Monitoring an ephemeral stream with a Teensy 3.2 + audio shield to determine water level only from the noise of a stream

Linus Fässler[1], Natalie Ceperley[2], Peter Leiser[1], and Bettina Schaefli[2]
[1]University of Bern, Physical Geography, Hydrology, Switzerland (linus.faessler@giub.unibe.ch)
[2]Institute of Geography (GIUB) and Oeschger Center of Climate Change Research (OCCR), University of Bern, Bern Switzerland

CONTEXT

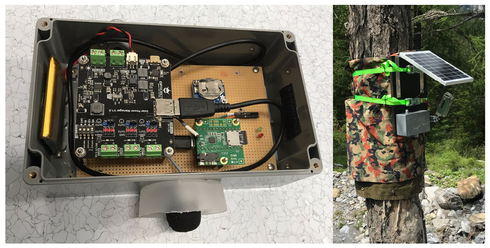
Study site: Vallon de Nant (Vaud, CH)
1497.7 meters above sea level

Intermittent and ephemeral streams (IRES) need new measuring technique

- Emerging subject of research in hydrology
- > 50% of the streams globally
- Lack of spatial and temporal data
- Current tool for water level detection = instream guage
- Harsh conditions = sensor destruction

DIY AUDIO SENSOR

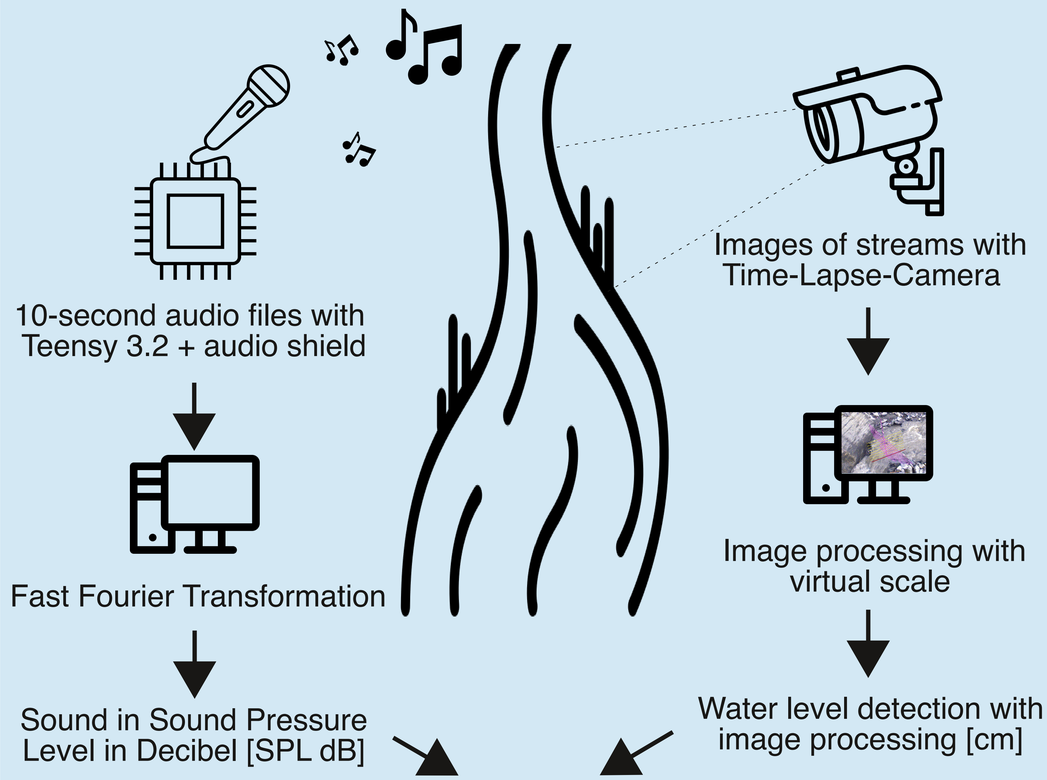
- Distance (<5m from stream)
- Calibration with images
- Monitor by day and night (24h) and powered with solar panel
- Self-made and cost-efficient
- Arduino code for Teensy 3.2 + audio shield

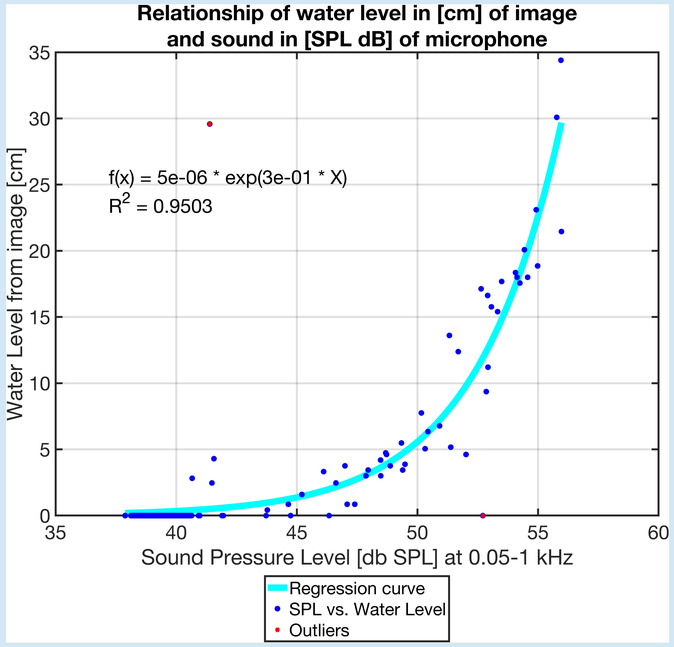


3 main components of audio sensor

1. Power supply, with voltage converter, solar panel, and lithium-polymer (LiPo) battery
2. Microcontroller with real-time clock
3. Audio shield with microphone

METHOD





RESULTS

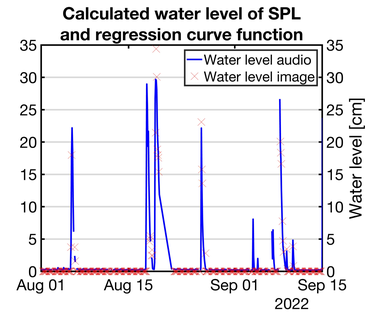
Sensor

- > 95% of the audio files were valid
- Autnomous power supply without problems for 2 months
- Cost: ≈ CHF 155.00

⚠ 2 unidentified shutdowns

Audio and image method

1. f(x) = 5e-0.06 • exp(3e-01 • x)
2. Use SPL dB values as x to get Water level in cm



Observation period: 01.08.23 - 15.09.23
7 stream flow events monitored whereas 2 happend at night
⚠ Image processing can not confirm stream flow events at night

CONCLUSION

Successful **determination of the water level** with audio files

Strong relationship between water level of images and SPL dB of audio

Requires **calibration** by another sensor

Cost-effective sensor to counteract lack of spatial and temporal data

Code **improvement required** for more stability

ACKNOWLEDGEMENT

The code for Fast Fourier Transformation is designed and developed by WM. Alexandre Osborne (Ph.D. thesis, Durham University).

All codes and corresponding thesis are available on Zenodo online available