

Effect of the Impact Contact Duration on Machine Learning Models for Impact Position Detection

Camilo Hernandez Mejia*, Jonathan Chavanne*, Paolo Germano*, Yves Perriard, Senior Member, IEEE* *
Laboratory of Integrated Actuators (LAI), École polytechnique fédérale de Lausanne EPFL, Lausanne, Switzerland
 Contact Email: camilo.hernandez@epfl.ch

Abstract—Touch-screens have become the most common way of interaction between humans and machines. Nevertheless, the lack of rich haptic-feedback on most of the devices limits the quality and effectiveness of the interaction. To develop novel haptic feedback generation strategies, the authors of this work, are initially experimenting with machine learning algorithms to detect the position of a finger over a tactile surface. This paper presents the experimental study to determine the effect of the impact contact duration on the different machine learning (ML) and neural network (NN) models that were previously proposed for impact position detection. A new version of a Linear Impact Generator (LIG) is presented and an experimental study, with a high-speed camera, is carried out to characterize the LIG. Additionally, two different pre-processing methods are compared (i.e. Magnitude Spectrogram representation and FFT frequency-domain representation), showing that the FFT representation contains richer information to describe the impact position. The best model achieved an error (Validation MAE) of 0.18 % or 0.31 mm and (Test MAE) of 1.16 % or 2 mm. Lastly, it was demonstrated that the impact contact duration has a direct effect on the precision of the impact position prediction. When the contact duration changes, the error increases to (Test MAE) 27.12 % or 48 mm on average.

Index Terms—Impact Position Detection, Machine Learning, Artificial Neural Networks, Piezoelectric Transducers, Touch-screen Technology.

I. INTRODUCTION

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 over a tactile screen.

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 and have proven more viable (effective) for larger surfaces; Transforming arbitrarily shaped solid objects into interactive interfaces as shown in [1] or into Digital Musical Instruments (DMI) [2].

A different application for impact position detection, which is highly related, appears in the field of composite materials for aerospace applications, structural health monitoring is crucial for composite structures as they suffer of "barely visible impact damage", thus, knowing the position and intensity of an impact it's crucial to determine the maintenance task to be carried out to preserve the structural integrity.

The use of piezoelectric transducers [4] or displacement sensors [3] 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.* [3], the authors use computer simulations to obtain the frequency response data and strain traces after an impact has occurred, and use this data to train an ANN. 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. Aluminum plates). In [4], experimental impact data is recorded using piezoceramic 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.* present a Genetic Algorithm approach to find the optimal placement for the sensors. In [5], 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.

In [6], the authors of the paper, use different machine learning (ML) methodologies to obtain an alternative way to detect the position of the fingers on the surface of a DMI. Impacts are generated in different known positions of a surface using a Linear Impact Generator (LIG) [7] and the flexural waves are recorded using piezoelectric transducers. These signals are used to train different machine learning models to detect the position where the impact occurred. The authors obtain a Mean Absolute Error (MAE) of 4% during the training stage and the models presented good generalization (MAE of 8% for test data points). Nevertheless, these models were developed using impacts that resemble a perfect impulse, but in reality, when humans interact with a surface the contact duration, between the fingertip and the surface, can vary.

This paper presents the study of the effect that the contact time, during an impact, has on the precision of the impact position detection. Also, some modifications to the LIG and the way that the data is preprocessed bring an important improvement in the train and test error.

Section II, shows some changes on the automated LIG, as well as the signal acquisition and data-set creation process. In addition, this section contains a study, carried out with a high-speed camera, to understand the contact duration during different types of impact.

Then, section III presents an analysis using the Principal Component Analysis (PCA) method to visualize the distribution of the data. Later, section IV contains the training and testing of the models with the new data-sets, including the different results. Finally, section V presents the performance of the previously trained models with data containing different contact times.

II. DATA ACQUISITION

Following the same approach as in [6], the automated LIG presented in [7] was used to generate several impacts over a surface, then piezoelectric transducers allowed to acquire the flexural waves (i.e. time-dependent stress signals) that occur after an impact.

A. Improvement of the LIG and High-speed Camera Analysis of the Contact Duration

The previous version of the LIG was developed using an electromagnetic linear actuator. After an analysis of the displacement profile of the tip of the impactor, it was possible to determine that the impacts were not perfectly repeatable, one could demonstrate that this was due to the hysteresis of the magnetic system. Seeing that, and to control both the impact force and time of contact, a novel version of the impact generator was developed. In this case, a pneumatic actuator was chosen, the functional prototype and the pneumatic system diagram can be seen in figure 1.

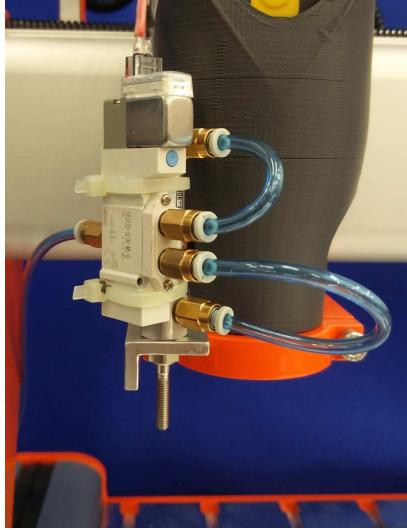


Fig. 1. The new version of the Linear Impact Generator. LEFT, functional prototype LIG V3. RIGHT, Pneumatic system diagram.

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, different accessories can be screwed in the rod of the piston to change the interaction between the tip of the impactor and the surface. To control the force (i.e. the magnitude of the impact) a service unit (FESTO - LFR-D-7-MIDI) acts as a pressure regulator.

An arbitrary signal generator (TG 2512A), which can be remotely controlled via ethernet, creates the signal to control the solenoid valve, enabling the possibility to control the period during which the impactor stays in the lower position. All other components remain the same, as in the previous version of the LIG.

Due to the conservation of momentum, it was expected that the impactor tip and the impacted surface would not remain in constant contact right after the impact. Seeing that, the contact time was validated using a high-speed camera (Photron FASTCAM Mini AX100 recording @ 8400 FPS) as shown in figure 2. It was confirmed that after the impactor hits the surface for the first time, the tip of the LIG bounces, and the surface of the beam goes down and comes back to have a second contact with the impactor, after that they remain in contact. Moreover, using the FPV4 software, one can track a reference point in the moving part and obtain the displacement profile for the tip of the impactor, it is presented in figure 3.

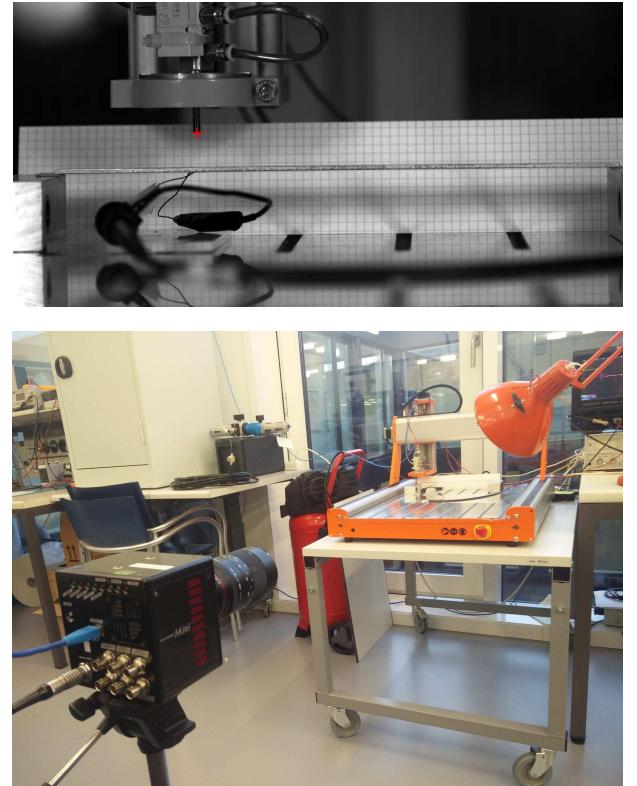


Fig. 2. High-speed camera experimental set-up for the analysis of the displacement profile for the tip of the LIG.

Different type of impacts are presented in figure 3, by changing the length of the square pulse that drives the solenoid valve, it is possible to modify the contact duration. The previous version of the LIG achieved a contact time of approximately 4 ms (Note that due to the hysteresis of the magnetic system, this time varied in the range of ± 1 ms), the new version can go down to a contact duration of 0.24 ms. Also, the signals acquired with the new actuator were proven to be more consistent among several repetitions.

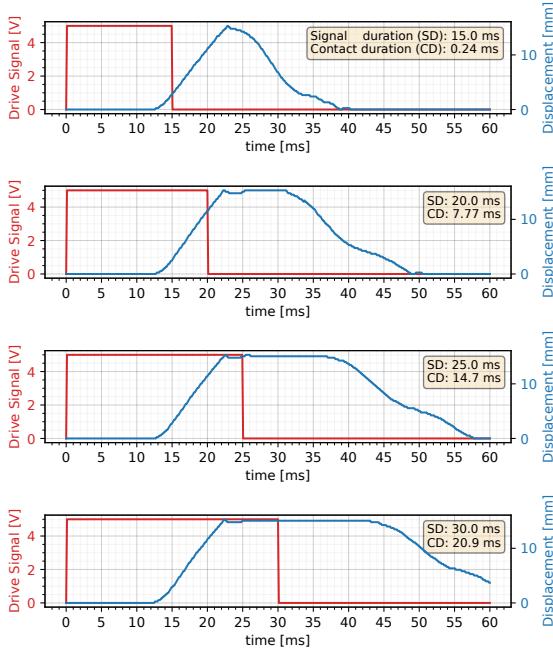


Fig. 3. Displacement profile for the tip of the LIG. The 4 graphs present the behavior of the LIG after different driving pulses are used to actuate the solenoid valve. As a consequence, 4 different types of impacts are created, which have different contact durations as presented on each figure.

B. Acquisition Methodology

A $7 \times 7 \times 0.2 \text{ mm}^3$ piezoceramic transducer (Steminc - SMPL7W8T02412WL) was bonded to a $250 \times 16 \times 2 \text{ mm}^3$ aluminum beam (Aluminum AW-6082) using epoxy glue. The piezoelectric transducer was placed at position $X = 62.5 \text{ mm}$ (i.e. in the length direction) and $Y = 8 \text{ mm}$ (i.e. in the width direction) relative to the edge of the beam.

The novel version of the automated LIG (described in section II-A) is used to acquire the stress signal of 30 individual impacts every 1 mm , starting at position $X = 63 \text{ mm}$ and ending in position $X = 240 \text{ mm}$ (The Y position is kept constant at $Y = 8 \text{ mm}$). The acquisition was distributed in five different data-sets that were acquired at different times of the day to take into account the variation of variables that are not controlled in the experiment (e.g. room temperature). The five data-sets are distributed as follows:

- Data-set # 1: Start $X = 63 \text{ mm}$ steps 5 mm end $X = 238 \text{ mm}$ (e.g. $X = 63, X = 68, \dots, X = 238$).
- Data-set # 2: Start $X = 64 \text{ mm}$ steps 5 mm end $X = 239 \text{ mm}$ (e.g. $X = 64, X = 69, \dots, X = 239$).
- Data-set # 3: Start $X = 65 \text{ mm}$ steps 5 mm end $X = 240 \text{ mm}$ (e.g. $X = 65, X = 70, \dots, X = 240$).
- Data-set # 4: Start $X = 66 \text{ mm}$ steps 5 mm end $X = 236 \text{ mm}$ (e.g. $X = 66, X = 71, \dots, X = 236$).
- Data-set # 5: Start $X = 67 \text{ mm}$ steps 5 mm end $X = 236 \text{ mm}$ (e.g. $X = 67, X = 72, \dots, X = 237$).

The data-sets acquisition is very similar to [6]. The main differences for this study are:

- The acquisition was repeated 3 times for every scenario, to validate the consistency of the error during the training of the different models.

- The signal sampling frequency at the moment of the acquisition was 250 kHz instead of 50 kHz , as a consequence, the recording period was 200 ms instead of 1000 ms . This modification allowed to capture a higher frequency bandwidth and to recover important spectral content that was omitted in [6] (e.g. 42 kHz components, see figure 6). An example of a single acquisition, with the new sampling conditions, is presented in figure 4.

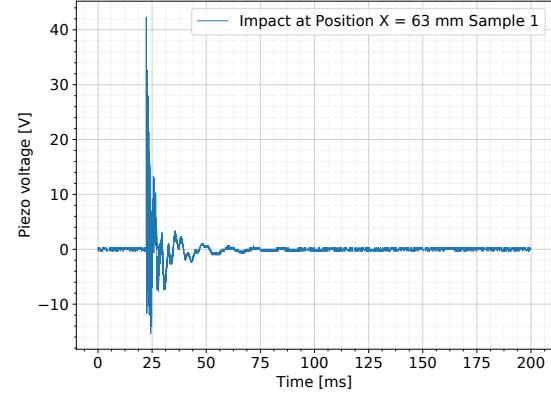


Fig. 4. Example of a single stress signal acquisition, sample # 1/30 for an impact induced at position $X = 63 \text{ mm}$.

To train the models and to test the effect that the duration of the contact, after an impact has occurred, has on the position detection algorithms, four types of impacts are defined (i.e. the displacement profile for each type of impact is presented in figure 3). The reference contact duration was measured as 0.24 ms . This type of impact resembles a perfect impulse and it is obtained when a drive signal of 15 ms is sent into the solenoid valve. The data-set obtained from this type of impact is used to train and test the model as described in section IV.

Furthermore, three additional contact duration (i.e. 8 ms , 15 ms , and 21 ms approximately) are defined. During the acquisition of each group of 5 data-sets, the Contact Duration (CD) was set to each of the different CD values. This data is used to test the effect of the contact duration, as presented in section V.

The whole process is repeated three times, thus a total of $5310 * 3$ impact signals are stored for each contact duration.

C. Data Transformation

The obtained signals are post-processed to obtain the training data. For this study, 2 different transformations are made and their effect on the precision of the impact detection is presented in section IV.

On the one hand, in the same manner as in [6], the recorded signals are transformed into magnitude spectrograms using the MATLAB spectrogram function [8], (i.e. short-time Fourier transform) with a window value of 256, an overlap value of 64, an NFFT value of 256 and the signal sampling frequency. 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 figure 5.

One can find that under specific circumstances the short-time Fourier transform (STFT) can output zero values, that in a logarithmic scale (i.e. the representation on the final result of the spectrogram) tends towards negative infinity, which may cause unexpected problems when input into the ML models. To avoid this, the epsilon of the machine (i.e. the smallest value or smallest delta possible for a specified precision, e.g. 2.2204e-16 for "double-precision" in MATLAB) is added to the result of the STFT.

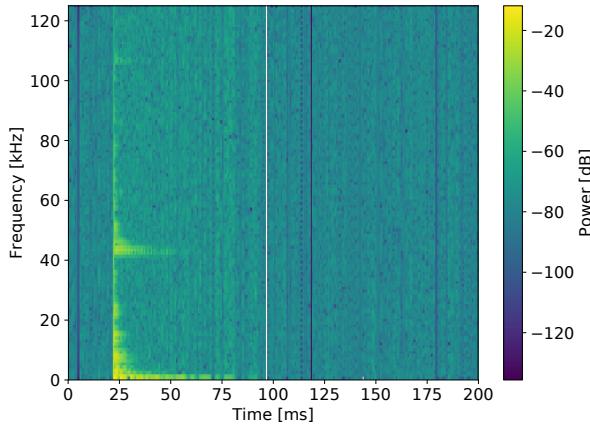


Fig. 5. Example of the magnitude spectrogram transformation for a single acquisition using the MATLAB Spectrogram function, the input was the same signal as the one presented in figure 4.

On the other hand, the signals are transformed into the frequency domain (i.e. the Discrete Fourier Transform (DFT) of the real value signal is obtained) by applying the Fast Fourier Transform (FFT) algorithm. In this case, the Python Numpy rfft module [9] is used, please see figure 6.

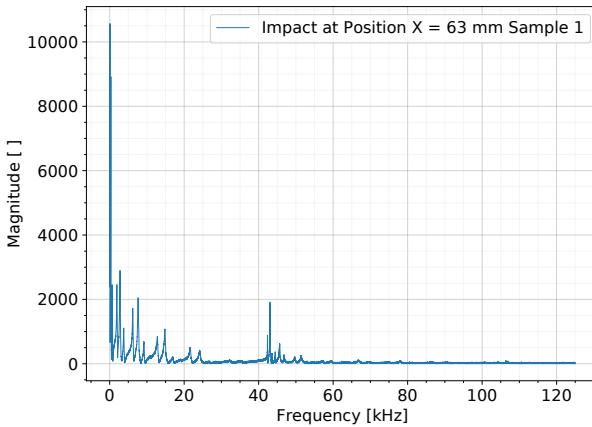


Fig. 6. Example of the frequency domain signal transformation, obtained from a single acquisition, using the FFT Algorithm. This transformation was obtained from the same signal from figure 4.

After obtaining the DFT, the magnitude of the complex number is calculated, for each impact a single column vector of length 25000 is obtained and it is stored in the dataset, labeled with the position where the impact was induced and the sample number (e.g."acquisitionX63mm_Y8mm_sample_1").

III. DATA ANALYSIS

The Principal Components Analysis (PCA) [15], is used as a dimensionality reduction method to get a three-dimensional (3D) visualization of the data-set (i.e. to project the data into the 3 orthogonal components with the highest variance), providing insight on the distribution of the data. In figure 7 it is possible to observe the 3D representation of the Dataset # 1, obtained with the spectrogram transformation. While in figure 8, one can see the 3D representation for the magnitude of the frequency domain transformation of the same impact signals.

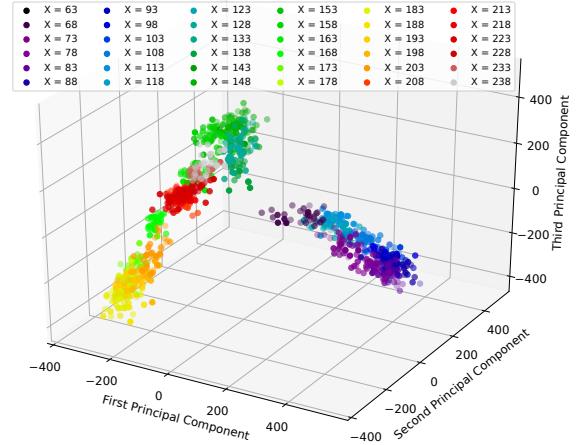


Fig. 7. Three-dimensional graphic representation, using PCA, for dataset # 1 containing the spectrogram transformation, each set of acquisitions at a specific position is represented by one unique color.

As expected, the spectrogram datasets (displayed in figure 7) created with the impact signals that were obtained with the new version of the LIG, present more clustered data that seems easier to separate or to classify. What's more, the frequency-domain transformation (presented in figure 8) of the time-domain impact signals appears to contain richer information for the impact position detection, this is why this data transformation leads to an even more separated representation in the 3D PCA dimensional reduction. The impact that these improvements in both datasets have on the machine learning algorithms is presented in section IV.

IV. MODELS TRAINING AND RESULTS

In the same way as in [6], the prediction task was treated as a regression problem. To predict the position where the impact has occurred, 4 different ML models were trained and evaluated. The Linear regression model [10], the XGboost model (i.e. XGB-Regressor) [11], an Artificial Neural Network (i.e. fully connected ANN), and a 2D Convolutional Neural Network (CNN) [12].

For the spectrogram dataset, the 4 models were trained. Whereas for the frequency domain data-set, the 2D CNN was not used due to the structure of the data (i.e. the FFT is a single vector as opposed to the 2D nature of spectrograms).

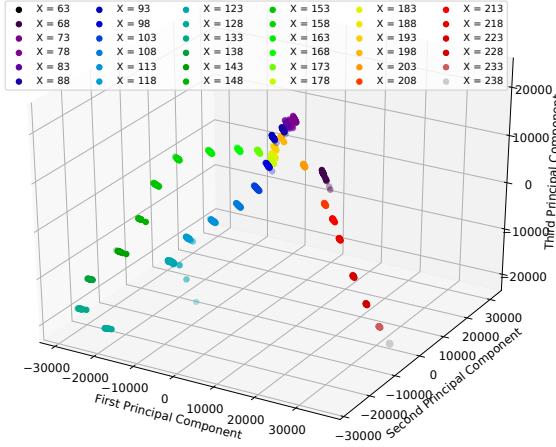


Fig. 8. Three-dimensional graphic representation, of the PCA dimensionality reduction, for dataset # 1 containing the magnitude of the frequency domain transformation, each set of acquisitions at a specific position is represented by one unique color.

A. Data-set Partition and Preparation

The data is separated into two main groups, train + validation set, and test set. The first group, contains the data to train the model and to validate after each iteration, it was obtained from the data-sets # 1, # 2, and # 3, which were merged and then randomly shuffled and split (i.e. 80% is used for training and 20% is used for validation). This split was carried out using the *"train_test_split"* function from the Scikit-learn [10] module. To get reproducible results, during the random shuffle of the data, the *"random_state"* variable was chosen to be 1992. The second group is used to evaluate how well how well the models perform with impact data that have never been observed by the models before (i.e. test the generalization of the model). For this purpose, the data-set # 4 and # 5 are used independently. It is important to mention that before training the 2D CNN with the spectrogram data, it is necessary to reshape the data-set columns to its original size (128 x 260).

B. ML Models

The models were implemented in a Jupyter notebook [13]. First, the linear regression and XGB-Regressor models are fit using the training portion of the data, then, the trained model is used to predict the results for the validation data. The Mean Absolute Error (MAE, see the equation below) is used to measure the error between the real position (i.e. known data-set labels - y) and the predicted value (\hat{y}).

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

After the models are fit (i.e. trained), the two independent test data-sets are used to check the generalization of both models. Later, the same two neural network architectures that were described in [6] where implemented and trained, the sole difference is the input size (32280 x 1 instead of 29952 for the NN and 128 x 260 instead of 128 x 234 for the 2D CNN).

Once again, both models have the MAE as the loss function and make use of the Adam optimizer [14] to update the network weights during the training phase. For the NN models, after each iteration of the learning process, the validation dataset is used to evaluate the progress of the training, if the Validation MAE improves from previous iterations, the weights for the model are stored in a temporal variable. In the same manner, after the models are trained, the two independent test data-sets are used to check if the model is generalizing to data that was never seen before.

C. Results

As a reference, one can find the results from previous studies [6] in the upper part of table I. Then, the results for the 4 models trained with the spectrogram data-sets, are presented in the lower part of table I. The results for the models trained with the frequency-domain (FFT) data-set appear in table II. For all the results, the error term refers to the MAE in mm (also known as the $L1$ -norm loss), while the percentage (%) refers to the ratio between the error and the total length of the studied area (i.e. 177 mm).

TABLE I

Previous Study Results LIG V1 (As in [6]) - Spectrogram Data-set				
Error	LinearReg	XGBoost	NN	CNN 2D
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%

Current Study Results LIG V3 - Spectrogram Data-set				
Error	LinearReg	XGBoost	NN	CNN 2D
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%

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

TABLE II

Current Study Results LIG V3 - Frequency-domain (FFT) Data-set				
Error	LinearReg	XGBoost	NN	CNN 2D
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

Lastly, the frequency-domain data transformation led to a better prediction of the position. This was already expected after the PCA 2D visualization of the data. It appears that the simpler ML models (XGB-Regressor and Linear Regression) performed better with this transformation of the data. However, the three evaluated models achieve 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).

The precision improvement (compared to previous studies [6]), can be associated with the shorter contact duration achieved by the new version of the LIG (i.e. the impact is closer to a perfect impulse, then, a larger number of frequencies are excited). The excited frequencies can be related to the natural harmonics of the plate, and during the different experiments, it was evidenced that for every position the excited frequencies are the same, but the magnitude of each frequency changes. This could be a strong justification for the FFT transformation performing better than the spectrogram transformation.

V. ML MODELS TEST FOR IMPACT DATA WITH DIFFERENT CONTACT DURATION

To test the effect of the contact duration, three additional spectrogram data-sets were obtained (as presented in section II-B and figure 3). The ML models with the best results were used to test the effect of different contact durations in the impact position detection accuracy. In this case, the three groups of five data-sets with different contact times were fed to the models, and the predicted position values were compared with the original labels to obtain the MAE for each test set, later the average MAE and the percentage, were estimated for each contact duration. In table III it is possible to observe the results for each data-set when the contact duration was set to 7.77 ms, then for the other contact duration, as values are very similar, only the average value on the 5 data-sets is presented.

TABLE III

Test of the Contact Duration (CD) effect - Spectrogram Data-set			
Error - CD [ms]	XGBoost	NN	CNN 2D
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 the contact duration 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 in the plate and affecting the natural frequencies that are excited.

VI. CONCLUSIONS

In general terms, this study presents the effect of the contact duration on the precision of the impact position detection when using different ML models to extract this information from the stress waves recorded using piezoelectric transducers.

This publication presented a new version of the LIG that can create more consistent impulse-like impacts and enables the control of the contact duration, the performance of the LIG was validated using a high-speed camera.

Moreover, two different data pre-processing methods were described and evaluated. It was found that even though the FFT 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. However, with both representations, the impact position detection precision improved four times (4 \times) compared to previous studies.

Finally, this study showed the effect of a different contact duration on the precision of different ML algorithms. In the future, the duration of the contact can be used as an input parameter for the models, so that the error can be improved and the models become more general to these particular cases.

ACKNOWLEDGMENT

The authors of this paper would like to thank Sebastian Correa Echeverri - Artificial Intelligence Engineer at Veridas and Sean Thomas - Ph.D. Student at EPFL - LAI for their insight and contributions during the development of this publication.

REFERENCES

- [1] R. K. Ing, N. Quieffin, S. Catheline, and M. Fink, "In solid localization of finger impacts using acoustic time-reversal process," *Appl. Phys. Lett.*, vol. 87, no. 20, p. 204104, 2005.
- [2] A. Crevoisier and P. Polotti, "Tangible Acoustic Interfaces and their Applications for the Design of New Musical Instruments," p. 4, 2005.
- [3] 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*, vol. 2444, pp. 469–480, 1995.
- [4] K. Worden and W. J. Staszewski, "Impact Location and Quantification on a Composite Panel using Neural Networks and a Genetic Algorithm," *Strain*, vol. 36, no. 2, pp. 61–68, 2000.
- [5] 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*, vol. 33, no. 8, pp. 38–47, 2018.
- [6] C. Hernandez, 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.
- [7] C. Hernandez, 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)*, 2019.
- [8] MathWorks, "Spectrogram using short-time Fourier transform - MATLAB spectrogram - MathWorks Switzerland".
- [9] T. Oliphant, *NumPy: A guide to NumPy*. USA: Trelgol Publishing, 2006.
- [10] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python", *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.
- [11] 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*, pp. 785–794, 2016.
- [12] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.
- [13] T. Kluyver et al., "Jupyter Notebooks – a publishing format for reproducible computational workflows," in *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pp. 87–90, 2016.
- [14] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization", 2014.
- [15] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, vol. 103. New York, NY: Springer New York, 2013.