WaveGAN S3 | 109.3159×10^3 |
| WaveGAN S4 | 92.1663×10^3 |
| WaveGAN S5 | 84.3247×10^3 |
| WaveGAN S6 | 108.0308×10^3 |
| WaveGAN S7 | 100.7355×10^3 |
| WaveGAN S8 | 92.9399×10^3 |

When σ_z is increased, the generated signals have less similarity and more diversity when compared to the signals from cWaveGAN where $\sigma_z = 1$. The effect of increasing the standard deviation is confirmed.

5.3.4 GAN Diversity Effect on the Localized Peak

The HiL experimental setup is used to validate the effect of the diversity from the GAN on the localized peak properties. A group of ten signals generated by the cWaveGAN model is studied. The signals are generated by feeding ten latent vectors with $\sigma_z = 1$ and $x_{desired} = 208$ mm as the condition to the cWaveGAN model. All the signals are cropped to a length of $T_s = 25$ ms.

The vibration scan is acquired for each signal, and the results are presented in Fig. 5.9, and the evaluated metrics are compiled in Table 5.7.

In the displacement profile, one can observe some variation among the samples. Most of the differences occur in the amplitude of the peak which was already expected after Chapter 4 experiments. During the different reinforcement learning experiments, it was observed that NNs tend to modify the $u + peak$ before any other peak property. However, the most exciting variation is the slight changes in the peak location. One can observe that the measured peak location, x_{peak} , varies in the order of 0 to +2 mm around the target location. The contrast ratio also has some variations, but this can be ignored for the moment.

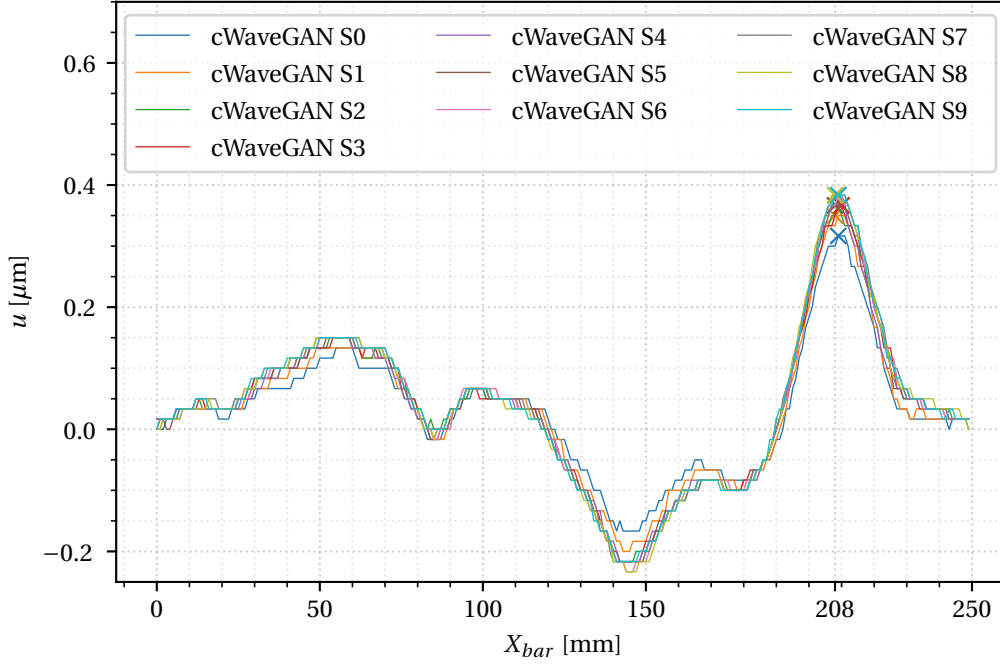


Figure 5.9: Displacement profile at the center-line of the aluminum bar after several vibration Scans for ten signals generated by the cWaveGAN, when $\sigma_z = 1$ and $c = 208$ mm.

Table 5.7: Localized peak evaluation metrics for ten signals generated by the cWaveGAN, when $\sigma_z = 1$ and $c = 208$ mm.

| Signal | x_{peak} [mm] | u_{peak} [μm] | t_{peak} [ms] | C_r |
|-------------|-----------------|------------------------------|-----------------|-------|
| cWaveGAN S0 | 209 | 0.3167 | 25 | 3.02 |
| cWaveGAN S1 | 208 | 0.3500 | 25 | 2.96 |
| cWaveGAN S2 | 210 | 0.3667 | 25 | 2.86 |
| cWaveGAN S3 | 210 | 0.3667 | 25 | 2.85 |
| cWaveGAN S4 | 208 | 0.3667 | 25 | 2.87 |
| cWaveGAN S5 | 208 | 0.3667 | 25 | 2.81 |
| cWaveGAN S6 | 209 | 0.3833 | 25 | 2.86 |
| cWaveGAN S7 | 209 | 0.3833 | 25 | 2.83 |
| cWaveGAN S8 | 208 | 0.3833 | 25 | 2.81 |
| cWaveGAN S9 | 209 | 0.3833 | 25 | 2.84 |

Repeatability of the Vibration Scan and Peak Localization

The repeatability of the acquisition process is evaluated to confirm that this variation is not coming from the vibration scan. To do this, the reference signal (TrIRDs for $x_a = 208$ mm) is used in 4 repetitions. The signal is used to create a localized peak, and the HiL experimental setup acquires the displacement profile for each repetition. The results are presented in Fig. 5.10 and the evaluated metrics are displayed in Table 5.8.

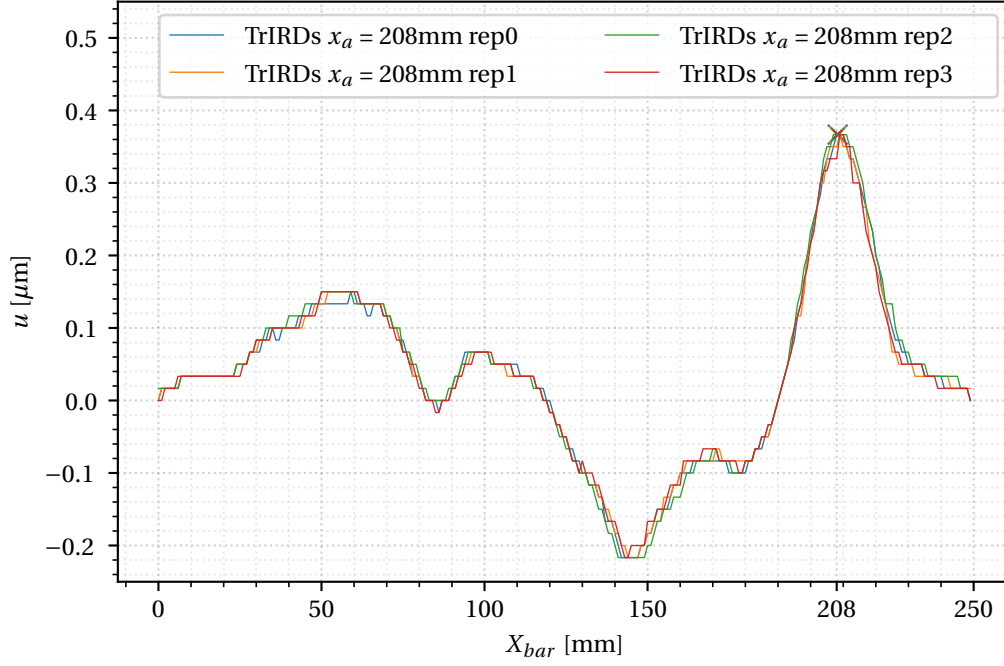


Figure 5.10: Displacement profile at the center-line of the aluminum bar after several vibration scans for the reference signal from the TrIRDs.

Table 5.8: Localized peak evaluation metrics after four vibration scan repetitions for the reference signal from the TrIRDs.

| Signal | x_{peak} [mm] | u_{peak} [μm] | t_{peak} [ms] | C_r |
|----------------------------|-----------------|------------------------------|-----------------|-------|
| TrIRDs $x_a = 208$ mm Rep1 | 208 | 0.3667 | 25 | 2.88 |
| TrIRDs $x_a = 208$ mm Rep2 | 208 | 0.3667 | 25 | 2.93 |
| TrIRDs $x_a = 208$ mm Rep3 | 208 | 0.3667 | 25 | 2.76 |
| TrIRDs $x_a = 208$ mm Rep4 | 208 | 0.3667 | 25 | 2.97 |

As expected, the vibration scan and peak localization are very repeatable. The localized peak appears in the same location with the same amplitude. The small variations in the contrast ratio are negligible. This experiment confirms that this variability in the location and amplitude comes from the diversity of the GAN.

5.3.5 Discussion: Diversity Perception Effect

The variation in the measured location of the peak is an exciting finding. This behavior motivates a novel pattern of stimulation for vibrotactile feedback. The traditional approach to time-reversal haptics is to repeat the localized peak at a given rate to create a vibration sensation. What if, instead of constantly repeating the peak in the exact same location, the

peaks are created at random locations around the target location?

As discussed in Chapter 2, the Pacinian channel is the main mechanoreceptor involved in the human perception of vibration. One remarkable property of this channel is that it is capable of spatial and temporal summation. This effect means that the contact area affects the perception threshold, i.e., a more significant contact area lowers the detection threshold or increases the perceived intensity. In this novel approach, the random location of the peak affects a larger area than a single-location peak. Thus, instead of increasing the contact area, the randomly localized peaks increase the area affected by the vibration.

The potential advantage of our novel approach is that, by affecting a larger area of the finger, such random stimulation could activate more mechanoreceptors, thus leading to lower vibration detection thresholds. As a result, with lower amplitude requirements, one could render lower peak amplitudes while obtaining the same perceptual effect on the user. This perceptual hypothesis will be validated in the next chapter (Chapter 6).

5.4 Conclusion

In this chapter, the GAN framework is explained. Then, a state-of-the-art GAN model for raw-audio generation was trained to run unconditional generation experiments. During these experiments, it is shown that GANs can capture the statistical distribution of the TrIRDs. The trained generator can synthesize signals that create localized peaks with similar properties to the reference peaks with the Time-reversal method.

Real-time generation was validated by measuring the time it takes to generate and reproduce a signal. It was shown that it is possible to synthesize signals to obtain a localized vibration within the human vibrotactile range. Several ideas to optimize the generation are proposed, including asynchronous generation. This idea is essential in the case of the conditional GAN, where the generation takes more time than in the unconditional model.

A strategy to transform the WaveGAN model into a conditional model is presented. This condition accelerates the training process, increases the quality of the generated signals, and allows to control the location where the peaks occur.

Experiments with the standard deviation of the latent vector (σ_z) show that increasing this parameter augments the diversity of the signals in the time and frequency domain. Nonetheless, if this parameter is too high, the generated signals contain too much noise and become distorted.

Several vibration scans were acquired for signals synthesized by the cWaveGAN with the default $\sigma_z = 1$. The synthesized signals present some diversity on the generated peaks. The main effect occurs in the amplitude of the peak. However, it was observed that the location of the peak varies in the range of 0 to +2 mm with respect to the target location. This random variation in the measured peak location (x_{peak}) inspired us to develop a novel stimulation pattern. Instead

of obtaining the localized vibration by repeating the same peak (i.e., driving the piezo with the same TrIR at every period), we propose to create peaks at randomly-distributed locations around the target. This novel stimulation pattern is expected to affect human perception of vibration since it can stimulate a larger finger area. As a result, it can potentially be perceived as a more alerting stimulus than a repeated localized peak. This effect will be validated in the next chapter (Chapter 6).

In the future, studying alternative GAN conditioning strategies, such as a continuous label, could be interesting. The continuous label could introduce more diversity into the generated signals and ensure an even higher quality for the generated signals. In addition, it is worth exploring the effect of interpolation in the conditions, an interesting approach that has already been studied in the image domain. The interpolations on the generated data permit the combination of two classes of data (e.g., two desired locations $x_{desired}$).

Lastly, it would be interesting to reduce the size of the network and generate a shorter signal. This modification would drastically reduce the generation time and overhead of the GAN model if it were to be implemented on an embedded device. Reducing the length of the Tr Signal makes sense because only 1 ms of the TrIR Signal is required to obtain a localized vibration at 250 Hz (The required T_s is explained in Chapter 6).

Publications related to this chapter :

- C. H. Mejia, J. Jayet, P. Germano, A. Thabuis, and Y. Perriard, "Linear Impact Generator for Automated Dataset Acquisition of Elastic Waves in Haptic Surfaces," in *2019 22nd International Conference on Electrical Machines and Systems (ICEMS)*, pp. 1–5, Aug. 2019.
- C. Hernandez-Mejia, X. Ren, A. Thabuis, J. Chavanne, P. Germano, and Y. Perriard, "Generative Adversarial Networks for Localized Vibrotactile Feedback in Haptic Surfaces," in *2021 24th International Conference on Electrical Machines and Systems (ICEMS)*, pp. 105–110, Oct. 2021.
- C. Hernandez-Mejia, X. Ren, A. Thabuis, J. Chavanne, P. Germano, and Y. Perriard, "Conditional GANs for Localized Vibrotactile Feedback in Haptic Surfaces," in *IEEE Transactions on Industry Applications (TIA)*[Under Review]
- C. Hernandez-Mejia, X. Ren, S. Papetti, H. Jarvelainen, T. Chavdarova, P. Germano, and Y. Perriard, "GANs for Localized Vibrotactile Feedback in Haptic Surfaces: Single Vs Randomly-distributed stimulation", in *IEEE Transactions on Haptics (ToH)*. [Pending]

6 Human Perception Experiments and Time-reversal Haptics Demonstrator

6.1 Introduction

The previous experiments with Generative Adversarial Networks inspired us to develop a novel stimulation pattern where the localized vibration is created with randomly-distributed (i.e., Uniform Distribution) peaks around the target location. The previous chapter hypothesized that this stimulus could be perceived as more alerting than a repeated localized peak (which is the traditional approach in time-reversal haptics). We consider a signal is more "alerting" if it catches the attention of the user easier than another signal.

This last chapter focuses on developing a time-reversal haptic demonstrator to validate the perceptual effect of the novel stimulation pattern. Moreover, this haptic demonstrator can serve as the base for developing a haptic-featured Digital Musical Instrument (DMI).

The research question for this chapter is *"What is the effect of randomly distributed peaks on the human perception of localized vibrations?"* The associated hypothesis is that *"Diverse signals in contrast to repeating the same time-reversed signal are perceived as more alerting and can reduce the detection threshold for a localized vibration"*.

In the context of tactile letter recognition, different studies have found that dynamically presented patterns and shapes are easier to perceive than static ones [132, 133, 134, 135, 136]. In general, the detection accuracy increments when the shape or pattern is presented dynamically in contrast to a static presentation. Furthermore, in the context of visual and auditory perception, it has been demonstrated that dynamic signals (e.g., a siren or a moving object) are given more attention than static objects or sounds [137, 138]. i.e., our brain focuses its attention on changing phenomena more than static ones. The fact that this effect exists on similar types of feedback and in alternative human perception channels supports the motivation to evaluate the effect of randomly-distributed peaks in the perception of localized vibrational feedback.

6.2 Methodology

To validate the hypothesis, a human perception experiment is designed and executed. The main goal is to investigate the perceived differences between a vibration created with a single peak (SINGLE stimulus) and a vibration created with randomly-distributed peaks around a target point (RANDOM stimulus). The experiment aims to evaluate which of the two stimuli is perceived as more alerting.

The perception study is an experimental survey research, where a group of voluntary participants (in the range of 20-60 years old) are asked to touch a vibrating surface with their fingers and answer questions about the perception of different vibration patterns. The main experiment is a paired comparison test where participants assess which type of vibration is more alerting. A pilot experiment is proposed to get an idea of the detection thresholds for both types of stimulus as a function of the displacement amplitudes (in the range of 1-10 μm).

The human detection threshold should be considered when designing the experimental setup for the perception experiments. As discussed in Chapter 2, the main mechanoreceptor involved in the perception of vibrotactile feedback is the Pacinian channel. The Pacinian corpuscles are mainly activated by higher frequencies (40 - 800 Hz [26]) and have shown that the sensitivity threshold has an up-side down U-shape, with the highest sensitivity around 250 Hz [30]. At this frequency, the receptor can detect vibratory amplitudes down to 0.01 μm [31].

According to these findings, one should be able to perceive the localized peaks that are created over the aluminum beam experimental setup, but they are not perceived. This can be explained by the differences in the stimulus that is used. In the perception studies by Bolanowsky *et.al.* [26, 139] and Verillo [30, 140], the stimulus is a continuous sinusoidal displacement. In contrast, in time-reversal haptics, the localized vibration is created by repeating the localized peak at a given rate (i.e., a train of pulses). The key aspect is that a localized peak has a limited duration, which is determined by the temporal resolution R_t (e.g., 21.2 μs). For this reason, the perception thresholds are different, in general, higher. Previous studies [57, 58] found that the perception threshold for localized peaks obtained with a TrIR varies between 4 and 7 μm .

For this reason, it is necessary to develop an optimized setup where the amplitude can be higher than the perception threshold. In addition, it is highly relevant to validate the SINGLE vs. RANDOM hypothesis in two dimensions. This means having a randomly-localized peak that not only varies in a single direction but can move around the target point. For this purpose, a time-reversal haptics demonstrator is developed. The system is designed to generate peaks of at least 1 to 10 μm that can be repeated at a rate of 100 to 500 Hz. The chosen material for the surface is glass, which enables future integration with screens for multimodal interaction. The development of the demonstrator is done in two steps:

First, a smaller prototype with a single piezo actuator is developed. This prototype is used to evaluate two bonding methods to attach the piezoelectric actuator to the plate and three fixing strategies to support the plate.

Second, a full-scale demonstrator with multiple actuators is developed. The surface is characterized using the engineering tradeoffs presented in Chapter 2. The resulting experimental setup allows obtaining peaks within the human perception threshold.

The demonstrator is used for the perception experiments. The TrIR signals between all the actuators and several locations in the center of the glass plate are acquired. These signals allow the creation of localized peaks in pre-determined locations. The signal in the center of the glass plate is used to create the single-point localized vibration ("SINGLE"). The randomly-localized vibrations ("RANDOM") are created using a group of five signals around the center of the plate. This pattern provides a simplified reproduction of the diversity brought by the GANs (Chapter 5). This stimulus is created by randomly (Uniform distribution) moving through the different locations, but only one location is actuated at a given time. The signals that are included in this group can vary, and this variation is a parameter of the perception experiment.

6.2.1 Ethics Committee Approval (HREC)

The executed perceptual experiments respect all the ethical regulations. Before starting the experimentation, the motivation and experimental protocol were submitted to the Human Research Ethics Committee (HREC) at EPFL (Proposal No: 016-2022 approved on 25.04.2022).

6.3 Development of the Tr Haptics Demonstrator

The optimized time-reversal haptics demonstrator is developed following the engineering tradeoffs presented in Chapter 2.

The first step is to select the glass material and the thickness of the plate (h). In Chapter 3, the model from Kim *et.al.* [95] was used to calculate the effective bending moment as a function of the piezo to plate thickness ratio (ρ_z) for different materials. In Fig. 3.3, it is clear that the AF32[®] glass presents the highest effective bending moment of all the analyzed glass structures. For this reason, this material is chosen for the haptic demonstrators. AF32[®] glass has a mass density, $\rho = 2430 \text{ kg m}^{-3}$, a Young's modulus, $E = 74.8 \text{ GPa}$, and a Poisson's Ratio $\nu = 0.238$.

Following that, the next parameter is h . Power consumption is one of the most important aspects to consider for the time-reversal method. When analyzing Eq. 2.20 (In Section 2.7.2), it is evident that h is the most influential parameter since it has a power of 3. Therefore it is crucial to keep this parameter as low as possible to reduce energy consumption. An $h = 400 \text{ }\mu\text{m}$ is defined for the glass structure. This thickness is enough to resist finger pressing force of up to 4 N and keep the energy requirements low.

One way to increase the localized peak amplitude is to optimize the interaction between the piezoelectric actuator and the surface. With h defined, the next step is to find the optimal piezo thickness. Based on the effective bending moment vs. thickness ratio analysis that was made in Section 3.2.1 (Fig. 3.3), it can be found that the optimal ρ_z for the AF32[®] glass

structures is 0.9. From this value, it can be inferred that the optimal thickness for the piezo is $t_z = 0.45h$. In [141], Kim's model was validated experimentally, and it was found that a piezo thickness that is slightly larger than the optimal value is preferred.

For this reason, the optimal piezo thickness range is defined between $[0.45h$ and $0.55h]$. Based on this, an off-the-shelf piezoelectric actuator is chosen. The Murata 7BB-12-9, which was presented in Fig. 2.18, has a diameter of $D_{piezo} = 9$ mm and a total thickness of 0.22 mm, which respects the optimal thickness range.

Before continuing with the development of the haptics demonstrator, let us revisit the aluminum beam experimental setup that was used in the previous chapters.

6.3.1 Aluminum Beam Experimental Setup

After evaluating these initial tradeoffs, it is clear that the aluminum beam experimental setup is not optimized for peak generation. The thickness of the plate is too high, which reduces the peak amplitude or demands a higher amount of energy to reach a given amplitude. In this setup, the maximum peak amplitude achieved for a driving voltage $V_{piezo} = 50$ V_{pp} is 0.7 μ m. One alternative to increasing the peak amplitude is changing the diving signal strategy. "One-bit quantization" is a signal processing strategy that can simplify the driving electronics while increasing the peak amplitude.

One-Bit Piezo Driving Strategy

The impulse response is, by definition, an analog signal. Up to this point, an arbitrary waveform generator coupled to a broadband amplifier has generated the piezo driving signal. This configuration is acceptable for the experimental study that was carried out. Nonetheless, considering the potential applicability of this vibrotactile haptic feedback strategy in portable devices, it is essential to consider a compact yet robust driving strategy. Furthermore, if a higher number of piezo actuators are required, it is even more relevant to simplify the driving signal. A piezoelectric transducer is mainly as a capacitive load, and different models have been developed to represent its electrical characteristics (e.g., [56]). Driving capacitive loads at high frequencies represents a big challenge, mainly because of the high transient current requirements and the need for a bipolar source to reproduce the original signal.

The first simplification comes from the fact that the time-reversal process is insensitive to DC components. Thus, the TrIR amplitude can be shifted by a constant offset to ensure that all the signal is positive, eliminating the need for bipolar amplification. The second simplification involves the one-bit quantization of the driving signal (i.e., only using the sign of the signal). In [142], Derode *et al.* demonstrate the feasibility of one-bit time-reversal of elastic waves. The most surprising discovery is that when the Tr signals are digitized by one bit, the focusing quality is preserved while the peak amplitude is amplified. Moreover, both temporal and spatial resolutions remain unchanged.

This strategy was first exploited in [68] to simplify the piezo driving strategy while rendering localized haptic feedback. However, it is demonstrated that the increase in amplitude comes at the expense of added noise emission. This noise is explained by the fact that binary drive signals excite all frequencies, therefore all the plate modes.

Several strategies to reduce noise emissions have been proposed. Some of them involve oversampling and filtering [57], and others consider alternative quantization strategies (e.g., high-frequency class-D amplifiers coupled with low-pass filters) [58]. Nonetheless, the trade-off between peak amplitude and audible noise is acceptable for this thesis. In the future, alternative driving and signal-preprocessing strategies can be implemented.

The 1-bit quantization process is very simple and is summarized in one equation: $V_{piezo}(t) = \frac{1}{2} V_{drive}(1 + \text{sgn}(IR(T_s - t)))$, where, sgn is the sign function (i.e., returns -1 when the signal is negative and 1 when the signal is positive), $IR(T_s - t)$ is the time-reversed impulse response signal of length T_s , and V_{drive} is the desired peak-to-peak amplitude of the signal.

The HiL experimental setup is used to compare the displacement profile on the aluminum beam for a TrIR signal and the 1-bit quantized version ("1-bit TrIR"). After the scan it can be found that u_{peak} for the TrIR signal is $0.41 \mu\text{m}$, while u_{peak} for the 1-bit TrIR signal is $2.1 \mu\text{m}$. The same driving voltage of $30 V_{pp}$ is used in both acquisitions. From this point on, the 1-bit TrIR signal is the piezo-driving signal of choice and will be used for both haptic demonstrators.

Development of the Driving Electronics

Apart from the amplitude increase, the one-bit quantization strategy's most exciting benefit is the driving electronics' simplification. Instead of using a bulk device to generate and amplify a broadband analog signal, the signal can be created using a digital output from a microcontroller or an FPGA. Then, a half-bridge driver can be used to amplify the signal to the desired driving voltage, V_{drive} . In this case, an FPGA with an 8-channel digital output module (NI9045 Compact Rio + Ni9041 High-speed DI/O Module) is used to reproduce the 1-bit TrIR signals.

A half-bridge driver with eight channels is designed and manufactured. The Printed Circuit Board (PCB) gets the digital signal (0 V to 5 V) as an input and outputs the same binary signal amplified to V_{drive} . For each channel, a high-voltage half-bridge gate driver (Texas Instruments LM5104) is used to control two N-channel MOSFET transistors (DiodesZetex ZXMN10A08). The half-bridge driver is designed for a V_{drive} from 12 V to 60 V.

A load resistor, R_L , is connected in series with the piezo actuator to create an RC filter. R_L is very important to dampen the voltage spikes due to the piezo's capacitive nature. The piezo driver was developed in two stages. First, a single-channel driver was developed and tested. Then this design was replicated to create the eight-piezo driver. The complete design and development process is presented in Appendix B.

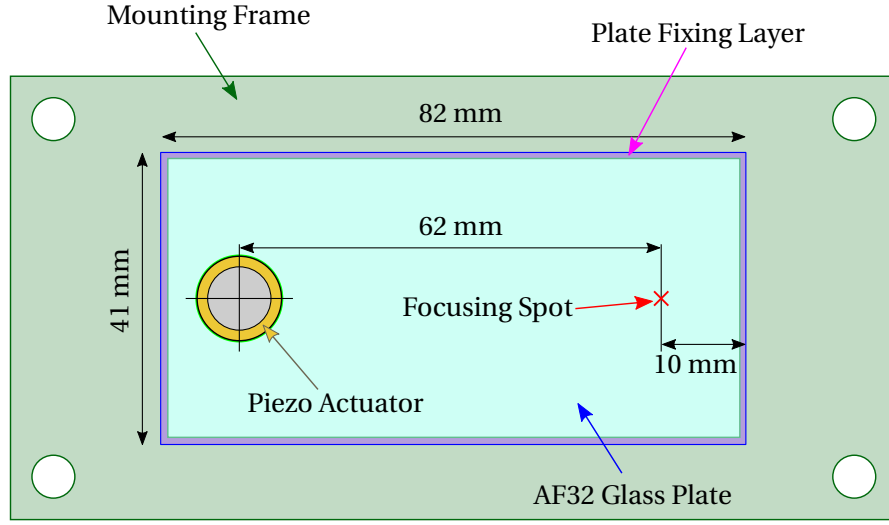


Figure 6.2: Single-piezo glass plate experimental setup.

ensures a homogeneous bonding layer across multiple piezo or devices (i.e., the tape can be pre-cut and easily applied on the surface).

The effect of both bonding strategies on the peak amplitude (u_{peak}) is compared experimentally. For this purpose, two twin prototypes are manufactured, as shown in Fig. 6.2. In both cases, the glass plate is mounted on a PMMA frame using a thick (1 mm) double-sided tape (3M VHB Tape). The Impulse Response (IR) between the piezo actuator and the focusing spot is acquired for both prototypes (same experimental setup and methodology from Section 4.2.1). The IR signal is time-reversed, cropped to $T_s = 1$ ms, and one-bit quantized. The resulting signals are given to the HiL experimental setup to carry out a vibration scan on the focusing spot.

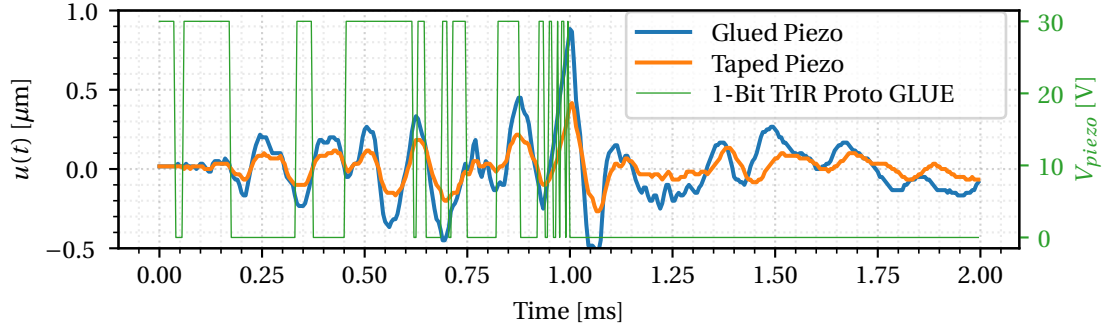
To validate the sensitivity of the IR acquisition to the bonding layer properties, the IR from both prototypes is used to create a localized peak on each device. Four vibration scans are acquired (In all scans $V_{drive} = 30$ V). Fig. 6.3, presents the displacement $u(t)$ measured at the focusing spot for both prototypes (Fig. 6.3a presents $u(t)$ when the 1-bit TrIR from the "Glued piezo" is used and Fig. 6.3b when the 1-bit TrIR from the "Taped piezo" is used).

The displacement values (u_{peak}) for each prototype are summarized in Table 6.1.

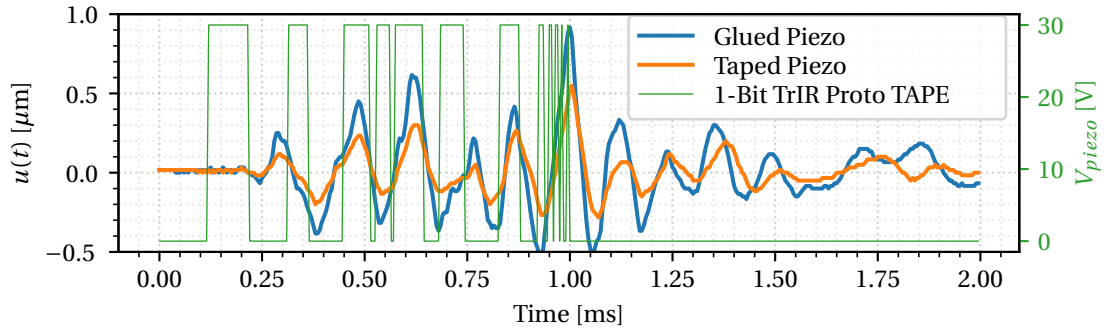
Table 6.1: (u_{peak}) for the two different piezo/plate bonding strategies.

| Prototype | u_{peak} [μm] for: 1-Bit TrIR Proto GLUE | u_{peak} [μm] for: 1-Bit TrIR Proto TAPE |
|-------------|--|--|
| Glued Piezo | 0.87 | 0.90 |
| Taped Piezo | 0.40 | 0.53 |

The peak amplitude (u_{peak}) for the "Glued piezo" is nearly two times higher than u_{peak} for



(a) $u(t)$ for both prototypes when the 1-Bit TrIR signal from the GLUE prototype is fed to the piezo.



(b) $u(t)$ for both prototypes when the 1-Bit TrIR signal from the TAPE prototype is fed to the piezo.

Figure 6.3: Displacement $u(t)$ measured at two different prototypes with different piezo/plate bonding strategies.

the "Taped piezo". Thus, the thickness of the tape is still too big to neglect the effects that the bonding layer elastic properties have on the effective bending moment. For this reason, the TAPE bonding strategy is discarded. An interesting observation is that the displacement measured on the prototype GLUE is slightly larger when injecting the IR acquired on prototype TAPE than the one acquired on itself. This result indicates that the bonding layer does not affect the time-reversal process and that the learning phase (IR acquisition) could be performed only once for similar devices.

Boundary Conditions Experiment

Boundary conditions are hard to model or reproduce in simulation. Furthermore, the fixing of the plate has a significant influence on the maximum amplitude of the localized peak (u_{peak}) and the plate's attenuation constant (τ). Three different fixing strategies are studied, where the plate fixing has a width of 1 mm on all edges.

The first condition (GLUE) is to attach the glass plate to the frame using instant glue (ULTRA Glue from the brand Pattex). This condition is a rigid constraint and permits reflections in the

border of the glass plate. The second condition (VHB) is to fix the glass plate using thick (1 mm) double-sided tape (VHB from 3M). This condition dampens the reflections in the border and allows for certain movement freedom because of the elasticity of the tape (silicon / PDMS). The third condition (MECH) is a mechanical fixing, where the glass plate is constrained using two plates of PMMA (top and bottom) and locked with screws. All the screws are tightened at a uniform torque (0.02 Nm). The single-piezo glass plate prototypes for the three fixing conditions are displayed in Fig. 6.4.

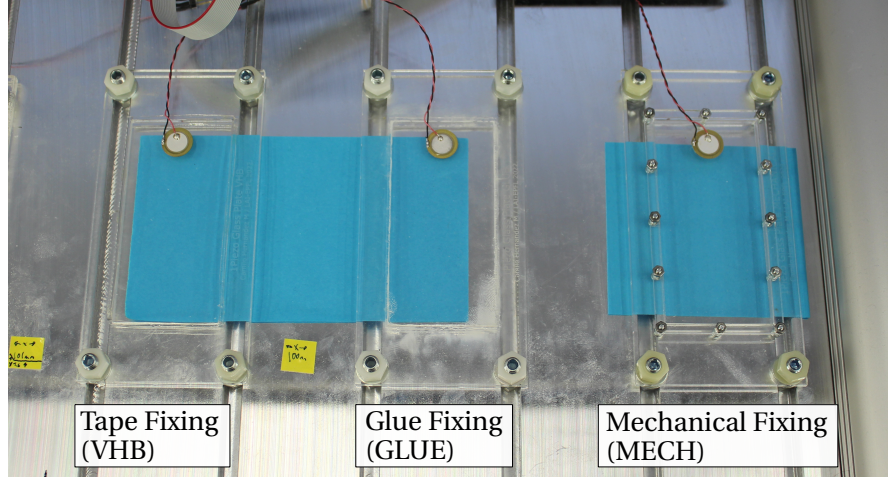


Figure 6.4: Single-piezo glass plate prototypes to evaluate the effect of the fixing conditions (Boundary conditions).

For this experiment, the focusing spot is located in the center of the plate (i.e., $x_a = 41$ mm and $y_a = 20.5$ mm, with respect to the border of the glass plate).

Like in the piezo/plate bonding experiments, the IR is acquired for the three plates and is used to create a localized peak in the focusing spot. In this case, the crossed effect of the IR from the different prototypes is not evaluated because the plate fixing boundary condition changes the natural modes of the system, thus the IR. Consequently, an impulse response for a plate fixed with glue would not reconstruct a peak on a plate fixed with double-sided tape.

To understand the relation between u_{peak} and the driving voltage for the different fixing conditions, u_{peak} is measured for different V_{drive} values (i.e., V_{drive} is increased from 30 V to 60 V in steps of 5 V). The results are presented in Fig. 6.5. The IR and displacement measurements were acquired on the same day at constant temperature conditions. The measured u_{peak} values for the extreme V_{drive} values are presented in Table 6.2.

The MECH fixing provides the most significant displacements, then the GLUE fixing, followed by the flexible VHB fixing. These results indicate that the higher the stiffness in the fixing, the higher u_{peak} . A rigid fixing enables more reflections on the borders and reverberation in the plate, which results in a better reconstruction of the localized peak. An interesting finding is the linear relation between V_{drive} and u_{peak} , which is beneficial for the perception

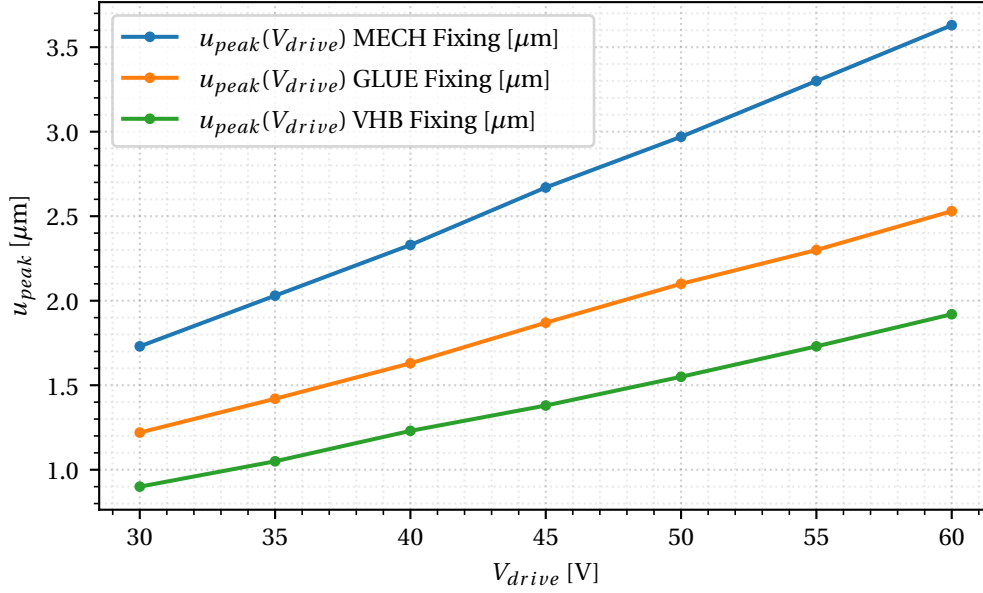


Figure 6.5: Displacement on the center of a single-piezo glass plate for different driving voltages. Three fixing strategies are compared. For each condition, the 1-Bit quantized TrIR of each prototype is used to create the peak.

experiments, where different peak amplitudes will be studied.

The increase in amplitude comes with an associated tradeoff. Measuring the attenuation constant (τ) for the three fixing conditions is important. In time reversal, τ plays a vital role because it limits the maximum repetition frequency (f_r), the duration of the signal (T_s), and the quality of the peak (C_r). This constant is difficult to estimate by analytical means, mainly due to the many mechanisms associated with it [144].

To estimate τ , a decreasing exponential, $e^{-\frac{t}{\tau}}$, is fitted to each IR. Fig. 6.6 presents the estimation τ for the glass plate with tape fixing. Since the signal is normalized, a reference line is drawn at $e^{-V_{peak}}$, where $V_{peak} = 1$ [-] is the peak amplitude of the signal. The intercept of this line with the decaying exponential marks the value of τ . For the VHB fixing, $\tau = 0.75$ ms.

Following the same procedure, the attenuation constant is estimated for the three plates. The results are presented in Table 6.2.

Table 6.2: $u_{peak}(V_{drive})$ and τ for the different plate fixing conditions.

| Fixing Strategy | $u_{peak}(30V)$ [μm] | $u_{peak}(60V)$ [μm] | τ [ms] |
|-------------------|-----------------------------------|-----------------------------------|-------------|
| Plate fixing TAPE | 0.90 | 1.92 | 0.75 |
| Plate fixing GLUE | 1.22 | 2.53 | 1.50 |
| Plate fixing MECH | 1.73 | 3.63 | 1.25 |

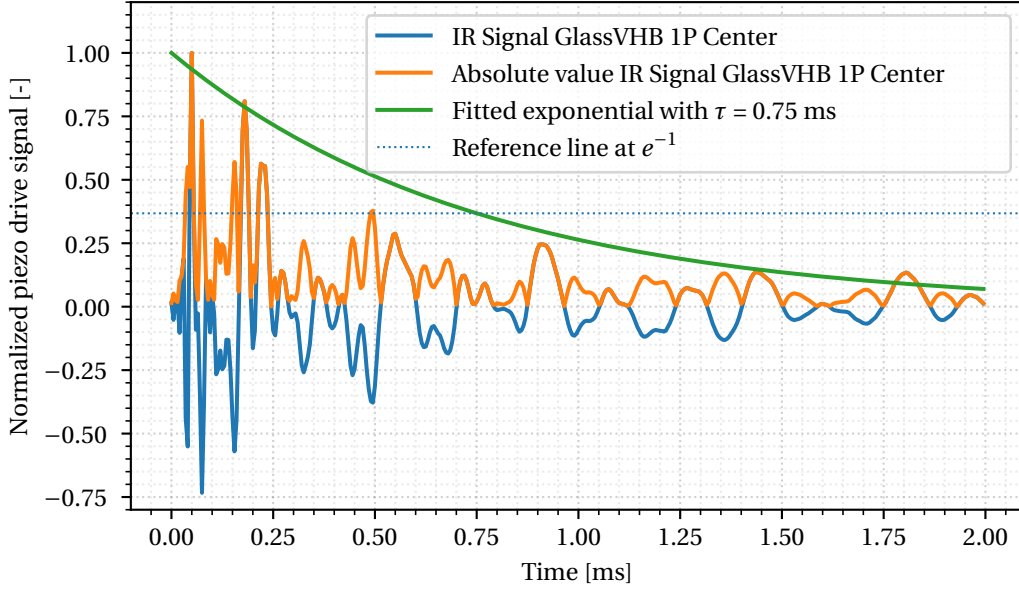


Figure 6.6: attenuation constant estimation for the single-piezo glass plate fixed with double-sided tape (3M VHB). A decreasing exponential with a time constant τ is fitted to the IR between the piezo and the focusing spot.

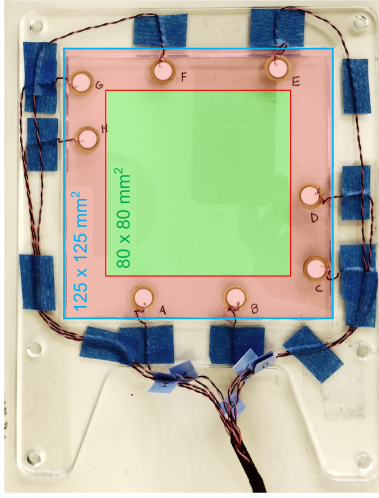
As expected, the flexible tape fixing provides the shortest attenuation constant ($\tau = 0.75$ ms). An unexpected behavior occurs for the mechanical fixing, which allows for the biggest u_{peak} i.e., u_{peak} MECH is 42 % bigger than u_{peak} GLUE), while having a smaller τ compared to the GLUE fixing (i.e., τ_{MECH} is 20 % smaller than τ_{GLUE}). In the future, it is worth exploring the effect of the tightening torque on u_{peak} and τ . It appears that the mechanical fixing is adding some pre-load into the system that contributes to increasing u_{peak} . An interesting idea for future works is to combine the mechanical and tape fixing strategies.

For this thesis, the tape fixing strategy is preferred. Regardless of the lowest u_{peak} value of the three conditions, it provides the shortest attenuation constant. This characteristic allows repeating the peaks at a higher rate. Moreover, it allows the energy from a peak to dissipate before the next peak, which improves the C_r (i.e., better localized peak quality).

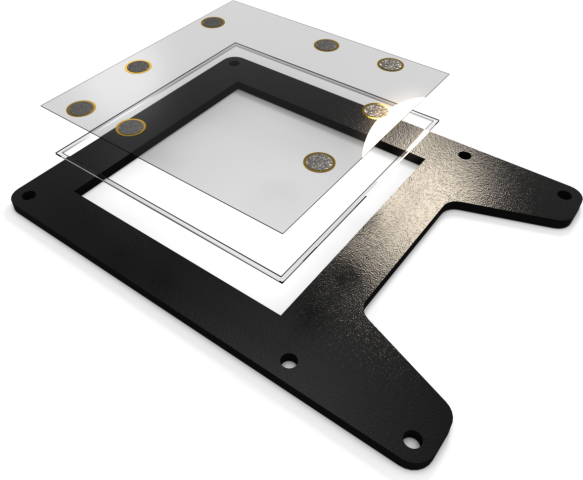
6.3.3 Eight-Piezo Glass Plate

Building on the knowledge from the single-piezo time-reversal demonstrator and the time-reversal engineering tradeoffs (Chapter 2), the next step is to develop and characterize the bigger demonstrator with multiple actuators. The time-reversal haptic demonstrator is presented in Fig. 6.7.

An AF32[®] Borosilicate glass plate with dimensions $125 \times 125 \times 0.4$ mm³ ($l \times w \times h$) is selected.



(a) Haptic demonstrator prototype.



(b) Haptic demonstrator plate fixing.

Figure 6.7: Eight-piezo glass plate haptic demonstrator.

An active area of $80 \times 80 \text{ mm}^2$ is defined in the middle of the glass plate (green area Fig. 6.7a). This space is the active area where the localized peaks are created (i.e., this is the interactive area). The number of piezo actuators is arbitrarily set to $Q = 8$. The actuators are randomly distributed in the 22.5 mm strip (red area Fig. 6.7a) avoiding symmetries. Each piezo is labeled with a capital letter, from Piezo A to Piezo H. The glass plate is fixed on top of a PMMA frame using thick (1 mm) double-sided tape (3M VHB) as illustrated in Fig. 6.7b.

With the defined mechanical properties, it is possible to calculate the time constant for the plate (Eq. 2.12), for this plate $T_c = 11.84 \text{ ms}$. This parameter is key because together with τ , they guide the selection of the diving signal duration, T_s .

To calculate the attenuation constant, the eight-piezo glass plate is mounted on the CNC, and the IR between each actuator and the center-point of the glass plate ($x_a = l/2 = 62.5 \text{ mm}$ and $y_a = w/2 = 62.5 \text{ mm}$) is acquired using the HiL experimental setup. Since the system remains constant, all signals contain the same attenuation constant. Then, τ is calculated by fitting a decaying exponential with a time constant τ to the Root-Mean-Square of the eight driving signals. This estimation is represented in Fig. 6.8. In this case, $\tau = 1 \text{ ms}$.

As discussed in Section 2.7.2, to maximize the contrast ratio, the following conditions should be met: $\tau \geq T_s$ and the maximum repetition frequency should respect $f_r \leq (1/\tau)$. Then, the driving signal duration is fixed at $T_s = 1 \text{ ms}$, and the maximum peak repetition frequency should not exceed 1 kHz (well above the detection threshold). Note that a longer T_s can increase u_{peak} and C_r but makes the system more sensitive to temperature variations and limits the maximum repetition frequency.

At this point, all the parameters for the eight-piezo haptic demonstrator have been defined.

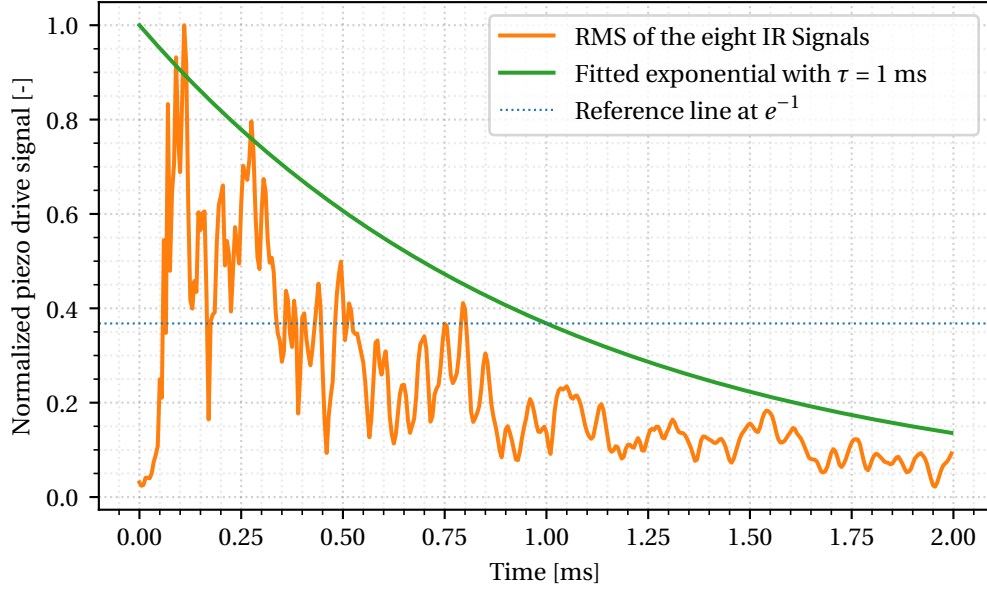


Figure 6.8: Time constant estimation for the eight-piezo glass plate fixed with double-sided tape (3M VHB). A decreasing exponential with a time constant τ is fitted to the RMS of the eight IR signals between each piezo and the center point.

The focalization capabilities of the demonstrator can be evaluated by creating a localized peak in the center-point of the glass plate. The IR signals corresponding to each piezo are cropped to $T_s = 1$ ms, time-reversed, and quantized in one bit.

HiL Experimental Setup Modifications

The HiL experimental setup is used to acquire the vibration scan. The signal generator and broadband amplifier are replaced with an FPGA connected to the eight-channel piezo driver. The 1-bit TrIR signals are stored on the FPGA, and this device is programmed to synchronously output the digital signals when a trigger signal from the HiL host computer is received. A two-channel programmable DC power supply (TTi CPX400 DP) is added to the experimental setup. This device provides the power for the eight-channel piezo driver PCB, and the piezo driving voltage (V_{drive}), which can be controlled from the host computer. Apart from these modifications, the vibration scan procedure remains the same. The experimental setup is presented in Fig. 6.9.

Localized Peak Evaluation

Two vibration scans are acquired. First, the entire active area is scanned while a localized peak in the center-point is created.

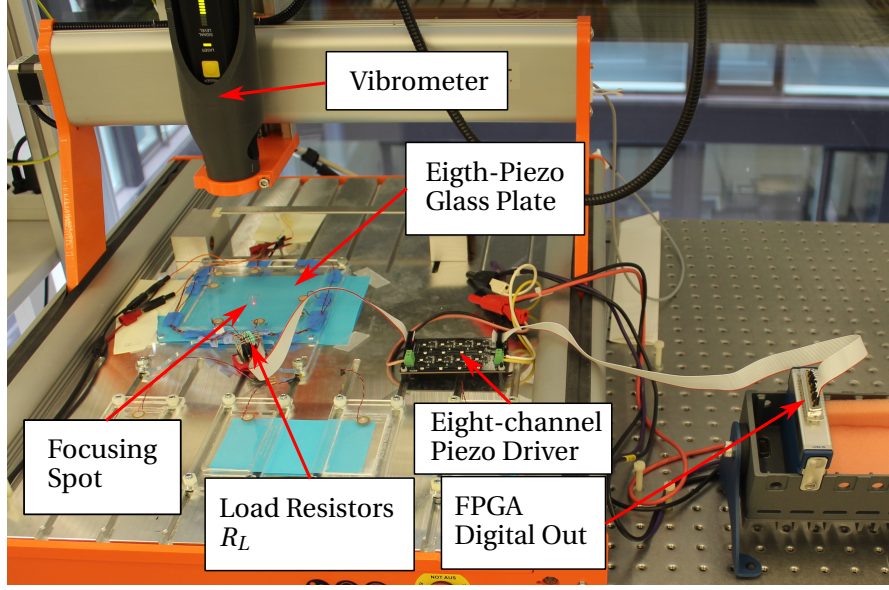


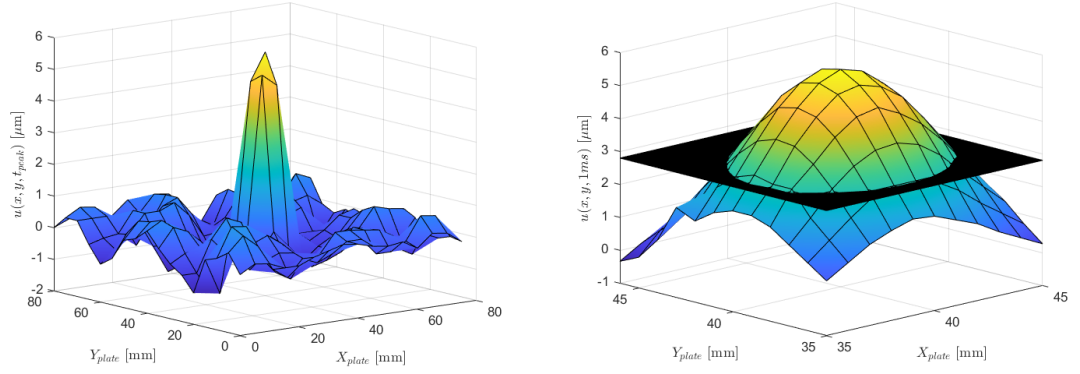
Figure 6.9: HiL experimental setup with the eight-piezo glass plate.

For this acquisition: $x_{start} = 0$ mm, $x_{end} = 80$ mm, $y_{start} = 0$ mm, $y_{end} = 80$ mm, and the scan spatial resolution is set to $R_{scan} = 5$ mm, which leads to a number of points in the X direction, $N = 17$ and a number of points in the Y direction, $M = 17$. The relative zero is located at the lower-left corner of the active area. The second scan provides a finer spatial resolution ($R_{scan} = 1$ mm) on an 11×11 mm² area at the center of the glass plate. This scan permits the evaluation of the spatial resolution (R_s) of the localized peak. (During both scans $V_{drive} = 30$ V).

The displacement of the glass surface at the instant of focalization ($t_{peak} = 0.995$ ms) is displayed in Fig. 6.10a. The localized peak appears at the center of the glass plate, and the peak amplitude is $u_{peak} = 5.6$ μ m. The measured contrast ratio, C_{rmeas} , is calculated by dividing u_{peak} by the RMS displacement at all other locations. For this prototype, the measured contrast ratio is $C_{rmeas} = 12.40$. The higher resolution scan for the center-point is presented in Fig. 6.10b, a reference plane is plotted at $u = u_{peak}/2$ to help the visualization of the spatial resolution of the peak. The measured spatial resolution is $R_s = 9$ mm, which matches the diameter of the piezo actuators (D_{piezo}).

Lastly, the temporal resolution (R_t) can be measured by studying the displacement of the center-point of the glass plate over time, which is presented in Fig. 6.11, by finding the instants when the displacement is equal to $u_{peak}/2$. From the figure, it is found that $R_t = 21.2$ μ s for the eight-piezo glass plate.

The theoretical contrast ratio, C_{rth} , that can be reached is calculated using Eq. 2.11 and the theoretical spatial and temporal resolutions are calculated using Eq. 2.16, Eq. 2.17, and Eq. 2.18. The theoretical and measured metrics are summarized in Table 6.3.



(a) $u(x, y, t_{peak})$ for the active area. The peak amplitude is $u_{peak} = 5.6 \mu\text{m}$. (b) High-resolution scan around the focus location, a plane at $u_{peak}/2$ highlights the spatial resolution.

Figure 6.10: Displacement ($u(x, y, t_{peak})$) at the center-point of the eight-piezo glass plate.

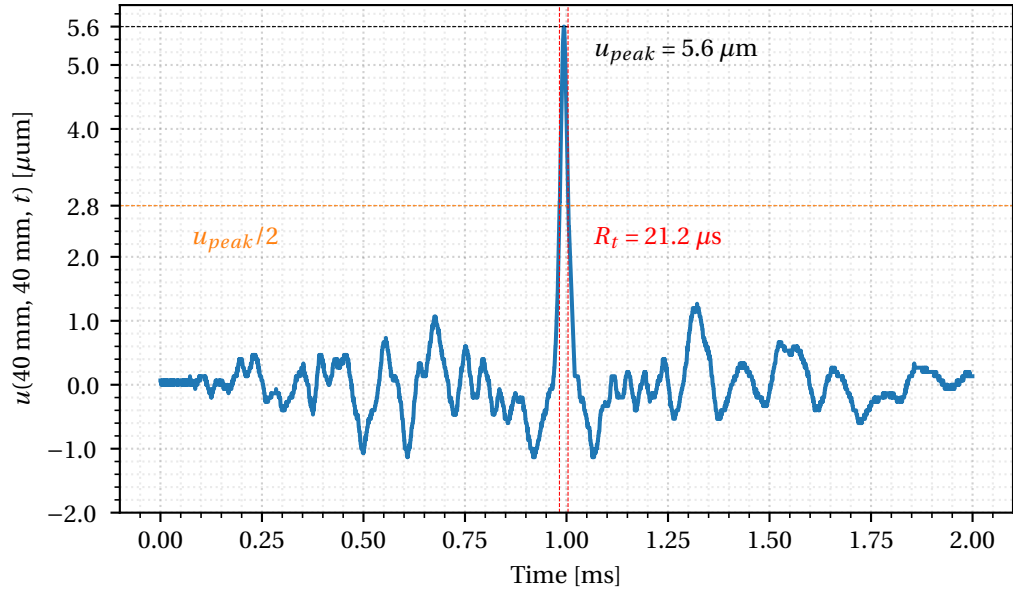


Figure 6.11: Temporal resolution (R_t) estimation for the eight-piezo glass plate. Displacement of the center-point over time ($u(40 \text{ mm}, 40 \text{ mm}, t)$).

Table 6.3: Measured and theoretical metrics for the eight-piezo glass plate.

| | C_r | R_s [mm] | R_t [μs] |
|-------------|-------|------------|-------------------------|
| Measured | 12.40 | 9 | 21.12 |
| Theoretical | 16.54 | 3.98 | 7.63 |

The difference between C_{rth} and C_{rmeas} is partially explained by the incertitude of the signal bandwidth (B in Eq. 2.11). The bandwidth for the driving signal (V_{piezo}) is not the same as

in the original TrIR because the original signal has gone through the one-bit quantization, and some noise is introduced after the amplification. To solve this, one can add a band-pass filter (i.e., add an inductance, L , in parallel with the piezo, C_{piezo} , to create an RLC filter) at the output of the half-bridge amplifier. This filter helps to ensure the desired bandwidth and get a more realistic calculation of C_{rth} . An additional benefit is that the filter can be designed to reduce audible noise by attenuating the frequencies in the audible range (below 20 kHz).

The diameter of the transducer (D_{piezo}) explains the difference between the theoretical and measured resolutions. The measured R_s matches the diameter of the piezo disc (D_{piezo}), which limits the smallest wavelength that can be created on the surface of the glass plate. Thus, D_{piezo} sets the lowest spatial resolution and temporal resolution (as mentioned in Chapter 2, these two parameters are inherently linked). To reach minimal theoretical values, it is necessary to reduce the dimension of the piezo actuator to ideally $D_{piezo} \leq R_s$ [57, p. 86].

Peak Amplitude Vs. Driving Voltage Vs. Number of piezo

Two parameters can control the amplitude of the localized peak. The first method is changing the piezo's driving voltage (V_{drive}). The second method is to decide the number of active piezo actuators, Q_a (i.e., which actuators are driven by the 1-bit TrIR Signal). Combining these two alternatives makes it possible to achieve a broader range of variation, given that the voltage range is limited [15 V to 60 V].

u_{peak} is measured for the different scenarios, where V_{drive} varies from 5 V to 60 V in steps of 5 V, while Q_a is augmented from 1 to 8 in steps of 1 piezo. The results are displayed in Fig. 6.12.

The relation between u_{peak} and V_{drive} is linear. Furthermore, each curve has a constant increment when Q_a is increased. Based on this trend, the line equation ($u_{peak} = m \times V_{drive} + c$, where m is the slope and c the intercept with the u_{peak} axis) is fit to each condition of Q_a . When analyzing the variation of m and c across the different values of Q_a , it can be observed that they vary linearly with respect to Q_a . Therefore, u_{peak} can be modeled in terms of V_{drive} and Q_a according to the following equation:

$$u_{peak}(V_{drive}, Q_a) = m(Q_a) \times V_{drive} + c(Q_a), \quad (6.1)$$

where, $m(Q_a)$ and $c(Q_a)$ are the line equations that represent the linear relation between the slopes and intercepts for the different conditions of Q_a (i.e., interpolation between the minimum number of active piezo, Q_{min} , and the maximum number of active piezo Q_{max}), which are defined as follows:

$$\begin{aligned} m(Q_a) &= m_m \times Q_a + c_m, \\ c(Q_a) &= m_c \times Q_a + c_c. \end{aligned} \quad (6.2)$$

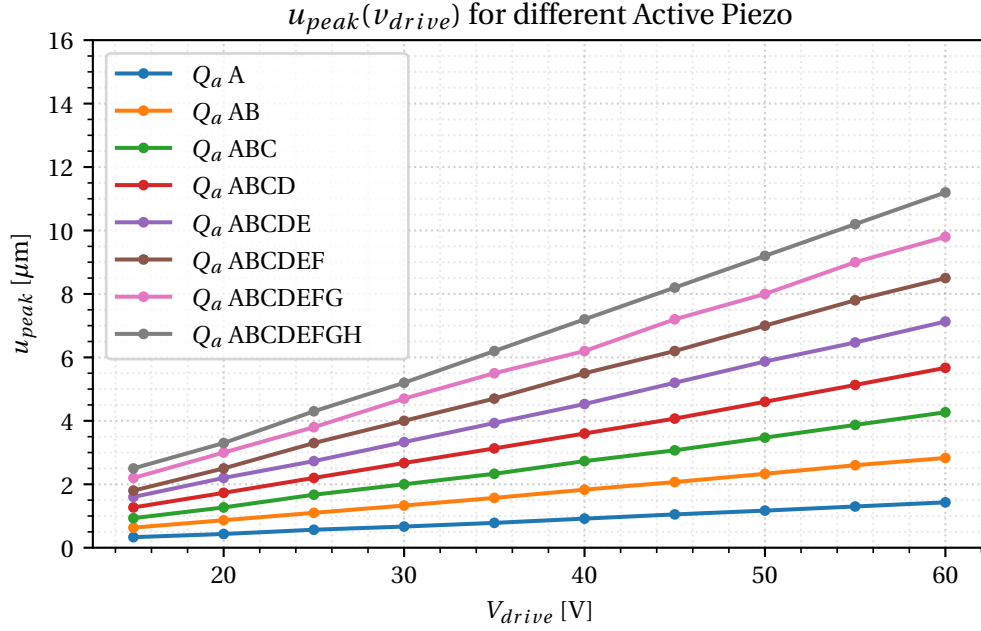


Figure 6.12: Peak amplitude for different driving voltages (V_{drive}) and number of active actuators (Q_a).

In Eq. 6.2, the constants m_m , c_m , m_c , c_c are calculated from the experimental data as follows:

$$\begin{aligned}
 m_m &= \frac{[u_{peak}(V_{min}, Q_{min}) - u_{peak}(V_{max}, Q_{min})] - [u_{peak}(V_{min}, Q_{max}) - u_{peak}(V_{max}, Q_{max})]}{(Q_{min} - Q_{max}) \times (V_{min} - V_{max})}, \\
 c_m &= \frac{u_{peak}(V_{min}, Q_{min}) - u_{peak}(V_{max}, Q_{min})}{V_{min} - V_{max}} - m_m \times Q_{min}, \\
 m_c &= \frac{[u_{peak}(V_{min}, Q_{min}) - m(Q_{min}) \times V_{min}] - [u_{peak}(V_{min}, Q_{max}) - m(Q_{max}) \times V_{min}]}{Q_{min} - Q_{max}}, \\
 c_c &= [u_{peak}(V_{min}, Q_{min}) - m(Q_{min}) \times V_{min}] - m_c \times Q_{min},
 \end{aligned} \tag{6.3}$$

where, V_{min} , Q_{min} , V_{max} , Q_{max} are the minimum and maximum values for V_{drive} and Q_a respectively. Let $V_{min} = 15$ V, $V_{max} = 60$ V, $Q_{min} = 1$ active piezo, $Q_{max} = 7$ active piezo. Then, the model fitted to the experimental data becomes:

$$u_{peak}(V_{drive}, Q_a) = (0.02363 \times Q_a + 0.00028) \times V_{drive} - 0.04333 \times Q_a + 0.01762. \tag{6.4}$$

The fitted model can be used to estimate $u_{peak}(V_{drive}, Q_a)$ for the complete range of the experimental measurements (Q_a [1 -8] and V_{drive} [15 V to 60 V]).

The amplitude values estimated by the model ('+' marker) and the measured u_{peak} values ('x' marker) are presented in a 3D plot in Fig. 6.13. The proposed model fits the experimental data with an RMSE of $0.079 \mu\text{m}$. Particularly, looking at data points that are outside of the range of data that was used to fit the model ($Q_a = 8$ and $V_{drive} [55 \text{ V}, 60 \text{ V}]$), the RMSE between the predicted and measured values is $0.12 \mu\text{m}$. This result confirms that the model can accurately (an RMSE error under $0.2 \mu\text{m}$ is acceptable for our requirements) model u_{peak} in terms of V_{drive} and Q_a .

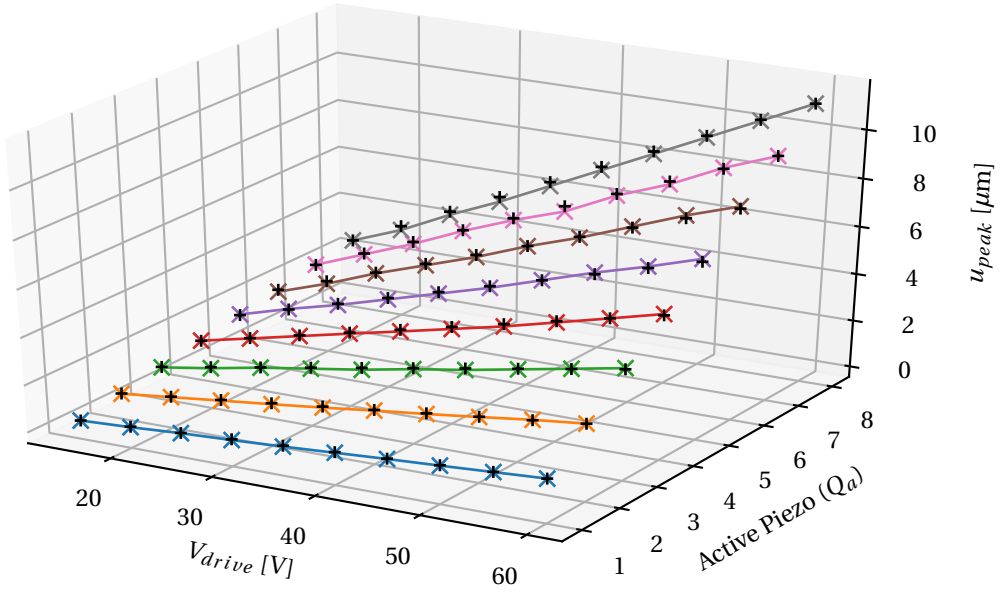


Figure 6.13: $u_{peak}(V_{drive}, Q_a)$ model validation. The measured values are presented with an 'x' marker and the predicted values with a '+' marker.

This model is used during the perception experiments to define the driver's parameters to obtain a desired peak amplitude. Furthermore, the experimental results permit us to obtain the amplitude range for the eight-piezo Tr haptics demonstrator. The range for u_{peak} is $[0.33 \mu\text{m} \text{ to } 11.20 \mu\text{m}]$, which is enough for the perception experiments where the desired range of variation is $[1 \mu\text{m} \text{ to } 11 \mu\text{m}]$.

Discussion on the Achieved Characteristics

As discussed on the time-reversal engineering tradeoffs, the theoretical maximum achievable C_{rMAX} can be increased by augmenting the bandwidth of the driving signal or by changing T_c (increasing the surface area of the glass plate or changing the material). To get C_r as close as possible to C_{rMAX} , the relation $Q \times T_s > T_c$ should be satisfied. In this case, the amount of piezo ($Q = 8$) is not enough to satisfy this condition, so it could be increased. Another option is to increase T_s . Nevertheless, keeping it equal to τ is preferred to ensure that the wave field for

a localized peak will completely dissipate before sending the next peak. Especially considering that we are not using the traditional approach of repeating the same wave field, but we want to send peaks at different locations.

The obtained parameters are good enough for the perception experiments where a variation in the amplitude of $[1 \mu\text{m to } 10 \mu\text{m}]$ is desired. Furthermore, a spatial resolution of 9 mm is enough to validate a novel pattern with $\pm 5 \text{ mm}$ variation in the random locations around the desired spot.

IR Data Acquisition

The final step is to acquire a dataset of 1-Bit TrIR signals between the eight actuators and several locations on the center of the glass plate. These signals can be stored on the FPGA to obtain a localized peak at any point within the active area.

The HiL experimental setup is used once again, and the acquisition area is displayed in Fig. 6.14. The IR acquisition is made in 16 points along the X and Y directions with a spatial resolution of $R_{scan} = 5 \text{ mm}$. The Acquisition starts at location $x_a = 2.5 \text{ mm}$ and $y_a = 2.5 \text{ mm}$ (with respect to the relative zero from the active area). A total of 256 IR signals are acquired. All the signals are cropped to $T_s = 1 \text{ ms}$, time-reversed, and quantized by one bit.

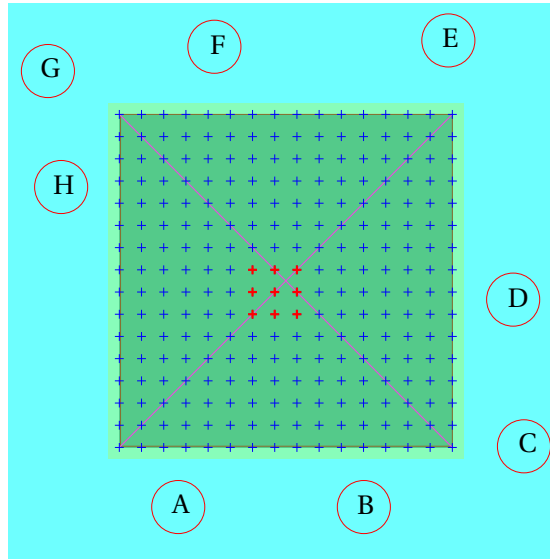


Figure 6.14: Distribution of the focus locations to acquire the 1-Bit TrIR dataset for the eight-piezo haptics demonstrator.

In the figure, the red circles represent the locations where the piezo actuators are fixed to the glass plate, and each piezo has its name. The blue crosses represent the locations where the IR signals are acquired. The red crosses represent a grid of nine locations whose corresponding signals will be used in the perception experiments in the following section.

6.4 Perception Experiments

With a haptic demonstrator that can create localized peaks within the human perception threshold, it is the moment to validate the hypothesis on the perception of the randomly-localized peaks vs. single-location peaks. This is done by conducting a survey research where a group of voluntary participants is asked to evaluate the perception of two different vibration patterns while one finger is in contact with a vibrating glass plate.

6.4.1 Single Vs Randomly-distributed Peak

The main experiment is a paired comparison test, where two vibration patterns are presented (A and B). The participants are asked to classify which of the two patterns is felt more alerting (i.e., more attention-catching), A or B, or if they feel the same, answer 'SAME'. For each participant, 20 different trials are presented. Each trial contains two stimuli (A and then B). One stimulus is the SINGLE pattern, and the other is the RANDOM pattern. Within the 20 trials, the stimuli are presented in balanced-randomized order (e.g., Trial 1: A - SINGLE then B - RANDOM, Trial 2: A - RANDOM then B - SINGLE, ...). The 20 trials are repeated 4 times. Both stimuli have the same duration, $T_{pattern}$ [ms], and there is a pause T_{pause} between the two stimuli within a trial. $T_{pattern}$ has to be longer than 1000 ms to ensure the mechanoreceptor's temporal summation effect has stabilized [145, 146]. After each trial, the user reports which pattern is felt as more alerting.

The 'SINGLE' vibration pattern is created with a single localized peak in the center of the plate. In contrast, the 'RANDOM' pattern is created with peaks at a randomly chosen location within five spots around the center. Each time, the location is chosen by drawing a number from a white noise generator, which returns a uniformly distributed pseudo-random pattern. Both vibration patterns can be reproduced at the desired frequency.

Experimental Parameters and Experimental Setup

During the perception experiments, several parameters must be defined and controlled. a) the amplitude of the peak (u_{peak}) is controlled by changing the driving voltage of the piezo (V_{drive}) or the number of piezo actuators that are used to recreate the peak (active piezo Q_a). b) The vibration frequency, f_r is modified by changing a delay between each localized peak. c) The duration of the stimulus ($T_{pattern}$) is controlled by setting the number of peaks to be created. d) The pause between both stimuli (T_{pause}) is determined by a delay between the stimulus reproduction.

To obtain the stimuli, the 1-bit TrIR signals for the 9 locations highlighted in Fig. 6.14 are used to create the localized peaks, which are repeated at an f_r rate.

Two options are defined for the RANDOM pattern: an 'x' configuration or a '+' configuration. The selected option defines the group of five signals that are used during the experiment.

The spatial distribution of the SINGLE pattern and the two configurations for the RANDOM pattern are presented in Fig. 6.15.

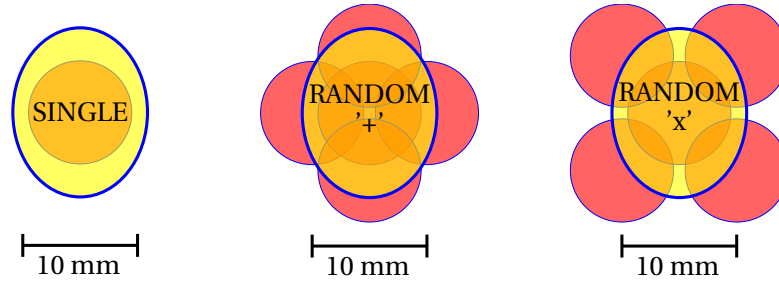


Figure 6.15: Spatial distribution for the SINGLE pattern and the two configurations for the RANDOM pattern.

The finger maximum contact size is displayed in a yellow ellipse, and the localized peak locations are displayed with a red circle of diameter R_s . The '+' configuration has a higher overlap between the localized peaks, and all the points are closer to the center. In contrast, the 'x' configuration offers more separation between the points (i.e., more variation on the peak location).

The finger pressing force, F , is crucial because it affects the effective contact area between the finger and the surface, which is highly correlated with the vibrotactile perception threshold. Hence, the pressing force must remain constant during the perception experiment.

To control this parameter, the haptic demonstrator is mounted on a PMMA frame attached to one end of a load cell. The other end of the load cell is mounted on a rigid base. An Arduino[®]-based DSP is used to acquire the pressing force, which is graphically presented to the participant in real time. This information helps her/him keep a constant pressing force throughout the experiment. This parameter remains fixed during the whole experiment. In addition, an arm support is mounted on the base and extends over the glass surface without touching it. The purpose of this support is to hold the arm and wrist and restrict the motion of the finger. The user is able to decide the pressing force by moving the tip of the finger.

A small square is drawn in the center of the glass plate to guide the finger's position. The human subject is asked to cover this square during the experiment completely. The experimental setup is presented in Fig. 6.16.

The experiment is controlled by a master PC running Labview. The Labview script controls the power supply and the FPGA. The FPGA is configured to generate digital driving signals to create the selected vibrational pattern.

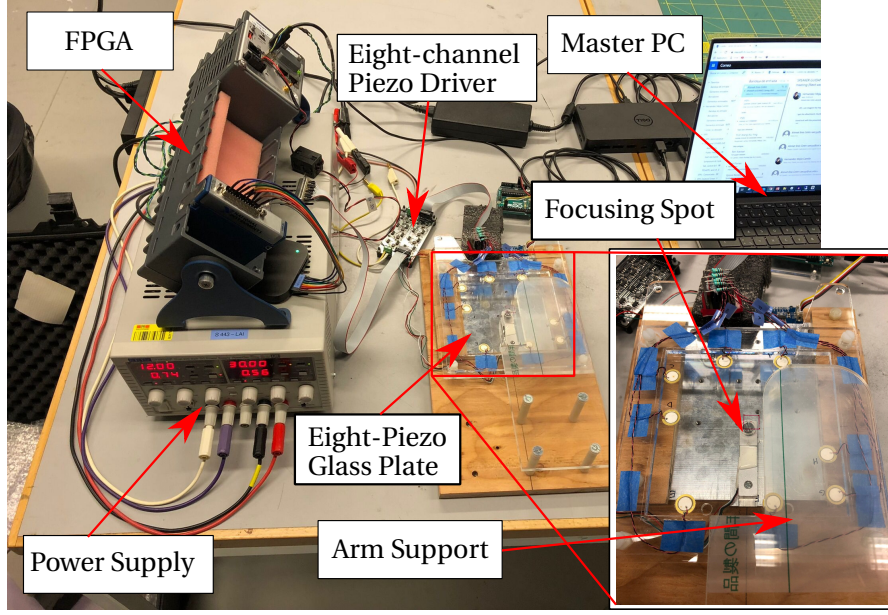


Figure 6.16: Experimental setup for the perception experiments.

Preliminary Experiments and Parameter Definition

Before running the final experiments, an extensive piloting phase was executed to test the effect of each parameter on the differentiation of both patterns. One single parameter was changed at a time. This phase aims to ensure the biggest difference possible so that the user can concentrate on judging which pattern is more alerting during the perception experiment.

On a first pilot V_{drive} is set to 30 V, Q_a to 8 piezo, f_r to 250 Hz, RANDOM pattern to '+', $T_{pattern} = T_{pause} = 1000$ ms. Four participants were asked to compare the two vibrational patterns presented in several trials (10 trials). On this initial pilot, the pressing force is changed in the range of [0.3 N to 2.5 N]. For each occasion, a full set of trials is run, and the force is constant throughout the trial. For a smaller pressing force, the difference between the stimuli is less clear, and 3 out of 4 participants reported the highest differentiation at $F = 2.0$ N.

Following that, the contact area for $F = 2$ N is measured for eight different human subjects (adults between [18 - 42] years old, including females and males). The mean contact area is $S_{finger}(2N) = 207 \text{ mm}^2$ and a standard deviation of 46.13 mm^2 . The displacement at the focusing spot is also measured for an $F = 2$ N (using the CLV). It was found that u_{peak} is attenuated by 25 % compared to the finger-less measurements (Fig. 6.12).

For the second pilot, F remained constant. A pair of experiments were executed where V_{drive} is set to 40 V and 20 V. It was found that with a higher V_{drive} , hence a bigger u_{peak} , it is easier to perceive the difference between the two stimuli, which was expected.

For the third pilot, f_r was tested for 100 Hz, 250 Hz, and 500 Hz. In this case, it is clear that for

both patterns, the highest intensity is perceived at $f_r = 250$ Hz (confirming the findings from previous studies [30]).

After the three pilots were run, it was found that $T_{pause} = 1000$ ms was too long and that the ability to compare the two patterns decreased. Then, this parameter is set at $T_{pause} = 600$ ms, and $T_{pattern}$ is set to 1200 ms to ensure the effect of temporal summation is not present.

In the fifth pilot, the RANDOM pattern 'x' was found to provide a better differentiation between the SINGLE and RANDOM patterns. It can be explained by the higher variation in the position and lower overlapping with the center point. In many cases, the RANDOM pattern '+' was perceived as SAME compared to the SINGLE pattern. Thus, RANDOM 'x' is selected. An interesting finding is that randomly changing the location of the peak at every peak repetition (i.e., changing x_a, y_a 250 times per second) was too fast to be perceived, which masked the effect of the randomness. Hence, an additional parameter is introduced to maximize the differentiation between RANDOM and SINGLE. This parameter is the number of repetitions before changing to the following random location, n_r . The sweet spot was found for $n_r = 10$ (i.e., the peak is repeated ten times at a given location before changing to the following random location).

During the pilot phase, it was identified that the difference between the SINGLE and RANDOM is clear but that telling which one is more alerting than the other is challenging and requires much concentration. Furthermore, some differences in the evoked sensation were noted. One pattern evoked a tickling sensation in the tip of the finger, while the other evoked a deeper vibration inside the finger.

Experimental Protocol

The experiment is executed as follows: First, the protocol is explained to the participant, and the explanation is always the same: "You will be presented with two vibrational stimuli, one after the other. For both stimuli, the duration is the same. Your goal is to judge which one is more alerting (Attention catching). Imagine you have a task to complete on a touch screen: which vibration alerts you more that something happened? You have to answer A, B, or SAME (If you feel the same or you can not decide for one)."

Second, the participant is trained to maintain a constant pressing force with the surface and to keep the finger position constant during the experiment (2 min are given to get familiar with the force control).

Third, the participant is presented with ten trials to get used to the sensation.

Forth, the perception experiment starts. The participant is presented with 20 trials, and the answer (A, B, or SAME) is registered after each trial. After 20 trials, the participant is given a 2 minutes pause to rest the finger. The experiment is repeated three times more (for a total of 80 trials).

Chapter 6. Human Perception Experiments and Time-reversal Haptics Demonstrator

Lastly, a qualitative assessment of the difference between the SINGLE and RANDOM patterns is carried out. For this purpose, the participant is presented several times with a unique trial, where pattern A is SINGLE, and pattern B is RANDOM. The question is "which stimulus is more alerting, A or B?" and "what sensations were evoked by each stimulus?"

To avoid auditory biasing (due to the noise coming from the vibrating surface), the participant wears noise-canceling headphones playing a continuous track of Pink noise during the whole experiment. Pink noise is a random signal where the power spectral density is inversely proportional to the frequency, thus carrying an equal amount of noise energy per octave (halve or double the frequency). This noise spectrum was the most effective in masking the sound emitted by the haptic demonstrator.

Experiments

For the main perception experiment, 12 voluntary participants (aged 26 - 52 years old) are invited to evaluate the vibrational patterns. All the participants used their right-hand index finger, and each experiment took 25 minutes on average. The experimental setup is mounted on top of the vibration-dampening table. A screen is placed in front of the setup to display the pressing force and indicate when stimulus A or B is being reproduced. The participants are seated in front of the setup, and the experimental protocol is followed. The experimental parameters are summarized in Table 6.4.

Table 6.4: Experimental parameters for the SINGLE Vs. RANDOM perception experiment.

| Parameter | $T_{pattern}$ [ms] | T_{pause} [ms] | F [N] | f_r [Hz] | n_r [times] | RANDOM Pattern | V_{drive} [V] | Q_a [piezo] |
|---------------|-----------------------|---------------------|------------|---------------|------------------|-------------------|--------------------|------------------|
| Defined Value | 1200 | 600 | 2.0 | 250 | 10 | 'x' | 60 | 8 |

Note that V_{drive} and Q_a are set at the maximum value to ensure that the vibration is perceived and to get a higher differentiation between the two patterns. The results for the 12 participants are summarized in Table 6.5. The responses A and B are replaced with the label for the pattern presented on the selected stimulus. The total amount answers per category are displayed for each participant (e.g., the total number of times that RANDOM was chosen within all trials).

Table 6.5: SINGLE Vs. RANDOM perception experiment results for the 12 participants (P1,...P12).

| Alternative | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | P11 | P12 | TOTAL |
|-------------|----|----|----|----|----|----|----|----|----|-----|-----|-----|-------|
| RANDOM | 36 | 29 | 11 | 36 | 30 | 38 | 37 | 35 | 28 | 33 | 20 | 33 | 366 |
| SAME | 29 | 19 | 3 | 31 | 28 | 23 | 15 | 20 | 12 | 5 | 23 | 40 | 286 |
| SINGLE | 15 | 32 | 6 | 13 | 22 | 19 | 28 | 25 | 40 | 42 | 37 | 7 | 248 |

SINGLE vs. RANDOM Experimental Result Analysis

A descriptive analysis of the data shows that the RANDOM pattern was selected more often (41 % of the trials) than the SINGLE pattern (32 % of the trials). An important finding is that the SAME category was chosen in 27 % of the trials, which confirms the task's difficulty. 30% of all participants perceived SINGLE as more alerting than RANDOM, which indicates that in some cases, the perceived intensity or alertness is a relative metric that depends on the participant's perception.

The Box plot in Fig. 6.17 presents the distribution of the experimental data (median, min, max, lower and upper quartiles).

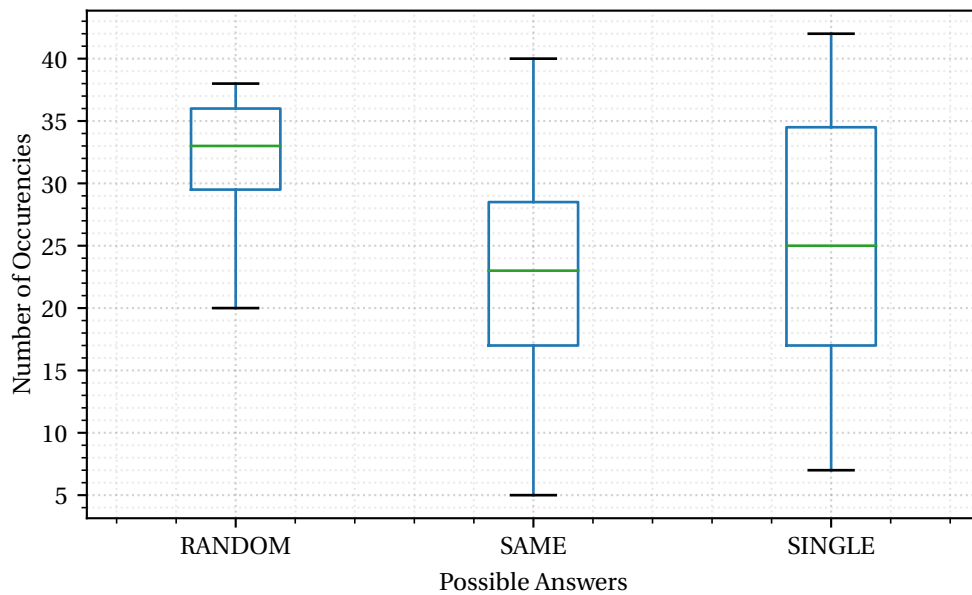


Figure 6.17: SINGLE Vs. RANDOM perception experiment results in a Boxplot.

It can be observed that the data for the SINGLE and SAME categories are more dispersed than the RANDOM pattern. It is hard to explain this behavior without further experiments. However, it is possible to say that the RANDOM response appears more concise for the participants, while the SINGLE response is often mixed with the SAME response.

To test the significance of the difference between the response categories, a "Multinomial Logistic Regression" model, which is commonly used for categorical data with more than two categories, was fit to the experimental data.

The fitted model estimates the following proportions for the response categories: RANDOM: 0.40667, SAME: 0.27556, and SINGLE: 0.31778. The difference between the RANDOM and SAME categories was significant, with a P-value = 2.221×10^{-6} , as well as the difference between RANDOM and SINGLE, with a P-value = 0.0017 (i.e., in multinomial regression the P-values is

the probability that the Null hypothesis is true, in general P-value below 0.05 confirms that the effect is significant). This allows the conclusion that there are significantly more "RANDOM" responses than "SINGLE" responses to the question "Which pattern is perceived as more alerting?". These results confirm the hypothesis that a random-excitation signal is perceived as more alerting than a single-excitation signal for the experimental parameters presented in Table 6.4.

The qualitative assessment also shows some interesting results. On the one hand, most of the participants (75 %) reported that the SINGLE pattern is felt more regular. They describe it as a superficial sensation at the tip of the finger. On the other hand, most of the participants (66.6 %) perceived that the RANDOM pattern goes deeper in the finger and that they perceive it as more dynamic. Several participants remarked that the RANDOM pattern provides a granular sensation. All participants agree that determining which pattern is more alerting is difficult, especially in the first round. Nevertheless, at the end of the experiment, they could determine that there were two different patterns. The fact that a representative portion of the participants (> 50 %) determined that the two patterns were perceived in different locations or depths of the finger is quite interesting. This finding should be explored more in-depth because it can inspire novel types of sensations when providing haptic feedback.

Discussion on Novel Pattern Applications

The most interesting fact from the randomly-localized stimulation pattern is that it is viable for time-reversal haptics and can be implemented on other surface haptics techniques. For instance, it can be easily implemented on pin arrays, soft pneumatic actuators, and other vibrotactile interfaces. The alertness effect on devices with a higher spatial resolution could be amplified.

The fact that the RANDOM pattern was perceived in different places of the finger, compared to the SINGLE peak, shows that this pattern could be used as an alternative stimulation to evoke other sensations in the user, not only increasing the alertness. Furthermore, alternative patterns such as lateral sweeping or circular sequence were tested during the development of the perception experiment. These patterns also evoked different sensations. Thus, further exploring this approach is a field of study worth furthering in the future.

Perception Threshold Experiment Pilot

After validating the hypothesis that a vibration created with randomly-localized peaks is perceived as more alerting than one created with a single location peak, the next step is to quantify the difference in the perception. To do this, a pilot experiment is proposed to get an idea of the perception threshold for each vibrational pattern (i.e., the minimum amplitude where the vibration is perceived).

Each stimulation pattern is evaluated in a separate experiment, where the vibrational pattern is presented several times with increasing amplitude. The participant is asked to report when she/he starts feeling a vibration and the respective u_{peak} is registered. Each time, the stimulation pattern is presented during $T_{pattern}$ milliseconds, followed by a short pause (T_{pause}) before displaying the next stimulus with a higher u_{peak} . V_{drive} is kept constant during $T_{pattern}$, and it is only changed during the pause. For the pilot, the threshold experiment is repeated two times for each pattern. Before running the experiment, it is important to find the amplitude range (i.e., the value for Q_a) where the vibration is not felt for the lowest u_{peak} value, but it is perceived for the highest u_{peak} value. Using the RANDOM pattern ($F = 2N$, the RANDOM pattern 'x', and $n_r = 10$), it was found that the perception threshold for the haptic demonstrator occurs for $Q_a = 2$, where the amplitude range is $[0.475 \mu m \text{ to } 2.12 \mu m]$.

Four participants (male and female, aged 25 - 35 years old) are asked to touch the glass plate with a constant pressing force. During the experiment, they are asked to wear noise-canceling headphones with a continuous track of Pink noise. The experimental parameters are summarized in Table 6.6.

Table 6.6: Experimental parameters for the threshold perception pilot.

| Parameter | $T_{pattern}$ [ms] | T_{pause} [ms] | F [N] | f_r [Hz] | n_r [times] | RANDOM Pattern | V_{drive} [V] | Q_a [piezo] |
|---------------|-----------------------|---------------------|------------|---------------|------------------|-------------------|--------------------|------------------|
| Defined Value | 1200 | 600 | 2.0 | 250 | 10 | 'x' | [15 - 60] | 2 |

In this experiment, ten amplitude levels are presented. To do this, V_{drive} is incremented in steps of 5 V (increments of $0.18 \mu m$, Eq. 6.4) starting at 15 V and ending at 60 V. The results are presented in Table 6.7.

Table 6.7: Threshold perception pilot results.

| | P1.1 | P1.2 | P2.1 | P2.2 | P3.1 | P3.2 | P4.1 | P4.2 |
|------------------------------------|------|------|------|------|------|------|------|------|
| Threshold Level SINGLE [μm] | 1.37 | 1.55 | 1.18 | 1.37 | 1.37 | 1.75 | 1.55 | 1.55 |
| Threshold Level RANDOM [μm] | 1.37 | 1.55 | 1.37 | 1.37 | 1.18 | 1.37 | 1.37 | 1.55 |

This pilot evaluates the threshold on a rough scale, and the exact detection threshold has yet to be found. Nevertheless, an interesting trend was observed. Three out of eight times, the RANDOM pattern threshold level was smaller than the SINGLE threshold. One out of eight times, the opposite behavior is observed. Half of the time, both patterns have the same threshold level. These results do not permit us to conclude a perception threshold yet, but they motivate the hypothesis that the RANDOM pattern has a lower threshold than the SINGLE pattern.

6.4.2 Threshold Experiment Outlook

In future work, a more detailed threshold measurement is planned following standard methods from signal detection theory [147, 148]. The experiment is designed based on the Method of Constant Stimuli (MSC) [149], where the strategy is to present the participant with several trials that contain stimuli ranging in intensity. The stimuli are presented in random order. The threshold is determined by finding the intensity at which the participant can detect the stimulus in 50 percent of the trials.

The main challenge when designing an experiment with the MSC is determining the range for the amplitude of the stimuli. To determine this range, a preliminary experiment is carried out using a similar approach to our detection threshold pilot. i.e., the vibrational pattern is presented several times with increasing amplitude, and the participant is asked to report when she/he starts feeling the vibrations. Then, the opposite experiment is executed, the vibrational pattern is presented several times with decreasing amplitude, and the participant is asked to report when the vibration is not felt anymore. The threshold to start feeling the vibration is expected to be higher than the threshold to stop feeling it. These two values are recorded and determine the range for u_{peak} that will be used for that participant during the main threshold experiment (following the MSC).

6.5 Outlook on the Haptics Demonstrator as a DMI

The developed haptics demonstrator permits to obtain localized vibration over the surface of a glass plate. This device has the potential to be transformed into a Digital Musical Instrument (DMI). For this purpose, a three-dimensional touch sensing layer (i.e., 3D = X, Y: position of the finger + Z: Force) must be integrated to capture the motion of the fingers. The TrIR dataset and the driving electronics can be used to create localized peaks on the desired locations to provide feedback to the user.

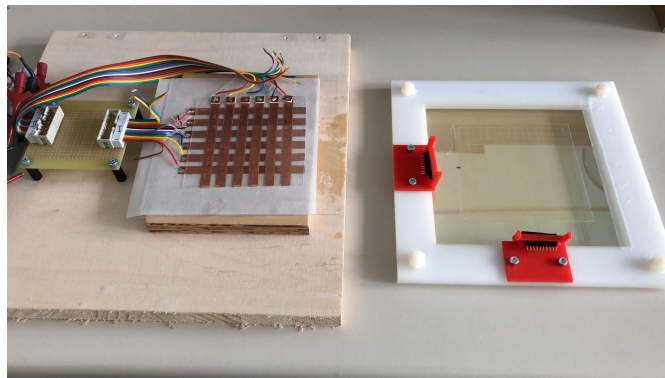


Figure 6.18: 3D touch sensing layer prototype.

Due to time constraints, it was impossible to finish the development of the 3D sensing layer for the haptic demonstrator surface. The sensing layer is under development and will continue

as future work. The concept uses a thin ($100\ \mu\text{m}$) layer of silicon (PDMS) with transparent electrodes on both sides. The electrodes are divided into a matrix to develop an array of capacitive sensors. The finger-pressing force deforms the silicone, which changes the capacitance at a given location. When the capacitance is measured at a high rate, it is possible to detect simultaneous contacts over the surface. The sensing layer prototype is presented in Fig. 6.18.

With this idea in mind, an experiment was carried out to validate the effect on u_{peak} of adding a sensing layer on top of the glass surface. The single-piezo haptic demonstrator (Section 6.3.2) was used. First, a silicon layer of $100\ \mu\text{m}$ is laid on top of the glass plate. Then, the amplitude of the localized peak is compared to the original displacement when there is no silicon layer. The results are presented in Fig. 6.19.

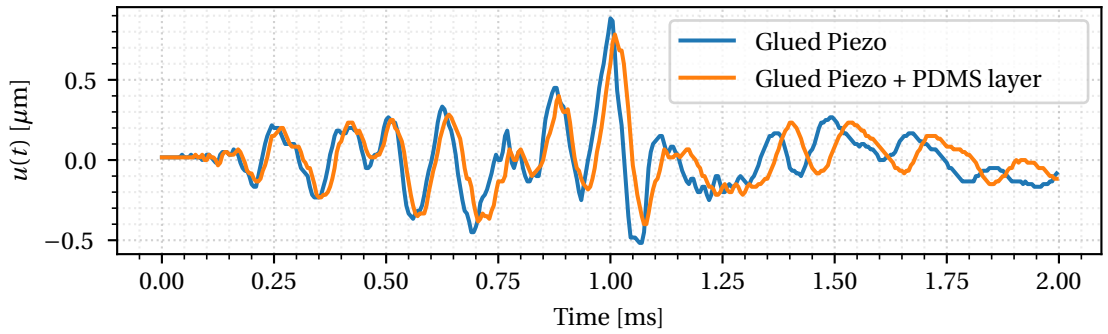


Figure 6.19: u_{peak} for the single-piezo glass plate with and without a thin ($100\ \mu\text{m}$) overlay of silicon (PDMS).

It was found that the silicon layer reduces the peak amplitude by 11%. Furthermore, an increasing delay can be observed. This delay should be considered since it indicates a change in the plate's attenuation constant, affecting the repetition rate (f_r) and the optimal duration of the driving signal (T_s).

6.6 Conclusion

In this chapter, the perceptual effect of the novel pattern is evaluated.

A time-reversal haptics demonstrator was developed guided by the Time-Reversal Method (TRM) knowledge and the engineering tradeoffs introduced in Chapter 2. The development was done in two phases. First, a demonstrator with a single actuator was built to compare two different strategies to bond the piezoelectric actuators to the glass plate and three different methods to fix the glass plate to a rigid frame. It was found that using glue to fix the piezo is better than using thin ($50\ \mu\text{m}$) double-sided tape and that using a thick (1 mm) tape to fix the glass plate helps to reduce the attenuation constant (τ) which permits to repeat the localized peaks at a higher rate. An interesting behavior was found for a mechanical fixing that increases the peak amplitude (u_{peak}) while keeping τ smaller than other rigid fixing conditions. Second,

a demonstrator with multiple actuators and an active area of $80 \times 80 \text{ mm}^2$ was developed. The haptics demonstrator is characterized, and the key metrics are compared to the theoretical values. It is possible to create localized peaks in the range of $[0.33 \text{ to } 11.20 \mu\text{m}]$ with a contrast ratio of 12.4, a temporal resolution (R_t) of $21.2 \mu\text{s}$, and a spatial resolution (R_s) of 9 mm.

A paired-comparison experiment was executed to compare the vibration created with randomly-localized peaks (RANDOM) with a vibration created with single-location peaks (SINGLE). The haptics demonstrator was used to develop an experimental setup for the perception experiments. The experimental parameters are described, and an extensive piloting phase is carried out to find the parameters that provide the biggest differentiation between the SINGLE and RANDOM patterns. The experimental protocol is described, and an experiment with 12 voluntary participants is carried out. The perception experiment shows that there are significantly more "RANDOM" responses than "SINGLE" responses to the question "Which pattern is perceived as more alerting?" Furthermore, the difference between the two categories is significant [P-value = $0.0017 < 0.05$]. These results confirm the hypothesis that a random-excitation signal is perceived as more alerting than a single-excitation signal for the experimental parameters presented in Table 6.4.

The results from a perception threshold pilot allow formulating the hypothesis that the RANDOM pattern has a lower perception threshold than the SINGLE pattern. The experimental protocol to validate this hypothesis is proposed.

The potential to use the haptic demonstrator to develop a digital musical instrument is discussed, and the effect of adding a sensing layer on top of the glass plate on u_{peak} is evaluated with an experiment. A thin ($100 \mu\text{m}$) sensing layer only decreases the peak amplitude by 11 %.

In the future, the paired-comparison experiment could be expanded to include manipulating the experimental parameters. This deeper study can provide information on which parameter maximizes the perceptual difference between SINGLE and RANDOM. Moreover, the novel pattern can evoke alternative sensations when providing haptic feedback, so it is interesting to study the applications of these sensations and validate the effect of the randomly-localized stimulation on alternative surface haptic feedback methodologies (e.g., pin array interfaces).

Publications related to this chapter :

- C. H. Mejia, J. Jayet, P. Germano, A. Thabuis, and Y. Perriard, "Linear Impact Generator for Automated Dataset Acquisition of Elastic Waves in Haptic Surfaces," in *2019 22nd International Conference on Electrical Machines and Systems (ICEMS)*, pp. 1–5, Aug. 2019.
- C. Hernandez-Mejia, X. Ren, S. Papetti, H. Jarvelainen, T. Chavdarova, P. Germano, and Y. Perriard, "GANs for Localized Vibrotactile Feedback in Haptic Surfaces: Single Vs Randomly-distributed stimulation", in *IEEE Transactions on Haptics (ToH)*. [Pending]

7 General Conclusion

This exploratory work introduces state-of-the-art machine learning techniques into the time-reversal haptics field of study and it shows the potential of these approaches to bring new knowledge into the haptics research field. By studying different deep-learning methods to generate the signals that are used to create a localized peak, it was possible to bring incremental improvements to the quality of the peaks and to develop a novel vibrational pattern. The novel pattern involves creating the vibration with randomly-localized peaks (RANDOM stimulus) in contrast to the traditional approach of repeating a localized peak (SINGLE stimulus). A human perception experiment proved that the RANDOM stimulus is perceived as more alerting than the SINGLE stimulus. Furthermore, the randomly localized pattern can evoke different sensations compared to the single peak pattern, which opens the door to developing new vibrotactile feedback cues.

The scientific approach was to study the problem in separate building blocks. At each step, one or several research questions were formulated, and the associated hypothesis was proposed. The general conclusion is compiled by answering each of these questions:

"How can ML models represent the relevant features of the impact data?"

By developing a Machine Learning (ML) framework and choosing the right representation of the signal, ML can be used to extract relevant features from impact data. In particular, a novel impact position detection strategy was developed. ML models were trained to predict the location where an impact occurred after receiving the impact signal as an input. The FFT magnitude of the impact signals proved to be a compact yet powerful signal representation that is associated with the natural modes of the surface. It was possible to determine that the contact duration affects the precision of the impact source location task. In the future, this feature should be considered as an input to the model to maintain high accuracy and make the detection model more robust.

"How can Machine Learning (ML) and Deep Learning (DL) be used to represent the time-reversed signals used to create a localized peak?" and "How can DL be used to optimize the contrast ratio and ensure the desired location of the peak?"

A deep neural network was pre-trained to learn the mapping between the desired location of the peak and the time-reversed signal (TrIR signal) that can create a localized peak. Using the FFT magnitude representation, it is possible to simplify the generated signal into a low-dimensional vector containing the FFT Magnitude for 88 most relevant natural modes. This simplification allowed Reinforcement Learning (RL) to optimize the pre-trained NN. The RL framework modified the DNN in such a way that the generated signals had the same performance as the full spectrum TrIR signal or even higher (The optimized model improves the contrast ratio by 2 % compared to the traditional signals and by 14 % compared to the pre-trained neural network)

"How can Deep-Generative Models be used to generate time-reversed signals that can create a localized displacement peak?" and "What is the effect of the diversity from the GANs on the properties of the localized peaks?"

An alternative generative method that learns the probabilistic distribution of a TrIR dataset is studied. Using an existing GAN model (WaveGAN) it was validated that this deep-generative model can accurately capture the distribution of the TRIRDS, and generate peaks with a similar quality compared to the original dataset. By adding a conditioning label to the WaveGAN model, it is possible to control the type of signals that are generated, thus the location where the localized peak occurs. Including a condition increases the similarity of the generated signal with the reference TrIR and the quality of the localized peak. Nevertheless, the generative model still provides some diversity (i.e., the generated signals are not always the same, they have slight variations with respect to the original TrIR) which affects the properties of the peak. It was found that the generated signals create peaks with varying amplitude and a slight variation in the location of the peak (0 to +2 mm around the target location). This variation inspired the RANDOM stimulation pattern. This novel pattern can increase the user's alertness, potentially reducing the perception threshold. This reduces the peak amplitude requirements, which reduces the amount of power consumed (lower driving voltage or lower number of piezo).

"What is the effect of randomly distributed peaks on the human perception of localized vibrations?"

A haptics demonstrator surface was developed based on the time-reversal method theory and engineering tradeoffs. This device was used to build an experimental setup to validate the perceptual effect of the novel pattern. With authorization from the EPFL ethics committee (HREC), a paired comparison experiment is designed and executed with 12 voluntary participants. The results confirm the hypothesis that the randomly-localized peaks are perceived as more alerting (or stronger by some users) than the single-location peaks. An important finding is that the RANDOM stimulus is perceived deeper in the finger, while the SINGLE stimulus is perceived more towards the tip of the finger (superficial). This finding can inspire the development of novel vibrotactile feedback cues where alternative sensations are evoked. Furthermore, a perception threshold pilot permitted the formulation of the hypothesis that the novel pattern has a lower perception threshold.

7.1 Original Contributions

- An iterative methodology to acquire data, evaluate the data transformation and train a model to extract information from impact signals was developed. This methodology was used to create a novel impact position detection strategy using a single piezoelectric transducer and ML models.
- A compact representation of the impact signal was proposed based on the domain knowledge of wave propagation. The FFT magnitude of the signal permits to enhance the feature extraction and improves the impact position detection precision. This is explained by the relation of the stimulated natural modes with the location where the peak occurred.
- A new approach to optimize the localized peaks that are obtained with time-reversed impulse response signals was developed. A Deep Neural Network was trained to learn the one-to-one mapping between the desired location of the peak and the TrIR signal to generate the peak. The reinforcement learning framework was used to iteratively optimize the NN and improve the peak's quality (contrast ratio) while ensuring the right location and maximizing the amplitude of the peak.
- A novel approach to storing and generating time-reversed signals was developed using Generative Adversarial Networks (GANs). The conditional model generates peaks with the same quality as the original signal, but the diversity involved in the GAN framework inspired the development of a novel pattern of vibrotactile stimulation.
- A novel stimulation pattern that increases the user's alertness compared to the continuous vibration obtained with a single localized peak was proposed. This novel stimulus can reduce power consumption when providing vibrational feedback to a user and evoke alternative sensations.
- A haptic demonstrator was developed, and a perceptual experimental protocol was proposed to evaluate the effect of different vibrotactile patterns. This demonstrator can be used as the base for the development of a Digital Musical Instrument (DMI).

7.2 Outlook

This work explored the benefits of using Deep Learning strategies to optimize the localized peaks obtained with the time-reversal method. As a Ph.D. student, one desires to study every single aspect and continue digging further into the different explored approaches, but time is limited. The following list presents some aspects that we consider worth exploring in the future.

Regarding the proposed impact position detection strategy, developing an ML system with multiple outputs is interesting. Beyond predicting the impact location, it is of high relevance

to extract the impact force and the number of contacts (when there are several). If a limited number of contacts is defined, the model can be designed to extract multiple contact locations from the acquired impact signal.

Concerning the peak properties optimization, the RL optimization phase is strongly limited by the time it takes to make the vibration scan (i.e., HiL experimental validation of the signal). It is vital to reduce this step to enable a thorough exploration of alternative optimization models, signal representations, reward functions, and NN architectures. To achieve this, one could replace the HiL with a simplified simulation environment (i.e., using the wave propagation equations) early in the optimization stages.

Proving that GANs can accurately model the time-reversed signals opens the doors to experimenting with modifications inside the generator that could help discover novel stimulation patterns. For instance, one can explore the effect of interpolations on the discrete conditions (i.e., providing two desired locations simultaneously to the generator). Moreover, one could explore the effect of diversity when multiple piezo signals are generated (each piezo signal can be considered as an additional channel for the GAN). Lastly, it is compelling to experiment with additional conditioning parameters such as signal (e.g., the audio output of a digital musical instrument), which can modulate the amplitude of the generated peaks.

Future work on the time-reversal haptics demonstrator should focus on further exploring the effect of the boundary conditions on the amplitude of the localized peak and the attenuation constant. The preliminary finding of the relation between the fixing of the glass plate (Section 6.3.2) and the peak amplitude is very interesting. It appears that the mechanical fixing is adding some pre-load into the system that contributes to increasing peak amplitude. From the driving electronics point of view, it is highly relevant to design the driver to recover the energy of the piezo when discharging it. This could help to optimize energy usage when creating a localized peak.

From the perception point of view, it is highly interesting to evaluate the perceptual effect of randomly changing the amplitude of the peaks. This stimulation could also catch the user's attention and help reduce the perception threshold. A more extended paired-comparison experiment could be developed to experiment with different pressing forces, peak amplitudes, pattern frequencies, and random patterns. This deeper study can provide information on the effect of each parameter on perceived alertness which allows finding which parameter maximizes this perceptual difference. Moreover, the novel pattern shows some potential to evoke different types of feelings when providing haptic feedback. Thus, it is relevant to study the effect of the randomly-localized stimulation on alternative surface haptic feedback methodologies (e.g., pin array interfaces and other direct actuation methods).

A Neural Network Architectures

This Appendix presents the architectures for the different Neural Networks (NN) that are used in this thesis.

A.1 Impact Position Detection NN Architectures

A.1.1 Fully Connected NN

The architecture of the fully connected NN (Vanilla NN) that was used for impact position detection is displayed in the following table:

Table A.1: Impact position detection fully connected NN architecture.

| Operation | Layer Size |
|-----------------------|----------------|
| Input Layer | 1 |
| Fully Connected Layer | 1024 |
| ReLU Layer | – |
| Batch Normalization | – |
| Gaussian Noise (0.5) | – |
| Fully Connected Layer | 128 |
| ReLU Layer | – |
| Batch Normalization | – |
| Gaussian Noise (0.3) | – |
| Fully Connected Layer | 64 |
| ReLU Layer | – |
| Batch Normalization | – |
| Gaussian Noise (0.2) | – |
| Fully Connected Layer | 32 |
| ReLU Layer | – |
| Batch Normalization | – |
| Gaussian Noise (0.1) | – |
| RegressionLayer | output size: 1 |

Appendix A. Neural Network Architectures

The fully connected NN was trained using the MAE loss function, and the weights of the network are updated using the Adam optimizer [107]. The batch size is 32. The model was trained for 900 epochs (iterations) with a learning rate of $l_r = 0.001$. Then, for another 900 iterations with a $l_r = 0.00001$.

A.1.2 2D Convolutional NN (2DCNN) Architecture

The architecture of the 2D CNN that was used for impact position detection is displayed in the following table:

Table A.2: Impact position detection 2DCNN architecture.

| Operation | kernel Size | Number of filters or Layer Size | Stride |
|-----------------------|-------------|---|--------|
| Input Layer | | input size : 234×128 or 260×128 | |
| Conv2D | [3 3] | Number of Filters: 32 | [2 2] |
| ReLU Layer | | | |
| Batch Normalization | | | |
| DropOut (0.5) | | | |
| Conv2D | [3 3] | Number of Filters: 32 | [2 2] |
| ReLU Layer | | | |
| Batch Normalization | | | |
| DropOut (0.5) | | | |
| Conv2D | [3 3] | Number of Filters: 64 | [2 2] |
| ReLU Layer | | | |
| Batch Normalization | | | |
| DropOut (0.5) | | | |
| Conv2D | [3 3] | Number of Filters: 64 | [2 2] |
| ReLU Layer | | | |
| Batch Normalization | | | |
| DropOut (0.5) | | | |
| Flatten | | | |
| Fully Connected Layer | | Layer Size: 256 | [2 2] |
| ReLU Layer | | | |
| Batch Normalization | | | |
| DropOut (0.5) | | | |
| RegressionLayer | | output size : 1 | |

The 2DCNN was trained using the MAE loss function and the weights of the network are updated using the Adam optimizer [107]. The batch size is 32. The model was trained for 1000 epochs (iterations) with a learning rate of $l_r = 0.001$. Then, for another 1000 iterations with a learning rate of $l_r = 0.00001$.

A.2 RL Actor NN Architectures

A.2.1 RL Actor 1: Fully Connected NN Architecture

The architecture of the fully connected NN actor (Actor 1) is presented in the following table:

Table A.3: RL actor 1: fully connected NN architecture.

| Operation | Layer Size |
|-----------------------|------------------|
| Input Layer | input size : 1 |
| Fully Connected Layer | Layer size : 256 |
| ReLU Layer | |
| Fully Connected Layer | 600 |
| ReLU Layer | |
| Fully Connected Layer | 600 |
| ReLU Layer | |
| Fully Connected Layer | 600 |
| ReLU Layer | |
| Fully Connected Layer | 88 |
| RegressionLayer | output size: 88 |

A.2.2 RL Actor 2: 1D Convolutional Neural Network (1DCNN) Architecture

The architecture of the 1D Convolutional NN actor (Actor 2) is presented in the following table:

Table A.4: RL actor 2: 1DCNN architecture.

| Operation | kernel Size | Number of filters or Layer Size | Stride |
|-----------------------|-------------|------------------------------------|--------|
| Input Layer | | input size : 1 | |
| Fully Connected Layer | | Layer size : 256 | |
| ReLU Layer | | | |
| TransposedConv2dLayer | [5 1] | 128 | [4 1] |
| ReLU Layer | | | |
| TransposedConv2dLayer | [5 1] | 128 | [3 1] |
| ReLU Layer | | | |
| TransposedConv2dLayer | [3 1] | 64 | [4 1] |
| ReLU Layer | | | |
| TransposedConv2dLayer | [3 1] | 64 | [2 1] |
| ReLU Layer | | | |
| TransposedConv2dLayer | [3 1] | 32 | [1 1] |
| ReLU Layer | | | |
| TransposedConv2dLayer | [3 1] | 32 | [1 1] |
| ReLU Layer | | | |
| TransposedConv2dLayer | [1 1] | 1 | [1 1] |
| RegressionLayer | | output size : 88 | |

A.3 GAN Architectures

A.3.1 WaveGAN

The architecture for the WaveGAN model [82] is defined in terms of three parameters. 1) The number of channels, c , 2) the batch size $b = n_{batch}$, and 3) the dimensionality of the model d . For our experiments, we only use one piezo, thus $c = 1$, and the model size is fixed to $d = 64$.

The WaveGAN model was trained using the WGAN-GP [128] loss function (where WGAN-GP $\lambda = 10$) and the weights of the network are updated using the Adam optimizer [107]. The latent vector size is $n_z = 100$. The batch size (n_{batch}) is 64. The batch normalization was deactivated ($batch_{norm} = \text{False}$). The number of channels is, $c = 1$ (Note that for our application the number of channels is equal to the number of piezo). The model dimensionality is $d = 64$. The phase shuffle was set to 2 ($n_{phase} = 2$). The discriminator was trained 5 times per each training of the generator (D updates per G updates = 5). The model was trained for 250 k epochs (iterations) with a learning rate of $l_r = \alpha = 0.0001$, a $\beta_1 = 0.5$ $\beta_s = 0.9$. Original hyper-parameters from [82].

The architecture of the WaveGAN generator (G) is presented in the following table:

Table A.5: WaveGAN generator (G) architecture.

| Operation | kernel Size | Output Shape | Stride |
|-------------------------------------|-----------------|-----------------|--------|
| Input $z \sim \text{Uniform}(-1,1)$ | | $(n, 100)$ | |
| Fully Connected Layer | $[100, 256d]$ | $(n, 256d)$ | |
| Reshape | | $(n, 16, 16d)$ | |
| ReLU Layer | | $(n, 16, 16d)$ | |
| TransposedConv1DLayer | $[25, 16d, 8d]$ | $(n, 64, 8d)$ | 4 |
| ReLU Layer | | $(n, 64, 8d)$ | |
| TransposedConv1DLayer | $[25, 8d, 4d]$ | $(n, 256, 4d)$ | 4 |
| ReLU Layer | | $(n, 256, 4d)$ | |
| TransposedConv1DLayer | $[25, 4d, 2d]$ | $(n, 1024, 2d)$ | 4 |
| ReLU Layer | | $(n, 1024, 2d)$ | |
| TransposedConv1DLayer | $[25, 2d, d]$ | $(n, 4096, d)$ | 4 |
| ReLU Layer | | $(n, 4096, d)$ | |
| TransposedConv1DLayer | $[25, d, c]$ | $(n, 16384, c)$ | 4 |
| Tanh Layer | | $(n, 16384, c)$ | |

The architecture of the WaveGAN discriminator (D) is presented in the following table:

Table A.6: WaveGAN Discriminator (D) architecture.

| Operation | kernel Size | Output Shape | Stride |
|----------------------------------|-----------------|-----------------|--------|
| Input x or $G(z)$ | | $(n, 16384, c)$ | |
| TransposedConv1DLayer | $[25, c, d]$ | $(n, 4096, d)$ | 4 |
| LeakyReLU Layer ($\alpha = 2$) | | $(n, 4096, d)$ | |
| Phase Shuffle (n_{phase}) | | $(n, 4096, d)$ | |
| TransposedConv1DLayer | $[25, d, 2d]$ | $(n, 1024, 2d)$ | 4 |
| LeakyReLU Layer ($\alpha = 2$) | | $(n, 1024, 2d)$ | |
| Phase Shuffle (n_{phase}) | | $(n, 1024, d)$ | |
| TransposedConv1DLayer | $[25, 2d, 4d]$ | $(n, 256, 4d)$ | 4 |
| LeakyReLU Layer ($\alpha = 2$) | | $(n, 256, 4d)$ | |
| Phase Shuffle (n_{phase}) | | $(n, 256, d)$ | |
| TransposedConv1DLayer | $[25, 4d, 8d]$ | $(n, 64, 8d)$ | 4 |
| LeakyReLU Layer ($\alpha = 2$) | | $(n, 64, 8d)$ | |
| Phase Shuffle (n_{phase}) | | $(n, 64, d)$ | |
| TransposedConv1DLayer | $[25, 8d, 16d]$ | $(n, 16, 16d)$ | 4 |
| LeakyReLU Layer ($\alpha = 2$) | | $(n, 16, 16d)$ | |
| Reshape | | $(n, 256d)$ | |
| Fully Connected Layer | $[256d, 1]$ | $(n, 1)$ | |

A.3.2 Conditional WaveGAN (cWaveGAN)

The architecture for the cWaveGAN model is defined in terms of four parameters. 1) The number of channels, c , 2) the batch size $b = n_{batch}$, 3) the dimensionality of the model d . An additional parameter is defined, 4) the number of labels, $j = n_{labels}$. For our experiments, we only use one piezo, thus $c = 1$, and the model size is fixed to $d = 64$. The number of labels is $j = n_{labels} = 177$ (As explained in Section 5.3.3).

The cWaveGAN model was trained using the WGAN-GP [128] loss function (where WGAN-GP $\lambda = 10$) and the weights of the network are updated using the Adam optimizer [107]. The latent vector size is $n_z = 100$. The batch size (n_{batch}) is 128. The batch normalization was activated ($batch_{norm} = \text{True}$). The phase shuffle was deactivated ($n_{phase} = 0$). The discriminator was trained 5 times per each training of the generator (D updates per G updates = 5). The model was trained for 30 k epochs (iterations) with a learning rate of $l_r = \alpha = 0.00025$, a $\beta_1 = 0.5$ $\beta_s = 0.9$.

NOTE: For both WaveGAN and cWaveGAN, all layers include biases.

Appendix A. Neural Network Architectures

In the conditional WaveGAN, the condition c is taken as a parallel input. (i.e., for G , parallel to the latent vector z . For D , parallel to either the dataset sample x or generated sample $G(z)$). c is transformed using an embedding NN. Then, it is concatenated with the input of G and D right before the convolutional layers.

The architecture of the conditional generator G is presented in the following two tables:

Table A.7: Label Embedding for generator G .

| Operation | Output Shape |
|--|---------------|
| Input c | $(n, 1)$ |
| Embedding ($n_{labels} = j$) | $(n, j, 20j)$ |
| Fully connected | $(n, 16)$ |
| Reshape (Out is \mathbf{G}_{label}) | $(n, 16, 1)$ |

Table A.8: cWaveGAN generator (G) architecture.

| Operation | kernel Size | Output Shape | Stride |
|--|-----------------|--------------------|--------|
| Input $z \sim \text{Uniform}(-1,1)$ | | $(n, 100)$ | |
| Fully Connected Layer | $[100, 256d]$ | $(n, 256d)$ | |
| Reshape | | $(n, 16, 16d)$ | |
| Batch Normalization | | $(n, 16, 16d)$ | |
| ReLU Layer | | $(n, 16, 16d)$ | |
| Concatenate with \mathbf{G}_{label} | | $(n, 16, 16d + 1)$ | |
| TransposedConv1DLayer | $[25, 16d, 8d]$ | $(n, 64, 8d)$ | 4 |
| ReLU Layer | | $(n, 64, 8d)$ | |
| Batch Normalization (if $batch_{norm}$) | | $(n, 64, 8d)$ | |
| TransposedConv1DLayer | $[25, 8d, 4d]$ | $(n, 256, 4d)$ | 4 |
| ReLU Layer | | $(n, 256, 4d)$ | |
| Batch Normalization (if $batch_{norm}$) | | $(n, 256, 4d)$ | |
| TransposedConv1DLayer | $[25, 4d, 2d]$ | $(n, 1024, 2d)$ | 4 |
| ReLU Layer | | $(n, 1024, 2d)$ | |
| Batch Normalization (if $batch_{norm}$) | | $(n, 1024, 2d)$ | |
| TransposedConv1DLayer | $[25, 2d, d]$ | $(n, 4096, d)$ | 4 |
| ReLU Layer | | $(n, 4096, d)$ | |
| Batch Normalization (if $batch_{norm}$) | | $(n, 4096, d)$ | |
| TransposedConv1DLayer | $[25, d, c]$ | $(n, 16384, c)$ | 4 |
| Tanh Layer | | $(n, 16384, c)$ | |

The architecture of the conditional discriminator D is presented in the following two tables:

Table A.9: Label Embedding for discriminator D .

| Operation | Output Shape |
|--|-----------------|
| Input c | $(n, 1)$ |
| Embedding ($n_{labels} = j$) | $(n, j, 20j)$ |
| Fully connected | $(n, 16384)$ |
| Reshape (Out is \mathbf{D}_{label}) | $(n, 16384, 1)$ |

Table A.10: cWaveGAN discriminator (D) architecture.

| Operation | kernel Size | Output Shape | Stride |
|---------------------------------------|-----------------|---------------------|--------|
| Input x or $G(z)$ | | $(n, 16384, c)$ | |
| Concatenate with \mathbf{D}_{label} | | $(n, 16384, c + 1)$ | |
| TransposedConv1DLayer | $[25, c, d]$ | $(n, 4096, d)$ | 4 |
| LeakyReLU Layer ($\alpha = 2$) | | $(n, 4096, d)$ | |
| Phase Shuffle (n_{phase}) | | $(n, 4096, d)$ | |
| TransposedConv1DLayer | $[25, d, 2d]$ | $(n, 1024, 2d)$ | 4 |
| LeakyReLU Layer ($\alpha = 2$) | | $(n, 1024, 2d)$ | |
| Phase Shuffle (n_{phase}) | | $(n, 1024, d)$ | |
| TransposedConv1DLayer | $[25, 2d, 4d]$ | $(n, 256, 4d)$ | 4 |
| LeakyReLU Layer ($\alpha = 2$) | | $(n, 256, 4d)$ | |
| Phase Shuffle (n_{phase}) | | $(n, 256, d)$ | |
| TransposedConv1DLayer | $[25, 4d, 8d]$ | $(n, 64, 8d)$ | 4 |
| LeakyReLU Layer ($\alpha = 2$) | | $(n, 64, 8d)$ | |
| Phase Shuffle (n_{phase}) | | $(n, 64, d)$ | |
| TransposedConv1DLayer | $[25, 8d, 16d]$ | $(n, 16, 16d)$ | 4 |
| LeakyReLU Layer ($\alpha = 2$) | | $(n, 16, 16d)$ | |
| Reshape | | $(n, 256d)$ | |
| Fully Connected Layer | $[256d, 1]$ | $(n, 1)$ | |

B One-Bit Piezo Driving Electronics Development

The Printed Circuit Board (PCB) is designed to drive eight piezo actuators independently. The input V_{IN} is a digital signal (0 V to 5 V) which is generated by an FPGA (NI9045 Compact Rio + NI9041 High-speed DI/O Module). The output signals going to the piezo actuators, V_{piezo} are binary signals and range from 0 V to the supply voltage V_{drive} . The PCB is designed to receive a supply voltage of up to 60 V. Each output is driven by two N-channel MOSFET transistors (DiodesZetex ZXMN10A08) placed in a half-bridge configuration and their gate are controlled by a high-voltage half-bridge gate driver (Texas Instruments LM5104).

B.1 Single-piezo Diver

This version is meant to validate the single piezo driver's design before manufacturing the eight-channel PCB.

B.1.1 Components Layout

The typical application schematic for the LM5104 Gate Driver Data-sheet is presented in Fig. B.1. The external recovery diode is not necessary as the LM5104 Driver already has an internal diode which is sufficient for that purpose. The two gate diodes are also not necessary, the 100 Ω gate resistors are sufficient to discharge the MOSFETs' gate quickly enough, even at driving frequencies up to 200 kHz.

The electrical components arrangement on the PCB is inspired by the hints given in the LM5104 Gate Driver Data-sheet (Chapter 10.1: Layout Guidelines). The actual PCB design is shown on Fig. B.2. The LM5104 ($U1$) is placed as close as possible to the MOSFETs ($Q1$ and $Q2$) allowing the tracks that contain the two gate resistors ($R2$ and $R3$) to be as short as possible. The Bootstrap capacitor ($C2$) is also placed as close as possible to the LM5104.

The $P1$ connector is the driving signal input V_{IN} . Connectors $J1$, $J2$, and $J3$ are respectively the 12 V power supply for the LM5104 Driver, the V_{drive} power supply for the half-bridge

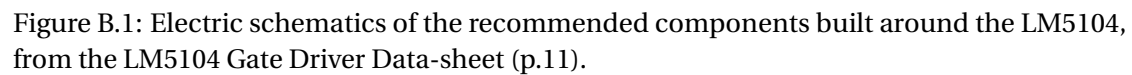


Figure B.2: Single piezo driver PCB design (Altium Designer).

B.1.2 Components Calculation

The most important component of the driver is the bootstrap capacitor. It shall be able to provide enough charge to the high-side MOSFET's gate at each switching cycle while keeping its voltage stable. In addition, it has to be small enough to keep its charging time low. The necessary calculations to choose the correct capacitance are given in the LM5104 Gate Driver Data-sheet (Chapter 8.2.2: Detailed Design Procedure). The total charge to provide to the MOSFET's gate is calculated as follows:

$$Q_{total} = Q_{gmax} + I_{HBO} \times \frac{D_{max}}{F_{SW}} = 7.7 \text{ nC} + 3 \text{ mA} \times \frac{0.5}{200 \text{ kHz}} = 15.2 \text{ nC} \quad (\text{B.1})$$

This charge has to be provided at a voltage of:

$$\Delta V_{HB} = V_{DD} - V_{DH} - V_{HBL} = 12 \text{ V} - 1.1 \text{ V} - (7.1 \text{ V} - 0.4 \text{ V}) = 4.2 \text{ V} \quad (\text{B.2})$$

Which gives the total capacitance:

$$C_{bootstrap} \geq \frac{Q_{total}}{\Delta V_{HB}} = \frac{15.2 \text{ nC}}{4.2 \text{ V}} = 3.6 \text{ nF} \quad (\text{B.3})$$

The Bootstrap capacitor must be larger than 3,6 nF, a standard value of 6,8 nF was been chosen.

The MOSEFTs' switching delay is set by the R_T resistor, a value of 10 k Ω has been chosen and sets a delay of about 85 ns reading from Fig. B.3 (this value is small compared to the 2500 ns half-period of a 200 kHz square signal).

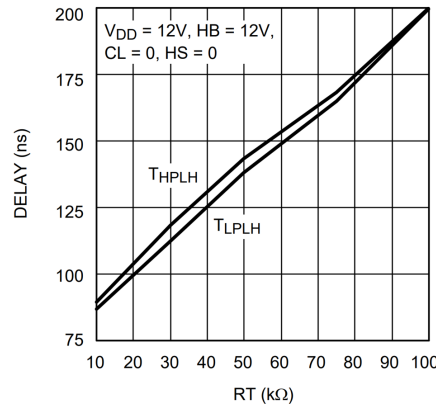


Figure B.3: Turn On Delay vs R_T Resistor Value, LM5104 Gate Driver Data-sheet (p.13).

Once the PCB has been manufactured, a test was carried out using a 200 kHz square signal as V_{IN} and a $V_{drive} = 30 \text{ V}$ for the piezo actuator. Fig. B.4 shows the voltage measurements on the PCB (V_{IN} , V_{piezo} , and the MOSFETs gates voltage (high-side, V_{HO} , and low-side, V_{LO}).

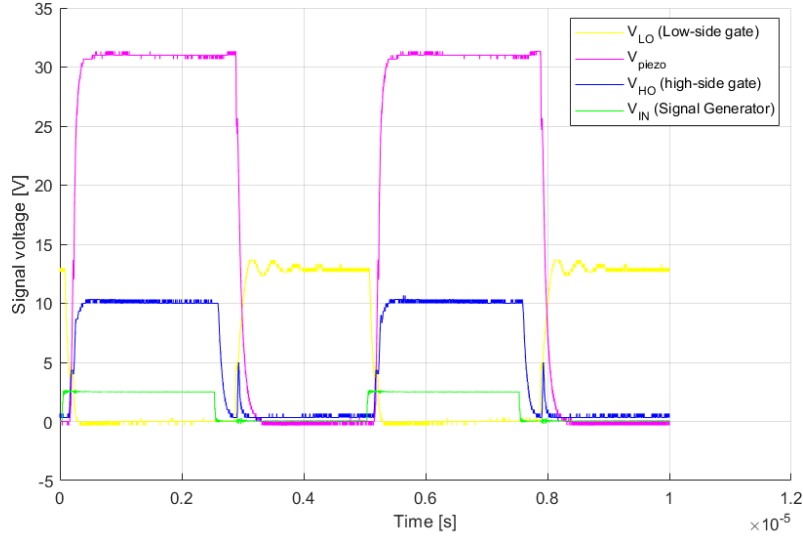


Figure B.4: Input signal (V_{IN} green trace), differential voltage measurement of the high-side and low-side gates (V_{HO} blue and V_{LO} yellow traces respectively) and output voltage (V_{piezo} pink trace) on which the piezo actuator is connected, on the single driver PCB.

The duty cycle of V_{IN} (green trace) is set to 0.5 but one can notice that V_{piezo} (pink trace) stays longer at 30 V than 0 V, setting a different duty cycle than the input signal (above 0.5). This effect can be a problem during the focusing process as the time-reversed IR signal in the piezo could be distorted. Fortunately, this delay is the same on each circuit of the PCB described in Section B.2, thus signal synchronization is still guaranteed.

A load resistor, R_L , is added in series with the piezo actuator. Its objective is to filter the voltage spikes on the output voltage, which are generated by the capacitive nature of the piezo actuator. The resulting circuit is presented in Fig. B.5. The selected value for R_L is $39\ \Omega$, which provides sufficient dampening without increasing the piezo charging time (i.e., the rising edge of the pink trace on Fig. B.4).

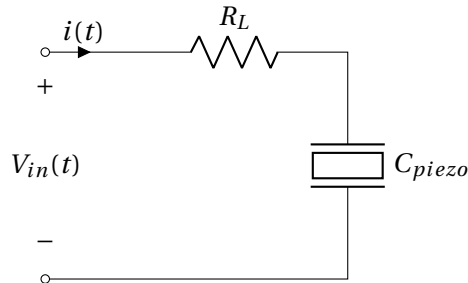


Figure B.5: Output circuit composed of a load resistor R_L of $39\ \Omega$ and the piezo actuator's capacitance C_{piezo} of 8 nF.

The load resistor is subject to high currents as the $V_{in}(t)$ voltage can go up to 60 V. The RC circuit is modeled by a simple transfer function, where $C = C_{piezo}$ and $R = R_L$:

$$\frac{I(s)}{V_{in}(s)} = \frac{C_s}{1 + RCs} \quad (B.4)$$

Using this transfer function one can simulate the current going through the resistor. Fig. B.6a presents the simulated values for $i_{piezo}(t)$ when $V_{in}(t)$ is a square signal at 200 kHz.

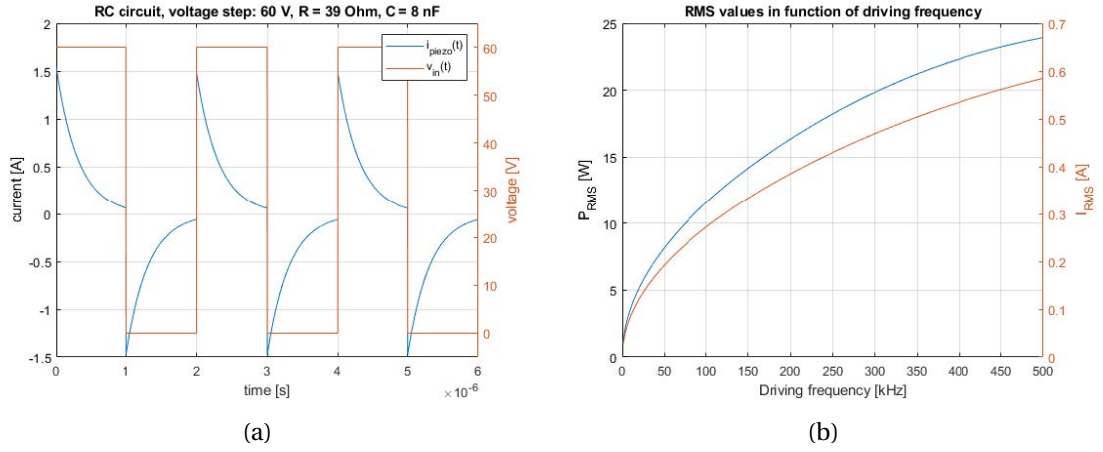


Figure B.6: (a) Current passing through the circuit when $V_{in}(t)$ is a 200 kHz square signal in V_{in} (b) RMS values of the current and power through the load resistor R_L in function of the square signal's frequency.

Considering a square signal with an amplitude of 0 V to 60 V as $V_{in}(t)$ and calculating the power dissipated into the resistor with $P_{R_L}(t) = R_L \times i^2(t)$ one can then obtain the Root Mean Square (RMS) power dissipated in the resistor on a duration of n periods T :

$$P_{RMS} = \sqrt{\frac{1}{nT} \int_0^{nT} P_{R_L}^2(t) dt} \quad (B.5)$$

Fig. B.6b is obtained after calculating the RMS power for a range of driving frequencies. The power dissipated in the resistor is significant and thus a high-power resistor shall be used for high driving frequencies. It has been noted that when using the setup for its nominal use (localized peak repetition at 250 Hz) a 1/2 W resistor is sufficient.

B.2 Eight-channel Piezo Driver

The eight-channel piezo driver is composed of an array of eight single piezo drivers. Each driver can independently amplify the driving signal for one piezo actuator. The PCB is presented in Fig. B.7).

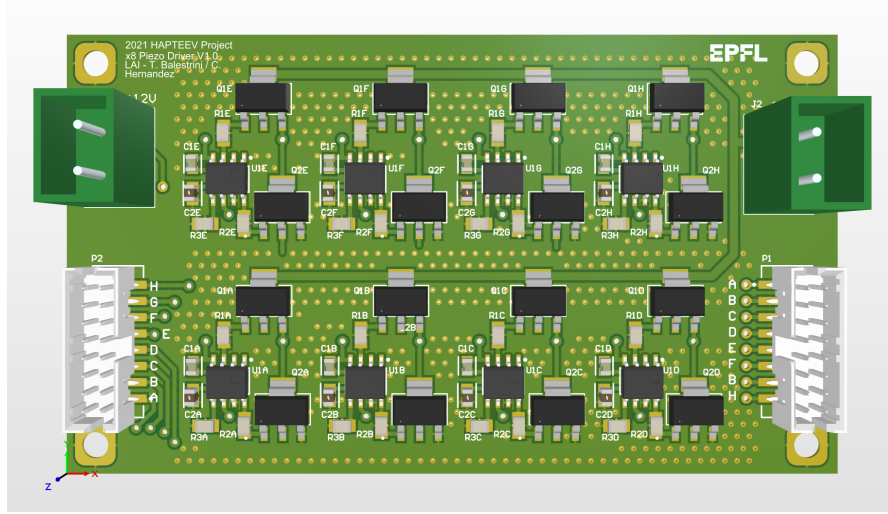


Figure B.7: PCB for the eight-channel piezo driver.

There are two connectors per side. On the left side, the 2-pin connector (green) is the 12 V power supply for the gate drivers, and the 16-pin connector (white) takes the digital inputs from the FPGA. On the right side, the 2-pin connector (green) is the V_{drive} power supply for the high side of the half-bridge, and the 16-pin connector (white) outputs the amplified digital signals to drive the piezoelectric actuators. The R_L resistors are placed close to each piezo actuator. As mentioned before, the current that is drained by the piezo actuators is significant, thus the 60 V power supply track distributing the voltage to the high-side of the half-bridge has been designed with a width of 1 mm, compared to the standard width of 0.5 mm for all other tracks.

Each independent circuit of this PCB has been tested following the same process as the single piezo driver. Fig. B.8 displays the test for one channel, it is observed that the output voltage width is consistent with the input width although a delay is present.

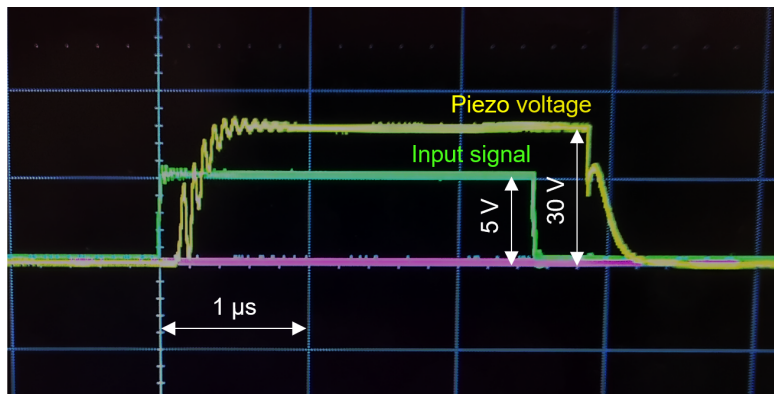


Figure B.8: Input signal (V_{IN} green trace) and output voltage (V_{piezo} yellow trace) for on one channel of the eight-channel piezo driver PCB.

Bibliography

- [1] H. Culbertson, S. B. Schorr, and A. M. Okamura, "Haptics: The Present and Future of Artificial Touch Sensation," *Annual Review of Control, Robotics, and Autonomous Systems* **1**(1), pp. 385–409, 2018.
- [2] C. Basdogan, F. Giraud, V. Levesque, and S. Choi, "A Review of Surface Haptics: Enabling Tactile Effects on Touch Surfaces," *IEEE Transactions on Haptics* **13**, pp. 450–470, July 2020.
- [3] M. S. Prewett, L. R. Elliott, A. G. Walvoord, and M. D. Coover, "A Meta-Analysis of Vibrotactile and Visual Information Displays for Improving Task Performance," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* **42**, pp. 123–132, Jan. 2012.
- [4] S. Papetti and C. Saitis, eds., *Musical Haptics*, Springer Series on Touch and Haptic Systems, Springer International Publishing, 2018.
- [5] E. N. Chavan and D. V. Rojarkar, "State of art of haptic technology," **4**(01), p. 7, 2017.
- [6] R. H. COOK, "An automatic stall prevention control for supersonic fighter aircraft," *Journal of Aircraft* **2**(3), pp. 171–175, 1965.
- [7] V. Z. Perez Ariza and M. Santís-Chaves, "INTERFACES HÁPTICAS: SISTEMAS CINESTÉSICOS Vs. SISTEMAS TÁCTILES," *Revista EIA* **13**, Mar. 2017.
- [8] "Force Dimension - Haptic Devices." <https://www.forcedimension.com/products>. Accessed: 2022-08-04.
- [9] M. Benali-Khoudja, M. Hafez, J.-M. Alexandre, and A. Kheddar, "Tactile interfaces: a state-of-the-art survey," p. 10.
- [10] "Gaming Piezo Shoulder Buttons." <https://www.boreas.ca/pages/gaming-piezo-shoulder-buttons>. Accessed: 2022-08-04.
- [11] T. Carter, S. A. Seah, B. Long, B. Drinkwater, and S. Subramanian, "UltraHaptics: multi-point mid-air haptic feedback for touch surfaces," in *Proceedings of the 26th annual ACM symposium on User interface software and technology - UIST '13*, pp. 505–514, ACM Press, (St. Andrews, Scotland, United Kingdom), 2013.

Bibliography

- [12] “Digital worlds that feel human | Ultraleap.” <https://www.ultraleap.com/>. Accessed: 2022-08-04.
- [13] R. E. Jones, “Intimate Control for Physical Modeling Synthesis (Madrona Labs founder master thesis),” Master’s thesis, University of Victoria, Canada, 2008.
- [14] “Alphasphere.” <https://alphasphere.com/>. Accessed: 2022-08-04.
- [15] “Madrona Labs - Soundplane.” <https://madronalabs.com/soundplane>. Accessed: 2022-08-04.
- [16] “Reactable Live.” <http://reactable.com/live/>. Accessed: 2022-08-04.
- [17] “Haken Audio.” <https://www.hakenaudio.com>. Accessed: 2022-08-04.
- [18] “Sensel Morph.” <https://morph.sensel.com/>. Accessed: 2022-08-04.
- [19] author, “KAOSSILATOR PRO | KORG (USA).” https://www.korg.com/us/products/dj/kaossilator_pro_plus/. Accessed: 2022-08-04.
- [20] Evelyn Glennie, “Hearing Essay.” <https://www.evelyn.co.uk/hearing-essay/>, 1993. Accessed: 2019-06-05.
- [21] T. T. Ahmaniemi, “Gesture Controlled Virtual Instrument with Dynamic Vibrotactile Feedback,” in *NIME*, 2010.
- [22] J. Lozada, M. Hafez, and X. Boutillon, “A novel haptic interface for musical keyboards,” in *2007 IEEE/ASME international conference on advanced intelligent mechatronics*, pp. 1–6, Sept. 2007.
- [23] D. Overholt, E. Berdahl, and R. Hamilton, “Advancements in Actuated Musical Instruments,” *Organised Sound* **16**, pp. 154–165, Aug. 2011.
- [24] S. Serafin and J. O. Smith, “Influence of Attack Parameters on the Playability of a Virtual Bowed String Instrument,” in *ICMC*, 2000.
- [25] S. Papetti, S. Schiesser, and M. Fröhlich, “Multi-Point Vibrotactile Feedback for an Expressive Musical Interface,” in *Proceedings of the International Conference on New Interfaces for Musical Expression, NIME 2015*, pp. 235–240, The School of Music and the Center for Computation and Technology (CCT), Louisiana State University, (Baton Rouge, Louisiana, USA), 2015.
- [26] S. J. Bolanowski, G. A. Gescheider, R. T. Verrillo, and C. M. Checkosky, “Four channels mediate the mechanical aspects of touch,” *The Journal of the Acoustical Society of America* **84**, pp. 1680–1694, Nov. 1988.
- [27] G. A. Gescheider, J. H. Wright, and R. T. Verrillo, *Information-processing channels in the tactile sensory system: A psychophysical and physiological analysis*, Information-processing channels in the tactile sensory system: A psychophysical and physiological analysis, Psychology Press, New York, NY, US, 2009. Pages: ix, 135.

-
- [28] G. A. Gescheider, S. J. Bolanowski, J. V. Pope, and R. T. Verrillo, "A four-channel analysis of the tactile sensitivity of the fingertip: frequency selectivity, spatial summation, and temporal summation," *Somatosensory & Motor Research* **19**, pp. 114–124, Jan. 2002.
- [29] D. Purves, G. J. Augustine, D. Fitzpatrick, L. C. Katz, A.-S. LaMantia, J. O. McNamara, and S. M. Williams, "Mechanoreceptors Specialized to Receive Tactile Information," *Neuroscience. 2nd edition*, 2001.
- [30] R. T. Verrillo, "Vibrotactile thresholds measured at the finger," *Perception & Psychophysics* **9**, pp. 329–330, July 1971.
- [31] L. Skedung, M. Arvidsson, J. Y. Chung, C. M. Stafford, B. Berglund, and M. W. Rutland, "Feeling Small: Exploring the Tactile Perception Limits," *Scientific Reports* **3**, p. 2617, Sept. 2013.
- [32] J. J. Zarate, O. Gudozhnik, A. S. Ruch, and H. Shea, "Keep in Touch: Portable Haptic Display with 192 High Speed Taxels," in *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems, CHI EA '17*, pp. 349–352, Association for Computing Machinery, (New York, NY, USA), May 2017.
- [33] "Using pot-magnets to enable stable and scalable electromagnetic tactile displays," *IEEE Transactions on Haptics*, 2016.
- [34] P. Smithmaitrie, J. Kanjantoe, and P. Tandayya, "Touching force response of the piezo-electric Braille cell," *Disability and rehabilitation. Assistive technology* **3**, pp. 360–5, Dec. 2008.
- [35] J. Xu, Y. Kimura, K. Tsuji, K. Abe, T. Shimizu, H. Hasegawa, and T. Mineta, "Fabrication and characterization of SMA film actuator array with bias spring for high-power MEMS tactile display," *Microelectronic Engineering* **227**, p. 111307, Apr. 2020.
- [36] N. Besse, S. Rosset, J. J. Zárate, E. Ferrari, L. Brayda, and H. Shea, "Understanding Graphics on a Scalable Latching Assistive Haptic Display Using a Shape Memory Polymer Membrane," *IEEE Transactions on Haptics* **11**, pp. 30–38, Jan. 2018.
- [37] C. Wagner, S. Lederman, and R. Howe, "A tactile shape display using RC servomotors," in *Proceedings 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. HAPTICS 2002*, pp. 354–355, Mar. 2002.
- [38] Y. Jansen, T. Karrer, and J. Borchers, "MudPad: Localized Tactile Feedback on Touch Surfaces," in *Adjunct Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology, UIST '10*, pp. 385–386, ACM, 2010.
- [39] A. A. Stanley and A. M. Okamura, "Controllable Surface Haptics via Particle Jamming and Pneumatics," *IEEE Transactions on Haptics* **8**, pp. 20–30, Jan. 2015.
- [40] W. Brenner, S. Mitic, A. Vujanic, and G. Popovic, "Micro-Actuation Principles for High-resolution Graphic Tactile Displays," 2003.

- [41] C. Hudin and S. Panëels, “Localisation of Vibrotactile Stimuli with Spatio-Temporal Inverse Filtering,” in *Haptics: Science, Technology, and Applications*, pp. 338–350, Springer International Publishing, Cham, 2018.
- [42] O. Bau, I. Poupyrev, A. Israr, and C. Harrison, “TeslaTouch: electrovibration for touch surfaces,” in *Proceedings of the 23rd annual ACM symposium on User interface software and technology - UIST '10*, p. 283, ACM Press, 2010.
- [43] C. Winter, *Friction feedback actuators using squeeze film effect*. PhD thesis, ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE, Lausanne, Switzerland, Mar. 2014.
- [44] M. Wiertlewski, R. Fenton Friesen, and J. E. Colgate, “Partial squeeze film levitation modulates fingertip friction,” *Proceedings of the National Academy of Sciences* **113**, pp. 9210–9215, Aug. 2016.
- [45] F. Giraud, M. Amberg, B. Lemaire-Semail, and G. casiez, “Design of a transparent tactile stimulator,” in *2012 IEEE Haptics Symposium (HAPTICS)*, pp. 485–489, Mar. 2012. ISSN: 2324-7355.
- [46] H. Xu, M. A. Peshkin, and J. E. Colgate, “UltraShiver: Lateral force feedback on a bare fingertip via ultrasonic oscillation and electroadhesion,” in *2018 IEEE Haptics Symposium (HAPTICS)*, pp. 198–203, Mar. 2018.
- [47] J.-H. Woo and J.-G. Ih, “Vibration rendering on a thin plate with actuator array at the periphery,” *Journal of Sound and Vibration* **349**, pp. 150–162, Aug. 2015.
- [48] S. E. Emgin, A. Aghakhani, T. M. Sezgin, and C. Basdogan, “HapTable: An Interactive Tabletop Providing Online Haptic Feedback for Touch Gestures,” *IEEE Transactions on Visualization and Computer Graphics* **25**, pp. 2749–2762, Sept. 2019.
- [49] E. Enferad, C. Giraud-Audine, F. Giraud, M. Amberg, and B. L. Semail, “Generating controlled localized stimulations on haptic displays by modal superimposition,” *Journal of Sound and Vibration* **449**, pp. 196–213, June 2019.
- [50] E. Enferad, C. Giraud-Audine, F. Giraud, M. Amberg, and B. Lemaire-Semail, “Differentiated haptic stimulation by modal synthesis of vibration field,” in *2018 IEEE Haptics Symposium (HAPTICS)*, pp. 216–221, Mar. 2018. ISSN: 2324-7355.
- [51] C. Hudin, J. Lozada, and V. Hayward, “Localized Tactile Feedback on a Transparent Surface through Time-Reversal Wave Focusing,” *IEEE Transactions on Haptics* **8**, pp. 188–198, Apr. 2015.
- [52] M. Fink, “Time reversal of ultrasonic fields. I. Basic principles,” *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control* **39**, pp. 555–566, Sept. 1992.
- [53] C. Draeger, J.-C. Aime, and M. Fink, “One-channel time-reversal in chaotic cavities: Experimental results,” *The Journal of the Acoustical Society of America* **105**, pp. 618–625, Jan. 1999.

- [54] M. Fink, "Acoustic imaging with time reversal methods: From medicine to NDT," *AIP Conference Proceedings* **1650**, pp. 13–23, Mar. 2015.
- [55] M. R. Bai and Y. K. Tsai, "Impact localization combined with haptic feedback for touch panel applications based on the time-reversal approach," *The Journal of the Acoustical Society of America* **129**, pp. 1297–1305, Mar. 2011.
- [56] X. Liu, *Acoustic emission source detection and wave generation on a working plate using piezoelectric actuators*. PhD thesis, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, July 2017.
- [57] C. Hudin, *Focalisation par retournement temporel dans les plaques minces : Application à la stimulation tactile*. PhD thesis, Université Pierre et Marie Curie - Paris VI, Mar. 2014.
- [58] H. Zophoniasson, *Static and dynamic haptic feedback using time reversal and electrovibration stimulations*. PhD thesis, Université Paris Saclay (COMUE), June 2017.
- [59] M. Fink and C. Prada, "Acoustic time-reversal mirrors," *Inverse Problems* **17**, pp. R1–R38, Jan. 2001.
- [60] H. C. Huang, "Elastic plates: Theory and application, Herbert Reismann, Wiley (New York), 1988. 400 pp, £54.20, ISBN 0 471856010," *International Journal for Numerical Methods in Engineering* **28**(11), pp. 2710–2710, 1989.
- [61] C. Hudin, J. Lozada, and V. Hayward, "Spatial, temporal, and thermal contributions to focusing contrast by time reversal in a cavity," *Journal of Sound and Vibration* **333**, pp. 1818–1832, Mar. 2014.
- [62] M. Fink, N. Quieffin, R. K. Ing, D. Clorennec, S. Catheline, G. Ribay, M. Fink, N. Quieffin, R. K. Ing, D. Clorennec, S. Catheline, and G. Ribay, "Acoustic impact localization in plates: properties and stability to temperature variation," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control* **54**, pp. 378–385, Feb. 2007.
- [63] K. F. Graff, *Wave Motion in Elastic Solids*, Ohio State University Press, 1975.
- [64] S. H. Rao, "High-definition haptics: Feel the difference!," *Texas Instruments Incorporated - Analog Applications Journal* **3Q**, p. 6, 2012.
- [65] Boréas Technologies, "Here's Why Piezo Should Be The Most Used Haptic Technology." <https://www.boreas.ca/blogs/events/why-piezo-is-not-the-most-used-haptic-solution-and-why-it-should-be>, June 2017. Accessed: 2019-06-06.
- [66] Flora Wang, "Haptic Energy Consumption," Tech. Rep. SLOA194 – May 2014, Texas Instruments Incorporated, May 2014.
- [67] C. G. PI, "Fundamentals of Piezo Technology." <https://www.piceramic.com/en/piezo-technology/fundamentals/>. Accessed: 2018-12-04.

Bibliography

- [68] C. Hudin and J. Lozada, “Localized Tactile Stimulation by Time-Reversal of Flexural Waves: Case Study With a Thin Sheet of Glass,” in *Proceedings of the World Haptic Conference 2013*, pp. 67–72, 2013.
- [69] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.
- [70] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, vol. 103 of *Springer Texts in Statistics*, Springer New York, New York, NY, 2013.
- [71] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE* **86**, pp. 2278–2324, Nov. 1998.
- [72] F. J. Vera-Olmos and N. Malpica, “Deconvolutional Neural Network for Pupil Detection in Real-World Environments,” in *Biomedical Applications Based on Natural and Artificial Computing*, J. M. Ferrández Vicente, J. R. Álvarez Sánchez, F. de la Paz López, J. Toledo Moreo, and H. Adeli, eds., *Lecture Notes in Computer Science*, pp. 223–231, Springer International Publishing, 2017.
- [73] M. Binkowski, G. Marti, and P. Donnat, “Autoregressive Convolutional Neural Networks for Asynchronous Time Series,” in *Proceedings of the 35th International Conference on Machine Learning*, pp. 580–589, PMLR, July 2018. ISSN: 2640-3498.
- [74] O. Triebe, N. Laptev, and R. Rajagopal, “AR-Net: A simple Auto-Regressive Neural Network for time-series,” Nov. 2019. arXiv:1911.12436 [cs, stat].
- [75] A. v. d. Oord, N. Kalchbrenner, and K. Kavukcuoglu, “Pixel Recurrent Neural Networks,” Aug. 2016. arXiv:1601.06759 [cs].
- [76] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, “Normalizing Flows for Probabilistic Modeling and Inference,” *Journal of Machine Learning Research* **22**, p. 64, 2021.
- [77] D. P. Kingma and M. Welling, “An Introduction to Variational Autoencoders,” *FNT in Machine Learning* **12**(4), pp. 307–392, 2019. arXiv: 1906.02691.
- [78] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative Adversarial Networks,” *arXiv:1406.2661 [cs, stat]*, June 2014. arXiv: 1406.2661.
- [79] I. Goodfellow, “NIPS 2016 Tutorial: Generative Adversarial Networks,” *arXiv:1701.00160 [cs]*, Dec. 2016. arXiv: 1701.00160.
- [80] P. Bontrager, A. Roy, J. Togelius, N. Memon, and A. Ross, “DeepMasterPrints: Generating MasterPrints for Dictionary Attacks via Latent Variable Evolution,” *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pp. 1–9, May 2017.

-
- [81] Y. Gao, R. Singh, and B. Raj, "Voice Impersonation using Generative Adversarial Networks," *arXiv:1802.06840v1*, Feb. 2018.
- [82] C. Donahue, J. McAuley, and M. Puckette, "Adversarial Audio Synthesis," *arXiv:1802.04208 [cs]*, Feb. 2019. arXiv: 1802.04208.
- [83] J. Nistal, S. Lattner, and G. Richard, "DrumGAN: Synthesis of Drum Sounds With Timbral Feature Conditioning Using Generative Adversarial Networks," *arXiv:2008.12073 [cs, eess]*, Aug. 2020. arXiv: 2008.12073.
- [84] J. Nistal, S. Lattner, and G. Richard, "Comparing Representations for Audio Synthesis Using Generative Adversarial Networks," *arXiv:2006.09266 [cs, eess]*, June 2020. arXiv: 2006.09266.
- [85] S. Dieleman, "Generating music in the waveform domain." <https://benanne.github.io/2020/03/24/audio-generation.html>, Mar. 2020. Accessed: 2020-07-17.
- [86] Y. Ujitoko and Y. Ban, "Vibrotactile Signal Generation from Texture Images or Attributes Using Generative Adversarial Network," in *Haptics: Science, Technology, and Applications*, D. Prattichizzo, H. Shinoda, H. Z. Tan, E. Ruffaldi, and A. Frisoli, eds., *Lecture Notes in Computer Science*, pp. 25–36, Springer International Publishing, (Cham), 2018.
- [87] Y. Ujitoko, Y. Ban, and K. Hirota, "GAN-Based Fine-Tuning of Vibrotactile Signals to Render Material Surfaces," *IEEE Access* **8**, pp. 16656–16661, 2020.
- [88] L. Pantera and C. Hudin, "Multitouch Vibrotactile Feedback on a Tactile Screen by the Inverse Filter Technique: Vibration Amplitude and Spatial Resolution," *IEEE Transactions on Haptics*, pp. 1–1, 2020.
- [89] R. K. Ing, N. Quieffin, S. Catheline, and M. Fink, "In solid localization of finger impacts using acoustic time-reversal process," *Appl. Phys. Lett.* **87**, p. 204104, Nov. 2005.
- [90] A. Crevoisier and P. Polotti, "Tangible Acoustic Interfaces and their Applications for the Design of New Musical Instruments," in *Proceedings of the 2005 International Conference on New Interfaces for Musical Expression (NIME05)*, p. 4, (Vancouver, BC, Canada), 2005.
- [91] K. Worden and W. J. Staszewski, "Impact Location and Quantification on a Composite Panel using Neural Networks and a Genetic Algorithm," *Strain* **36**(2), pp. 61–68, 2000.
- [92] R. T. Jones, J. S. Sirkis, E. J. Friebele, and A. D. Kersey, "Location and magnitude of impact detection in composite plates using neural networks," in *Smart Structures and Materials 1995: Smart Sensing, Processing, and Instrumentation*, **2444**, pp. 469–480, International Society for Optics and Photonics, Apr. 1995.
- [93] M. Ghajari, Z. Sharif-Khodaei, and F. M. H. Aliabadi, "Impact Detection Using Artificial Neural Networks," *Key Engineering Materials*, 2012.

- [94] G. Sarego, M. Zaccariotto, and U. Galvanetto, "Artificial neural networks for impact force reconstruction on composite plates and relevant uncertainty propagation," *IEEE Aerospace and Electronic Systems Magazine* **33**, pp. 38–47, Aug. 2018.
- [95] S. J. KIM and J. D. JONES, "Optimal design of piezoactuators for active noise and vibration control," *AIAA Journal* **29**(12), pp. 2047–2053, 1991.
- [96] J. Cuenca, *WAVE MODELS FOR THE FLEXURAL VIBRATIONS OF THIN PLATES - Model of the vibrations of polygonal plates by the image source method - Vibration damping using the acoustic black hole effect*. PhD thesis, Dec. 2009.
- [97] E. K. Dimitriadis, C. R. Fuller, and C. A. Rogers, "Piezoelectric Actuators for Distributed Vibration Excitation of Thin Plates," *J. Vib. Acoust* **113**, pp. 100–107, Jan. 1991.
- [98] S. Wöckel, U. Steinmann, and H. Arndt, "Modelling of time reversal for localized tactile feedback on displays," *Procedia Engineering* **120**, pp. 302–305, Jan. 2015.
- [99] S. Wöckel, U. Steinmann, and H. Arndt, "Haptics by time reversal of elastic waves," in *2016 IEEE International Ultrasonics Symposium (IUS)*, pp. 1–3, Sept. 2016.
- [100] NOCSAE, "Standard pneumatic ram test method and equipment used in evaluating the performance characteristics of protective headgear and face guards." <https://nocsae.org/wp-content/uploads/2018/05/1458844362ND08114m15PneumaticRamTestMethod.pdf>, 2016. Accessed: 2019-06-29.
- [101] J. T. Gwin, J. J. Chu, S. G. Diamond, P. D. Halstead, J. J. Crisco, and R. M. Greenwald, "An Investigation of the NOCSAE Linear Impactor Test Method Based on In Vivo Measures of Head Impact Acceleration in American Football," *J Biomech Eng* **132**, pp. 011006–011006–9, Dec. 2009.
- [102] L. Warnet and P. E. Reed, "Falling Weight Impact Testing Principles," in *Mechanical Properties and Testing of Polymers: An A–Z Reference*, G. M. Swallowe, ed., *Polymer Science and Technology Series*, pp. 66–70, Springer Netherlands, Dordrecht, 1999.
- [103] B. J. Schwarz and M. H. Richardson, "Experimental Modal Analysis," in *in proceedings of CSI Reliability Week*, 1999.
- [104] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python," *J. Mach. Learn. Res.* **12**, pp. 2825–2830, Nov. 2011.
- [105] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16*, pp. 785–794, ACM, (New York, NY, USA), 2016.

-
- [106] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: A System for Large-scale Machine Learning,” in *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation, OSDI’16*, pp. 265–283, USENIX Association, (Berkeley, CA, USA), 2016.
 - [107] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” *arXiv:1412.6980v9*, Dec. 2014.
 - [108] C. H. Mejia, J. Jayet, P. Germano, A. Thabuis, and Y. Perriard, “Linear Impact Generator for Automated Dataset Acquisition of Elastic Waves in Haptic Surfaces,” in *2019 22nd International Conference on Electrical Machines and Systems (ICEMS)*, pp. 1–5, Aug. 2019. ISSN: 2640-7841.
 - [109] C. H. Mejia, P. Germano, S. C. Echeverri, and Y. Perriard, “Artificial Neural Networks for Impact Position Detection in Haptic Surfaces,” in *2019 IEEE International Ultrasonics Symposium (IUS)*, pp. 1874–1877, Oct. 2019. ISSN: 1948-5719.
 - [110] N. Halko, P.-G. Martinsson, and J. A. Tropp, “Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions,” *arXiv:0909.4061 [math]*, Sept. 2009. arXiv: 0909.4061.
 - [111] C. H. Mejia, J. Chavanne, P. Germano, and Y. Perriard, “Effect of the Impact Contact Duration on Machine Learning Models for Impact Position Detection,” in *2020 23rd International Conference on Electrical Machines and Systems (ICEMS)*, pp. 2063–2068, Nov. 2020. ISSN: 2642-5513.
 - [112] T. Oliphant, *NumPy: A guide to NumPy*, USA: Trelgol Publishing, Jan. 2006. Accessed: 2020-07-01.
 - [113] T. M. Mitchell and T. M. Mitchell, *Machine learning*, vol. 1, McGraw-hill New York, 1997.
 - [114] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*, MIT Press, 2018.
 - [115] S. Müller, “Measuring Transfer-Functions and Impulse Responses,” in *Handbook of Signal Processing in Acoustics*, D. Havelock, S. Kuwano, and M. Vorländer, eds., pp. 65–85, Springer, New York, NY, 2008.
 - [116] A. Mosavi, Y. Faghan, P. Ghamisi, P. Duan, S. F. Ardabili, E. Salwana, and S. S. Band, “Comprehensive Review of Deep Reinforcement Learning Methods and Applications in Economics,” *Mathematics* **8**, p. 1640, Oct. 2020.
 - [117] S. So and K. K. Paliwal, “Reconstruction of a Signal from the Real Part of Its Discrete Fourier Transform [Tips and Tricks],” *IEEE Signal Processing Magazine* **35**, pp. 162–174, Mar. 2018.

Bibliography

- [118] M. Mirza and S. Osindero, “Conditional Generative Adversarial Nets,” *arXiv:1411.1784 [cs, stat]*, Nov. 2014. arXiv: 1411.1784.
- [119] Z. Pan, W. Yu, X. Yi, A. Khan, F. Yuan, and Y. Zheng, “Recent Progress on Generative Adversarial Networks (GANs): A Survey,” *IEEE Access* **7**, pp. 36322–36333, 2019.
- [120] C.-C. Hsu, C.-W. Lin, W.-T. Su, and G. Cheung, “SiGAN: Siamese Generative Adversarial Network for Identity-Preserving Face Hallucination,” *IEEE Transactions on Image Processing* **28**, pp. 6225–6236, Dec. 2019.
- [121] B. Liu, L. Wang, J. Wang, and J. Zhang, “Dual Discriminator Weighted Mixture Generative Adversarial Network for image generation,” *J Ambient Intell Human Comput*, Feb. 2022.
- [122] F. Xu, H. Zhang, R. Xiao, Z. Hou, and S. Chen, “Autonomous weld seam tracking under strong noise based on feature-supervised tracker-driven generative adversarial network,” *Journal of Manufacturing Processes* **74**, pp. 151–167, Feb. 2022.
- [123] A. Radford, L. Metz, and S. Chintala, “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks,” *arXiv:1511.06434 [cs]*, Jan. 2016. arXiv: 1511.06434.
- [124] J. Engel, K. K. Agrawal, S. Chen, I. Gulrajani, C. Donahue, and A. Roberts, “GANSYNTH: ADVERSARIAL NEURAL AUDIO SYNTHESIS,” *ArXiv abs/1902.08710*, p. 17, 2019.
- [125] K. Kilgour, M. Zuluaga, D. Roblek, and M. Sharifi, “Fréchet Audio Distance: A Reference-Free Metric for Evaluating Music Enhancement Algorithms,” in *Interspeech 2019*, pp. 2350–2354, ISCA, Sept. 2019.
- [126] M. Bińkowski, D. J. Sutherland, M. Arbel, and A. Gretton, “Demystifying MMD GANs,” in *ICLR 2018*, Feb. 2022.
- [127] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved techniques for training GANs,” in *Proceedings of the 30th International Conference on Neural Information Processing Systems, NIPS’16*, pp. 2234–2242, Curran Associates Inc., (Red Hook, NY, USA), Dec. 2016.
- [128] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, “Improved training of wasserstein GANs,” in *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17*, pp. 5769–5779, Curran Associates Inc., (Red Hook, NY, USA), 2017.
- [129] A. Barahona-Róos and S. Pauletto, “Synthesising knocking sound effects using Conditional WaveGAN,” in *Proceedings of the Sound and Music Computing Conferences, 2020-June*, pp. 450–456, 2020. ISSN: 2518-3672.
- [130] E. L. Denton, S. Chintala, a. szlam, and R. Fergus, “Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks,” in *Advances in Neural Information Processing Systems*, **28**, Curran Associates, Inc., 2015.

-
- [131] A. Brock, J. Donahue, and K. Simonyan, "Large Scale GAN Training for High Fidelity Natural Image Synthesis," *arXiv:1809.11096 [cs, stat]*, Feb. 2019. arXiv: 1809.11096.
 - [132] M. A. Plaisier, L. I. N. Sap, and A. M. L. Kappers, "Perception of vibrotactile distance on the back," *Sci Rep* **10**, p. 17876, Oct. 2020.
 - [133] J. M. Loomis, "Tactile letter recognition under different modes of stimulus presentation," *Perception & Psychophysics* **16**, pp. 401–408, Mar. 1974.
 - [134] S. Saida, Y. Shimizu, and T. Wake, "Computer-Controlled TVSS and Some Characteristics of Vibrotactile Letter Recognition," *Percept Mot Skills* **55**, pp. 651–653, Oct. 1982.
 - [135] S. D. Novich and D. M. Eagleman, "Using space and time to encode vibrotactile information: toward an estimate of the skin's achievable throughput," *Exp Brain Res* **233**, pp. 2777–2788, Oct. 2015.
 - [136] J. Wu, Z. Song, W. Wu, A. Song, and D. Constantinescu, "A vibro-tactile system for image contour display," in *2011 IEEE International Symposium on VR Innovation*, pp. 145–150, Mar. 2011.
 - [137] F. Ono, "The Effect of Ratio of Changing to Static Stimuli on the Attentional Capture," *Sci Rep* **8**, p. 17438, Nov. 2018.
 - [138] I. SanMiguel, D. Linden, and C. Escera, "Attention capture by novel sounds: Distraction versus facilitation," *European Journal of Cognitive Psychology* **22**, pp. 481–515, June 2010.
 - [139] S. J. Bolanowski and J. J. Zwislocki, "Intensity and frequency characteristics of pacinian corpuscles. II. Receptor potentials," *J. Neurophysiol.* **51**, pp. 812–830, Apr. 1984.
 - [140] R. T. Verrillo, "Effect of Contactor Area on the Vibrotactile Threshold," *The Journal of the Acoustical Society of America* **35**, pp. 1962–1966, Dec. 1963.
 - [141] M. J. Cunningham, D. F. L. Jenkins, and M. M. Bakush, "Experimental investigation of optimum thickness of a piezoelectric element for cantilever actuation," *IEE Proceedings - Science, Measurement and Technology* **144**, pp. 45–48, Jan. 1997.
 - [142] A. Derode, A. Tourin, and M. Fink, "Ultrasonic pulse compression with one-bit time reversal through multiple scattering," *Journal of Applied Physics* **85**, pp. 6343–6352, May 1999.
 - [143] C.-H. Nguyen, S. Pietrzko, and R. Buetikofer, "The influence of temperature and bonding thickness on the actuation of a cantilever beam by PZT patches," *Smart Mater. Struct.* **13**, pp. 851–860, June 2004.
 - [144] A. Chaigne and C. Lambourg, "Time-domain simulation of damped impacted plates. I. Theory and experiments," *The Journal of the Acoustical Society of America* **109**, pp. 1422–1432, Apr. 2001.

Bibliography

- [145] R. T. Verrillo, "TEMPORAL SUMMATION IN VIBROTACTILE SENSITIVITY," *J Acoust Soc Am* **37**, pp. 843–846, May 1965.
- [146] G. A. Gescheider and J. M. Joelson, "Vibrotactile temporal summation for threshold and suprathreshold levels of stimulation," *Perception & Psychophysics* **33**, pp. 156–162, Mar. 1983.
- [147] D. M. Green and J. A. Swets, *Signal detection theory and psychophysics*, Signal detection theory and psychophysics, John Wiley, Oxford, England, 1966. Pages: xi, 455.
- [148] N. A. Macmillan and C. D. Creelman, *Detection theory: A user's guide*, Detection theory: A user's guide, Cambridge University Press, New York, NY, US, 1991. Pages: xv, 407.
- [149] "Method of Constant Stimuli," in *Encyclopedia of Neuroscience*, M. D. Binder, N. Hirokawa, and U. Windhorst, eds., pp. 2347–2347, Springer, Berlin, Heidelberg, 2009.

Camilo Hernandez Mejia

Basel, Switzerland

+41 79 382 5016

camilo.hernandez.m@protonmail.ch

Education

- Aug 2018 - Dec 2022 **Ph.D. in Robotics, Control and Intelligent Systems**
Integrated Actuators Laboratory, EPFL, Switzerland
"Deep Learning For Localized-Haptic Feedback in Tactile Surfaces"
Thesis Director: Prof. Yves Perriard
- Jan 2010 - Dec 2014 **Mechatronics Engineering Diploma (5-year program)**
Escuela de Ingeniería de Antioquia (EIA University), Colombia.

Publications

- C. Hernandez-Mejia, X. Ren, A. Thabuis, J. Chavanne, P. Germano, and Y. Perriard, "Conditional GANs for Localized Vibrotactile Feedback in Haptic Surfaces," in *IEEE Transactions on Industry Applications (TIA)*, 2023. *[Under Review]*
- C. Hernandez-Mejia, X. Ren, A. Thabuis, J. Chavanne, P. Germano, and Y. Perriard, "Generative Adversarial Networks for Localized Vibrotactile Feedback in Haptic Surfaces," in *2021 24th International Conference on Electrical Machines and Systems (ICEMS)*, pp. 105–110, Oct. 2021.
- C. Hernandez-Mejia, M. Favier, X. Ren, P. Germano, and Y. Perriard, "Reinforcement Learning and Hardware in the Loop for Localized Vibrotactile Feedback in Haptic Surfaces," in *2021 IEEE International Ultrasonics Symposium (IUS)*, pp. 1–5, Sep. 2021.
- C. Hernandez-Mejia, J. Chavanne, P. Germano, and Y. Perriard, "Effect of the Impact Contact Duration on Machine Learning Models for Impact Position Detection," in *2020 23rd International Conference on Electrical Machines and Systems (ICEMS)*, pp. 2063–2068, Nov. 2020.
- C. H. Mejia, P. Germano, S. C. Echeverri, and Y. Perriard, "Artificial Neural Networks for Impact Position Detection in Haptic Surfaces," in *2019 IEEE International Ultrasonics Symposium (IUS)*, pp. 1874–1877, Oct. 2019.
- C. H. Mejia, J. Jayet, P. Germano, A. Thabuis, and Y. Perriard, "Linear Impact Generator for Automated Dataset Acquisition of Elastic Waves in Haptic Surfaces," in *2019 22nd International Conference on Electrical Machines and Systems (ICEMS)*, Aug. 2019.
- C. Hernandez-Mejia, X. Ren, S. Papetti, H. Jarvelainen, T. Chavdarova, P. Germano, and Y. Perriard, "GANs for Localized Vibrotactile Feedback in Haptic Surfaces: Single Vs Randomly-distributed stimulation", in *IEEE Transactions on Haptics (ToH)*. *[Pending]*

Student Projects Supervised

- 2022 **Development of a finger pressing-force data logger**
Edoardo BOCCHIOTTI, Special Bachelor Project
- 2021 **Time-reversed Impulse Response Haptic Demonstrator - Actuation**
Christopher HÉMON, Semester Project
- 2021 **Time-reversed Impulse Response Haptic Demonstrator - Sensing**
Théophile BALESTRINI, Semester Project
- 2021 **High voltage power supply system for tethered drones - Part II**
Clément BONGINI, Bachelor Project
- 2021 **Design and implementation of an FPGA digital signal generation system**
Andres CHICA, Semester Project
- 2021 **Experiments with reinforcement learning for reversed impact signal generation**
Marc FAVIER, Semester Project
- 2021 **Demonstrator surface for machine learning algorithm for impact Position detection**
Merlin LEUENBERG, Semester Project
- 2020 **High voltage power supply for tethered drones**
Shadi NAGUIB, Semester Project
- 2020 **Electromagnetic Keyboard/Mouse Switch Profile Tuning**
Kenny JEANMONOD, Master Thesis (Co-Supervised: Mr. Jan Stoeckli - Logitech SA)
- 2019 **Development of a CNC surface vibration scanning system**
Jérémy JAYET, Semester Project

Professional Experience

- | | |
|---------------------|---|
| Mar 2018 - Aug 2022 | Volant Aerial Inc. (StartUp USA) - R&D Advisor: Developed an automated tethered balloon and a Deep learning fire-detection system (improved fire-detection accuracy by 10%). |
| Oct 2015 - Dec 2017 | Concreto SA (Colombia) - Innovation Coordinator: Proposed, designed, and built a large-scale concrete 3D printer (First in LA). |
| Jan 2013 - Oct 2015 | Cadena S.A.(Colombia) - Process Engineer: Developed an automatic packaging & mail labeling system (reduced costs by 30%). |

Skills

Python, Deep Learning, LabView, Matlab, C++, Objective C, Latex, Signal Acquisition and Processing, Basic Clean-room Fabrication, 3D CAD, Additive Manufacturing, PCB Design, Sensor Design, Actuator Design, Robotics and Automation, R&D, Project Management.