Dataset for AI-assisted classification of damaged saw blades on a miter saw using microphone, accelerometer and ultrasonic data

Marcel Jongmanns[1] (marcel.jongmanns@ipms.fraunhofer.de), Dushyanth Kumar Pingali[1]

[1]Fraunhofer IPMS, Dresden, Germany

Licensing

Abstract

A miter saw was equipped with a sensor node consisting of a wideband MEMS microphone, an accelerometer and an ultrasound transducer. Data has been collected for 4 different saw blades: two different sharp blades, one blunt and one demolished. Reference data was measured when the saw was off and when the saw was actively rotating, but not inside wood. The sawing data, when wood is cut, consists of one dataset where narrow wooden laths were cut and one where wide wooden laths were cut.

Data origin

The data is part of the project WA.VE funded by the Federal Ministry of Education and Research of Germany (BMBF) under grant number 02P21K032. The motivation of the project was to test active ultrasonic measurements to gain additional information about vibrations compared to measurements using microphones and accelerometers.

Measurement setup

A sensor node including a wideband MEMS microphone (Knowles Lazarus), an accelerometer (Analog Devices ADXL1005), and a Fraunhofer IPMS LCMUT ultrasound transmitter[1]. The node was attached to a miter saw as close as possible to the saw blade without interfering with the integrity or safety of the saw (Figure 1).

The sensor data were sampled sequentially at a sample rate of 250kHz over a time frame of 20ms. First 20ms of acoustics were measured using the microphone. Then 20ms of vibrations were measured using the accelerometer. Then a constant 30kHz ultrasound signal was generated and

[1] J.M. Monsalve, A. Melnikov, M. Stolz, A. Mrosk, M. Jongmanns, F. Wall, S. Langa, I. Marica-Bercu, T. Brändel, M. Kircher, H.A.G. Schenk, B. Kaiser, H. Schenk, "Proof of concept of an air-coupled electrostatic ultrasonic transducer based on lateral motion", Sensors and Actuators A: Physical, Vol. 345, 2022, DOI: 10.1016/j.sna.2022.113813

another 20ms of acoustic data were acquired using the microphone. The ultrasound transmitter was deactivated afterwards. The data from the ultrasound measurement was filtered using a bandpass filter with cut off frequencies of 29kHz and 31kHz to remove anything but the echo of the transmitted signal.

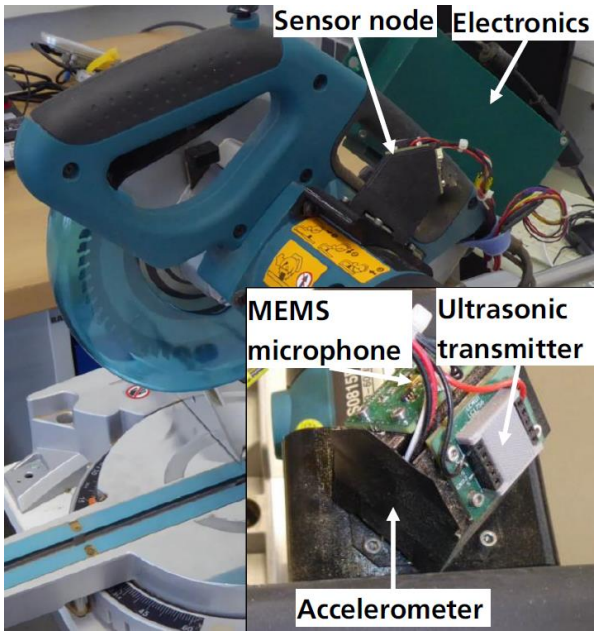

Figure 1: Miter saw equipped with the sensor node on the housing of the motor.

Measurement process

The measurement consists of 3 measurement conditions:

1. Background: The saw is not powered. In some data series the saw is inside the lab, in others outside. In some the saw is not touched, in others the handle is moved up and down like how it is done during normal usage of the miter saw. The goal is to provide a wide range of reference data of the system when nothing of interest is happening.
2. Idle: Comparable to the background measurements, but now the saw is powered and rotating in air. The process has been repeated for each of the 4 different blades.
3. Wood: This group contains measurements where wood is cut. It is divided into 8 sub-groups, 2 for each saw blade. In the first pass small, narrow wooden laths were cut. During the second pass, denoted by '_wide' next to the blade, the wooden laths were bigger than in the first pass.

To interpret the data from the third group it is important to understand the sequence of actions as the data sequence is segmented. First the data acquisition is started. Then the saw is activated, and then a cut is made. To make the next cut the wood needs to be moved while the blade is rotating in air. This results in an alternating pattern between 'blade is in air' and 'blade is in wood'. At the end of the process the data acquisition is usually stopped before the saw blade comes to a complete stop. First, the saw is off. This can be determined by comparing the data to the background data. Then the state is alternating between 'the saw is active, but in air' (Idle) and 'the saw is active and cutting wood' (not yet defined by reference data). This complex data can be segmented either by automatic approaches or manually by comparing it to the known reference data.

The following saw blades were used:

1. sharp36: A new blade with 36 sharp teeth
2. sharp60: A new blade with 60 sharp teeth
3. worn: A blade with 36 worn teeth
4. demolished: A blade with 36 teeth, some of which were damages with a hammer

Data format

The data is provided in the HDF5 format. The following code block shows an example of how to explore the data in Python:

```
import numpy as np
import h5py

hd5 = h5py.File('wave_miter_saw_data.hd5', 'r')

# print data structure
print(list(hd5.keys()))

print(' --- ')
print('Background:')
print(list(hd5['background'].keys()))

print(' --- ')
print('Idle:')
print(list(hd5['idle'].keys()))
for a in list(hd5['idle'].keys()):
    print('idle/' + a + ' (' + str(len(hd5['idle/' + a].keys())) + ')')
    print(list(hd5['idle/' + a].keys()))

print(' --- ')
print('Wood:')
print(list(hd5['wood'].keys()))
for a in list(hd5['wood'].keys()):
    print('wood/' + a + ' (' + str(len(hd5['wood/' + a].keys())) + ')')
    print(list(hd5['wood/' + a].keys()))

# extract the data of one dataset
mic_data = np.array(hd5['wood/worn/10/microphone'])
acc_data = np.array(hd5['wood/worn/10/accelerometer'])
us_data = np.array(hd5['wood/worn/10/ultrasound'])

print('mic data shape: ' + str(np.shape(mic_data)))
print('acc data shape: ' + str(np.shape(acc_data)))
print('us data shape: ' + str(np.shape(us_data)))

hd5.close()
```

The naming convention of the keys is: <measurement condition>/<type of saw blade>/<measurement series as consecutive number>/<sensor>

<measurement condition> can be: ['background', 'idle', 'wood'].

<type of saw blade> depends on the condition. *Background* is not divided by blade, so this key does not apply and must be omitted. For *Idle* it can be ['demolished', 'sharp36', 'sharp60', 'worn'] and for wood there are ['demolished', 'demolished_wide', 'sharp36', 'sharp36_wide', 'sharp60', 'sharp60_wide', 'worn', 'worn_wide'] including different lath sizes.

<measurement series as consecutive number> depends again on the measurement. As the numbers are consecutive it can be looped from 0 to the length of the corresponding keys.

<sensor> can be ['microphone', 'accelerometer', 'ultrasound'].

After extracting the data as numpy array, it has for example the shape (268, 5000). 268 is the number of measurements in the series and 5000 is the length of the time series sampled from the corresponding sensor.