POLITECHNIKA WROCŁAWSKA
WYDZIAŁ ELEKTRONIKI

KIERUNEK: Elektronika (EKA)
SPECJALNOŚĆ: Inżynieria akustyczna (EIA)

PRACA DYPLOMOWA INŻYNIERSKA

Narzędzie do detekcji i ekstrakcji zdarzeń akustycznych w materiale audio

A tool for detecting and extracting acoustic events in an audio material

AUTOR:
Konrad Kamil Janowski

PROWADZĄCY PRACĘ:

dr Maciej Walczyński, Katedra Akustyki i Multimediów (W4/K5)

OCENA PRACY:

WROCŁAW 2018

Moją pracę chciałbym zadedykować moim rodzicom, którzy zawsze mnie wspierali i pozwolili mi rozwinąć się w kierunku, który pragnąłem. Bez was ta praca byłaby niemożliwa. Dziękuję.

Spis treści

Rozdział 1

Wstęp

1.1 Cel i zakres pracy

Przedmiotem pracy jest oprogramowanie do detekcji i ekstrakcji zdarzeń akustycznych w materiale audio. Oprogramowanie zostało stworzone do obsługi plików w formacie wave o rozdzielczości bitowej szesnaście(16) i dwadzieścia cztery(24) bity oraz częstotliwości próbkowania czterdzieści cztery i sto dziesiątych (44,1) kilo Herca i czterdzieści osiem (48) kilo Herca. Oprogramowanie zostało przetestowane używając plików o tych parametrach. Oprogramowanie zostało stworzone modularnie aby możliwy był jego dalszy rozwój. Z powodu tego założenia, program ma możliwość obsługi dowolnej częstotliwości próbkowania oraz dowolnej rozdzielczości bitowej choć nie jest to zalecane. Praca zawiera cztery moduły:

1. Moduł odczytywania pliku,
2. Moduł przetwarzania sygnału,
3. Moduł detekcji zdarzeń,
4. Moduł zapisu wyników do pliku.

Autor ma [Foundation, 2018] nadzieję, że zaprojektowany w ten sposób system pozwoli na późniejszy rozwój i ułatwi powtórne wykorzystywanie tego narzędzia.

1.2 Motywacja

Analiza nagrań odgrywa bardzo ważną rolę we współczesnych zastosowaniach akustyki. Jednym z przykładów może być ochrona ludzi przed hałasem. Aby zbadać hałas na stanowisku pracy lub ocenić narażenie na hałas przestrzeni publicznej niejednokrotnie wykonuje się bardzo długie nagrania, trwające nawet kilkanaście godzin. Ze względu na metody analizy nieraz niezbędne jest wyselekcjonowanie z nagrania konkretnych zdarzeń akustycznych i analiza ich w izolacji, nie biorąc pod uwagę poziomu tła w pozostałym czasie.

Z drugiej strony, możemy wyobrazić sobie potrzebę wyizolowania z nagrania bardzo wielu zdarzeń akustycznych, które wydarzają się w krótkim odstępie czasu. Jako przykład może tutaj posłużyć strzelnica, gdzie chcąc policzyć liczbę strzałów w ciągu dnia możemy zrobić nagranie krótkie i na podstawie tego krótkiego wycinka czasu oszacować hałas podczas całego dnia. Innym razem możemy mieć potrzebę wykryć te zdarzenia to celów

z pozoru nie związanych z akustyką. Gdyby ktoś chciał policzyć ile razy została odbita piłka do koszykówki podczas badania jej wytrzymałości, jednym z możliwych sposobów na zliczenie ilości odbić mogłoby być policzenie zdarzeń akustycznych jakimi są odbicia piłki od ziemi. Być może ktoś ze względów medycznych chciałby sprawdzić, ile czasu w ciągu jego snu zajmuje chrapanie i w ten sposób ocenić jak duży problem ma z tą dolegliwością. Również w tym przypadku, przy pewnej kontroli hałasu środowiska mógłby zrobić to za pomocą stworzonego narzędzia. Jak widać, wyszukiwanie zdarzeń akustycznych wraz z czasem ich trwania może być niezwykle użyteczne w wielu z pozoru nie związanych z tym dziedzinach.

Biorąc pod uwagę, że przytoczone wyżej przykłady są jedynie wąskim wycinkiem zastosowań, które mogłyby przyjść do głowy komuś, kto ma widzę i zainteresowanie w innej dziedzinie niż autor pracy nie ulega wątpliwości, że warto posiadać takie oprogramowanie. Oczywiście wszystkie przytoczone sytuacje można analizować ręcznie, odsłuchując nagrane próbki. Analiza takich nagrań jednak pochłania dużo czasu i może być problematyczna. Z pomocą stworzonego narzędzia można proces zupełnie lub częściowo zautomatyzować co w obu przypadkach skutkuje znaczną oszczędnością czasu.

Praca powstała w połowie z chęci stworzenia użytecznego narzędzia a w połowie z chęci Autora do poznania współczesnych metod analizy cyfrowych sygnałów fonicznych. Rozwinięcie wiedzy w zakresie tworzenia oprogramowania w języki Python, pracy z systemem kontroli wersji oraz implementacja algorytmów znanych z książek do realnie działającego programu jest procesem, który autor chciał poznać od podszewki. Przedstawienie wyników w formie zrozumiałej, czytelnej i praktycznej jest nieraz jeszcze większym wyzwaniem i nie można nabyć w tym wprawy inaczej, niż pracując z tymi wynikami i samemu przekonać się na ile są one użyteczne.

Powyższe przesłanki zadecydowały o stworzeniu narzędzia.

```
def _filter_db_samples_with_time_constant(self, samples):
    """Take dynamic pressure samples and integrate it with time constant defined in IEC-61672-2013.
    Interact which command line do take time constant to use. Allowed constants are "slow" or "fast".

    Parameters
    ----------
        samples: [float]
        list of samples with dynamic pressure level.

    Returns
    -------
        time_weighted_samples: [float]
        list of samples weighted which defined time constant.

    """
    if self.time_weighting == 'slow':
        time_weighted_samples = standards.iec_61672_1_2013.slow(np.array(samples),
                                    self.wave_reader_object.frame_rate)
        print("Slow constant is applied")
    elif self.time_weighting == 'fast':
        time_weighted_samples = standards.iec_61672_1_2013.fast(np.array(samples),
                                    self.wave_reader_object.frame_rate)
        print("Fast constant is applied")
    else:
        raise ValueError('time weighting must be "slow" or "fast", not {}'.format(self.time_weighting))

    print('{} samples has been converted to {} samples with {} time constant'.format(len(samples),
                                            len(time_weighted_samples),
                                            self.time_weighting))
    print('')
    time_weighted_samples = list(time_weighted_samples)
    return time_weighted_samples
```

Rozdział 2

Wprowadzenie

2.1 Zdarzenie akustyczne

2.2 Przyjęta definicja zdarzenia akustycznego

2.3 Metoda wykrywania zakłóceń

2.4 Istniejące rozwiązania do wykrywania zdarzeń akustycznych

Rozdział 3

Metodologia

3.1 System kontroli wersji - Git, GitHub, GitKraken

3.2 Język programowania - Python

3.3 Struktura programu - Python Package

3.4 Środowisko programistyczne - Pycharm

Rozdział 4

Struktura oraz działanie programu

Rozdział 5

Notatki

5.1 PN-ISO-1996-1:2006 zdarzenie akustyczne

5.1.2 Należy podawać czas trwania zdarzenia w odniesieniu do pewnej cechy dźwięku, jak np, liczba przekroczeń pewnego ustalonego poziomu. PRZYKŁAD czas trwania zdarzenia można zdefiniować jako całkowity czas, w którym poziom ciśnienia akustycznego mieści się w zakresie 10 dB maksymalnego poziomu ciśnienia akustycznego podczas zdarzenia.

Dodatek A

Kod Programu

Bibliografia

[Foundation, 2018] Foundation, P. S. (2018). Python 3.7.1 documentation, data structures. http://web.archive.org/web/20080207010024/http://www.808multimedia.com/winnt/kernel.htm. Accessed: 2010-09-30.