

РАЗРАБОТКА АЛГОРИТМА АВТОМАТИЗАЦИИ ГРАДУИРОВКИ УЛЬТРАЗВУКОВОГО ТОЛЩИНОМЕРА

Выполнил: Кудюров В.В.
Руководитель: Журкин А.М.

Цель разработки алгоритма:

- вычислить время между сигналом возбуждения и сигналом эхо для ультразвукового преобразователя.

Задачи данной работы:

1. Определить времена сигналов;
2. Определить верный уровень сигналов;

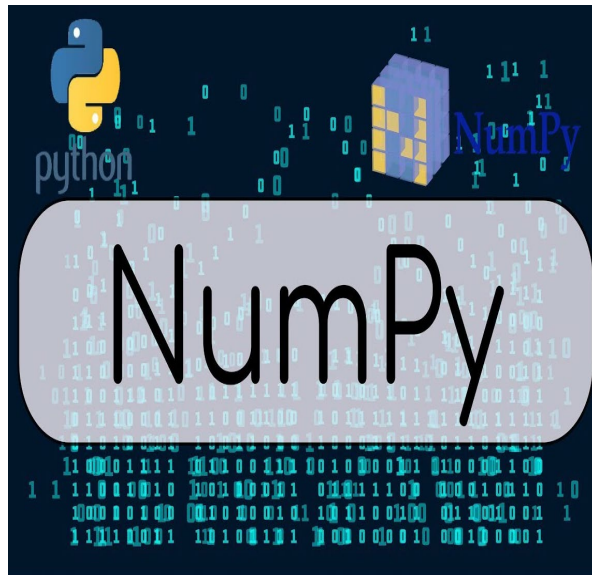
Ультразвуковой толщиномер – предназначен для измерения толщины изделий изготовленных из материалов с затуханием ультразвуковых колебаний.

Градуировка ультразвукового преобразователя является трудоемкой процедурой.

Задачей предварительной обработки сигнала является подавление шума и помехи.



Импортированные библиотеки



Python Math Module

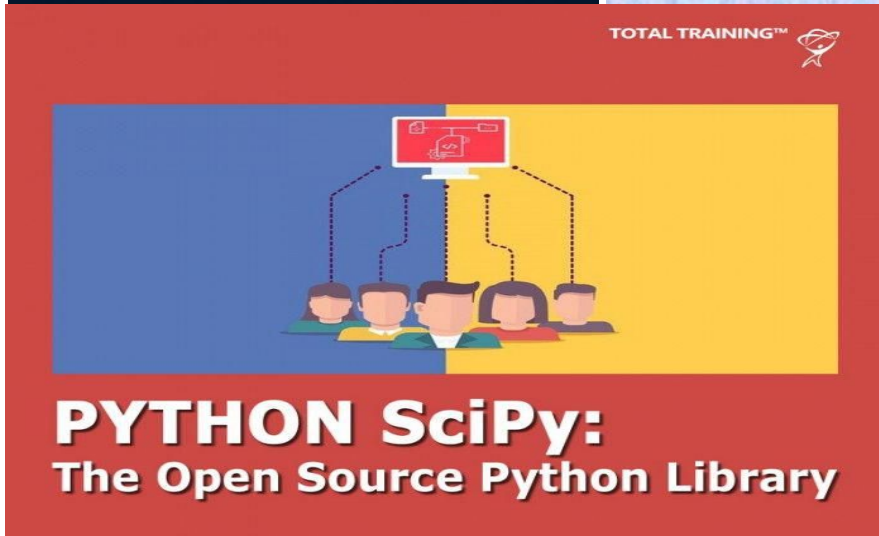


PyWavelets 1.1.1

```
pip install PyWavelets
```



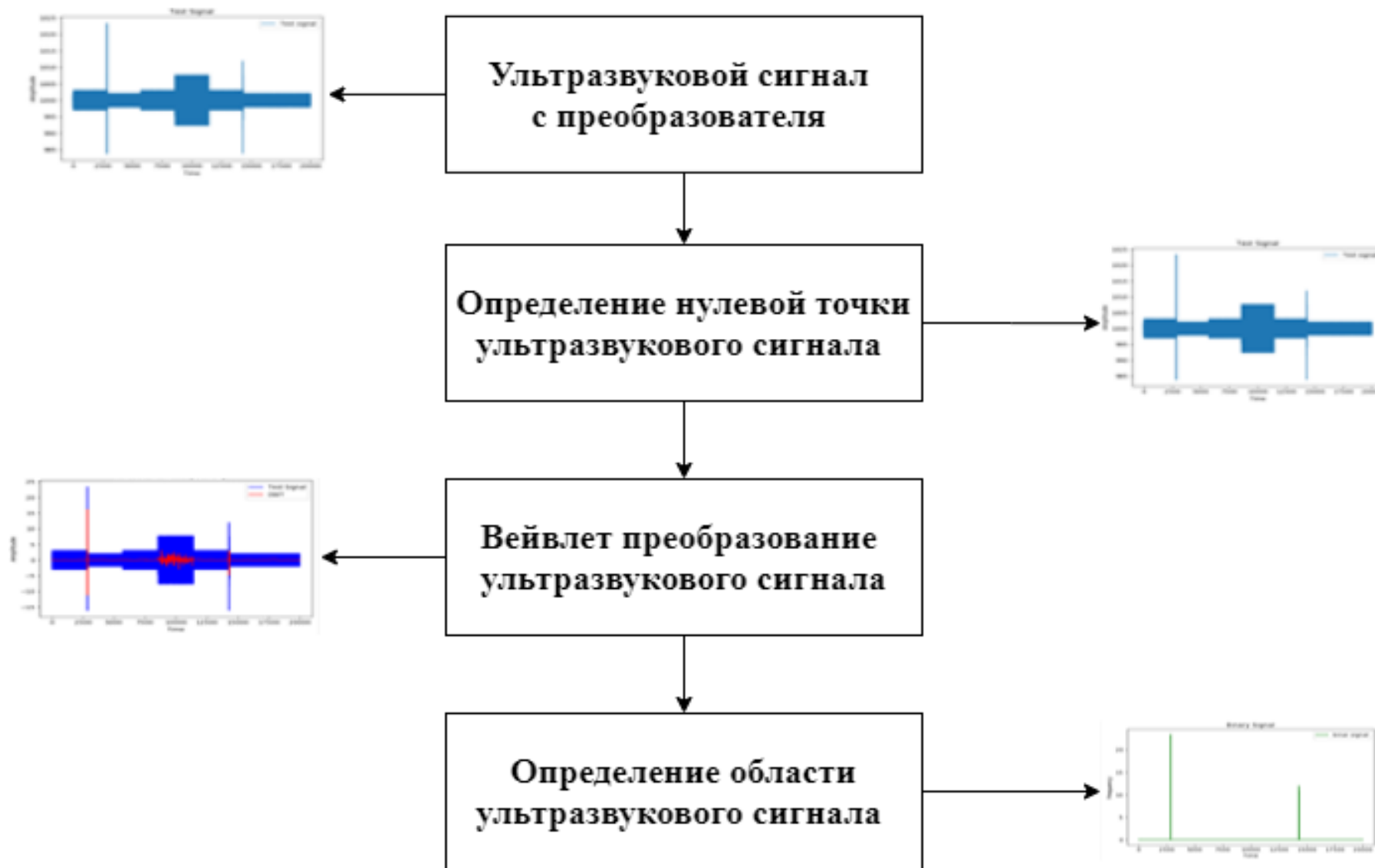
Copy to clipboard

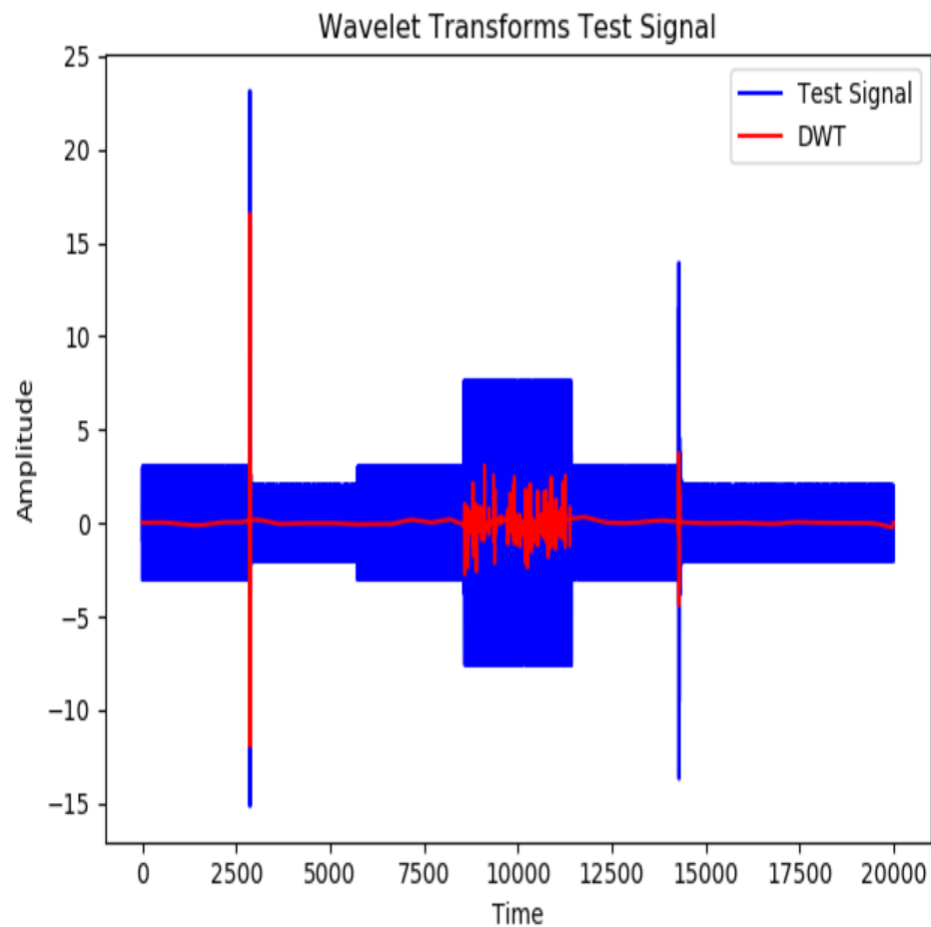


python
Matplotlib

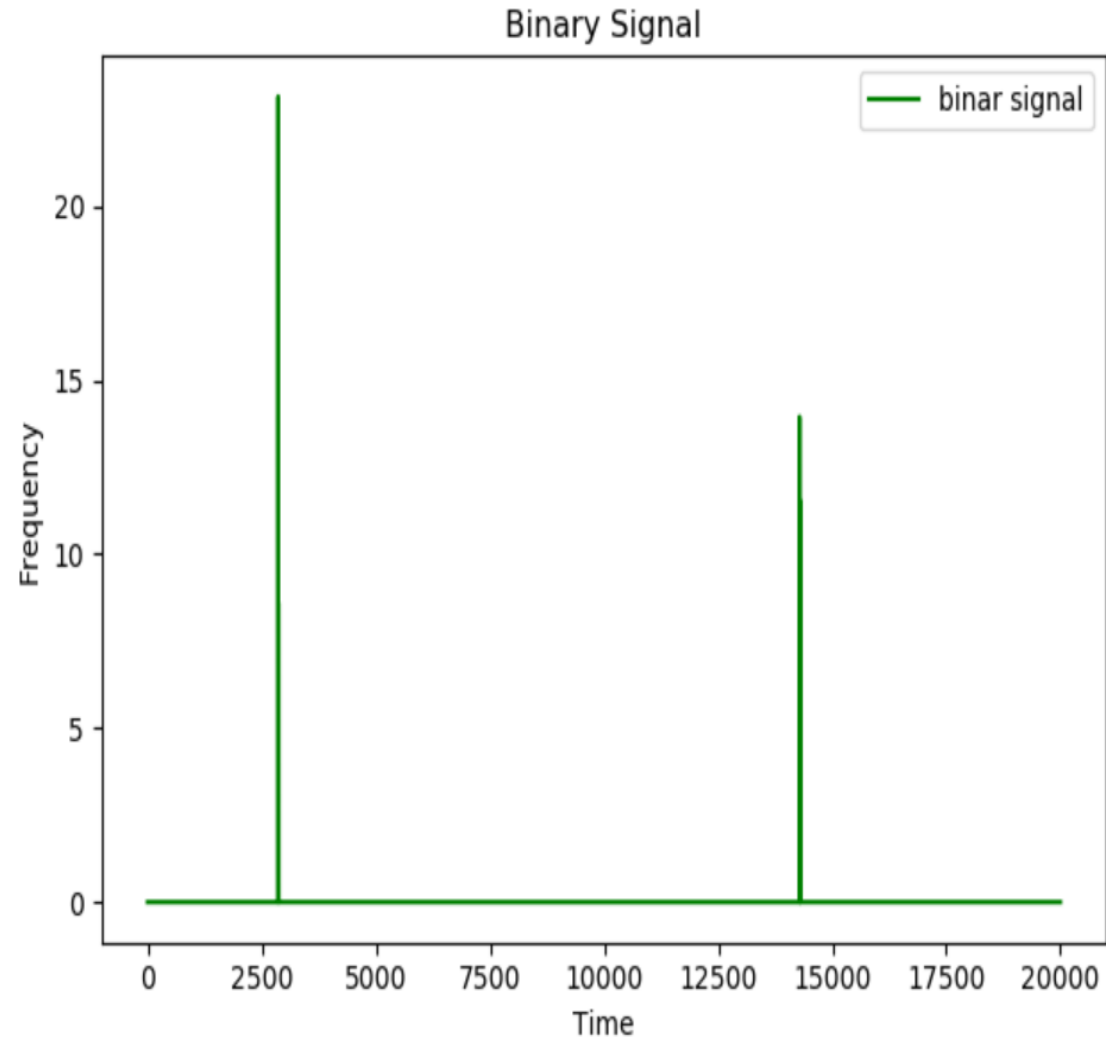
The image shows the Python logo (two interlocking snakes, one blue and one yellow) followed by the word 'python' in a grey font and 'Matplotlib' in a large, bold, blue font.

Структура предобработки ультразвукового сигнала



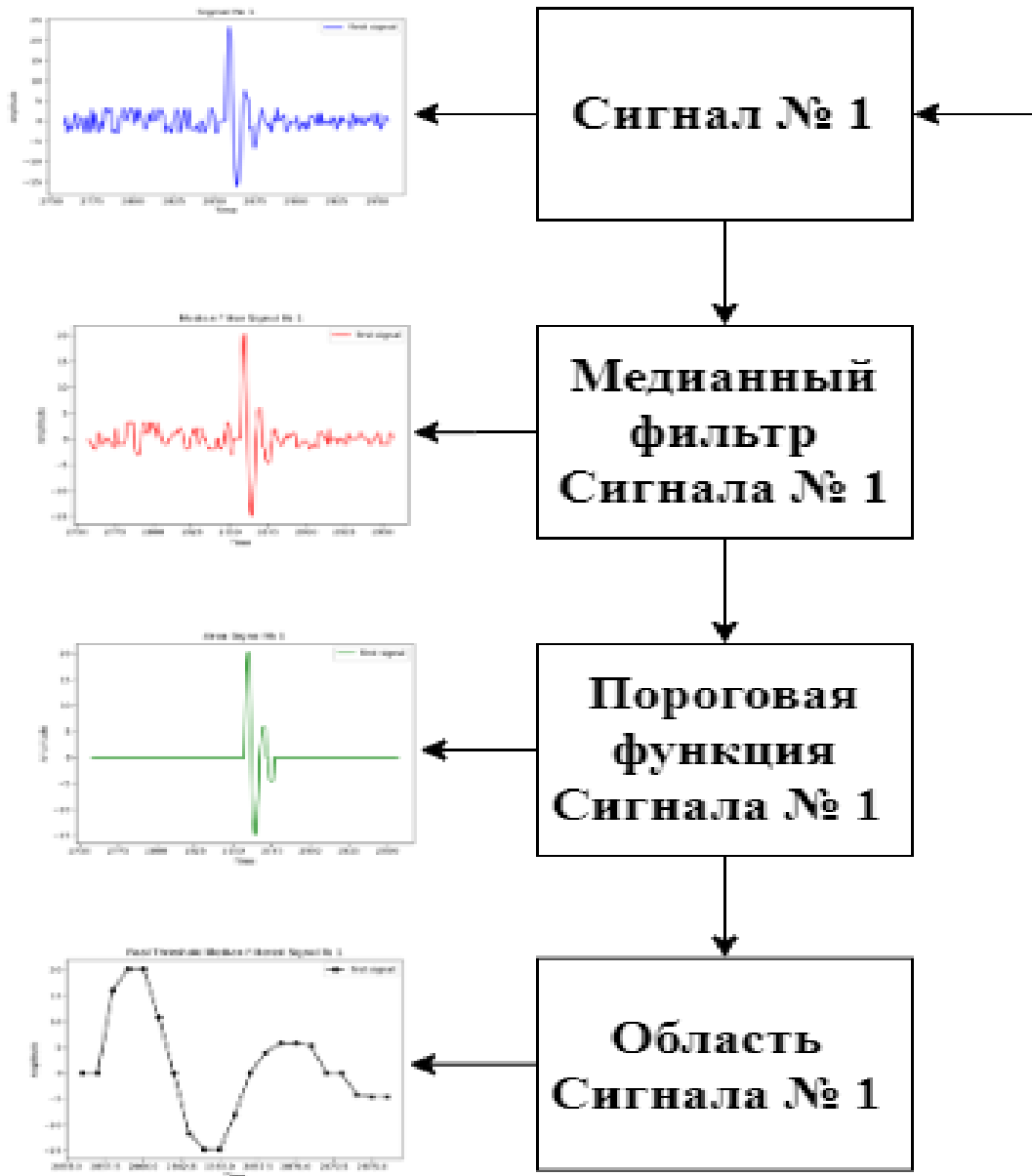


Способ преобразования сигнала в форму, которая позволяет сжать исходный набор данных



Цель бинаризации - радикальное уменьшение количества информации.

Предобработка сигнала № 1



Применение медианного фильтра оказывается более эффективным по сравнению с линейными фильтрами.

Фильтрация существенно уменьшает шумы, но чувствительна к выбросам

Предобработка сигнала № 2

Разработанный алгоритм сжатия ультразвукового сигнала состоит из следующих основных этапов:

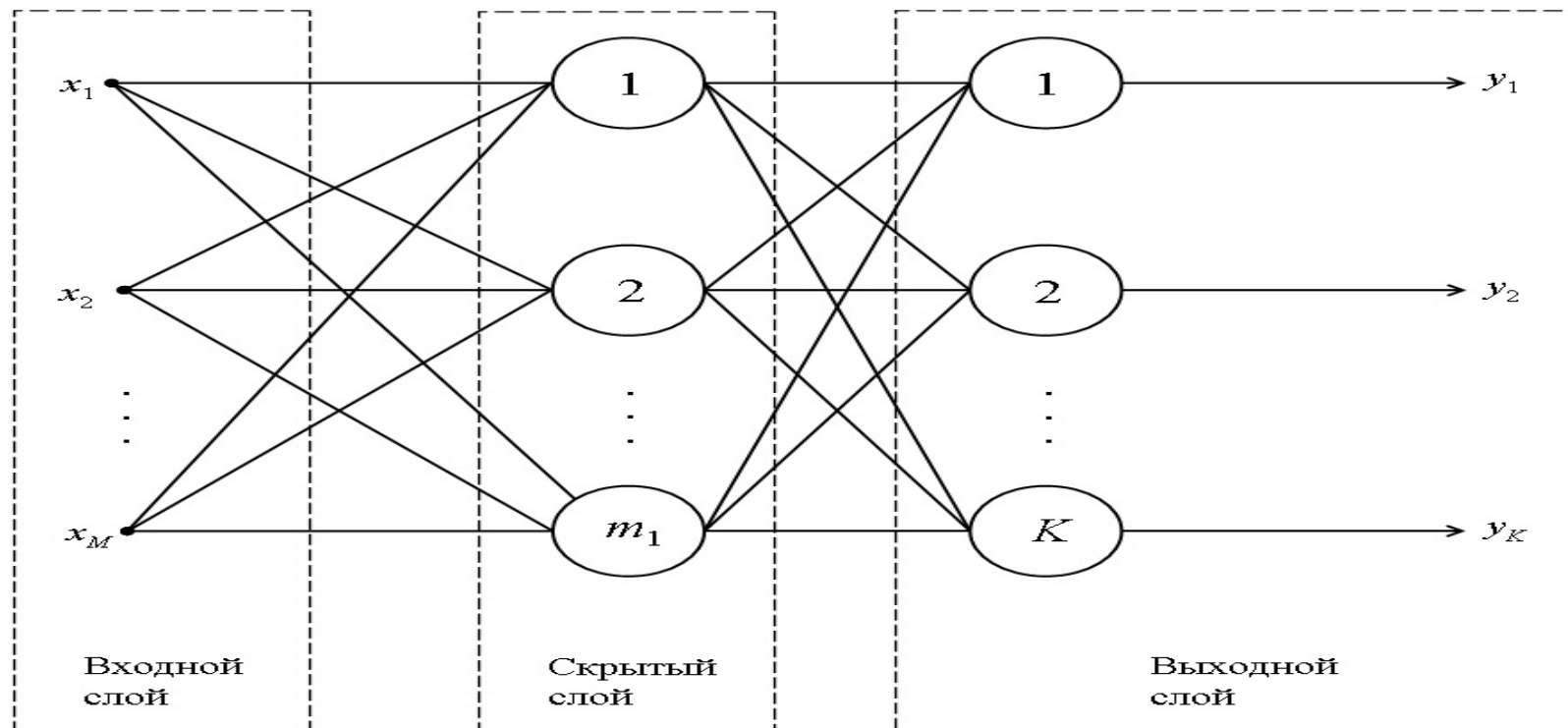
1. Разложение сигнала.
2. Модификация коэффициентов разложения.
3. Поиск областей ненулевых коэффициентов.
4. Сохранение результатов сжатия.



№	Сигнал № 1	D(t) Сигнала №1	Сигнал № 2	D(t) Сигнала №2
1.	0,000	0,000	0,000	0,000
2.	0,000	-17,941	0,000	-10,188
3.	17,941	-4,181	10,188	-4,524
4.	22,122	0,000	14,712	0,000
5.	22,122	11,317	14,712	2,713
6.	10,805	10,805	11,999	11,999
7.	0,000	8,692	0,000	5,842
8.	-8,692	2,143	-5,842	3,674
9.	-10,835	0,000	-9,516	0,000
10.	-10,835	-5,651	-9,516	-1,377
11.	-5,184	-5,184	-8,139	-8,139
12.	0,000	-7,855	0,000	-5,211
13.	7,855	-0,906	5,211	-0,349
14.	8,761	0,000	5,560	-1,580
15.	8,761	8,761	7,141	0,000
16.	0,000	0,000	7,141	7,141
17.	0,000	0,000	0,000	0,000
18.	0,000	5,227	0,000	0,000
19.	-5,227	0,000	0,000	0,000
20.	-5,227	-0,575	0,000	0,000
21.	-4,652	-4,652	0,000	0,000
22.	0,000	0,000	0,000	0,000
23.	0,000	-4,846	0,000	-5,889
24.	4,846	0,000	5,889	0,000
25.	4,846	None	5,889	None

Цель классификации – отнести входной образ к одному из известных классов.

Назначение классификатора – разделить это многомерное пространство на области, относящиеся к классам, и определять, к какой области относится тот или иной входной образ.



Этапы разработки сверточной нейронной сети в PyTorch

- Импортировать необходимые пакеты для создания нейронной сети
- Создать класс с представлением сверточной нейронной сети
- Определить необходимую комбинацию входных данных
- Создать модель с помощью встроенных функций
- Построить функцию потерь с помощью оптимизатора
- Реализовать модель с помощью итерационного цикла



Данный подход по обработке ультразвукового сигнала традиционными методами определяет область получения двух сигналов и рассчитывает время.

Метод классификации с использованием нейросети, при дальнейшей доработке сократит время и количество ошибок в процедуре градуировки ультразвуковых толщиномеров.

Спасибо за внимание.