**Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации**

**Ордена Трудового Красного Знамени**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Кафедра «Математическая кибернетика и информационные технологии»

Отчет по лабораторной работе №1

по дисциплине «Системы машинного зрения»

на тему:

«Разработка нейросетевых функций. Операция Convolution 2D»

Выполнил:

студент группы БВТ2003

Прус Владислав

**Цель работы**

Разработать алгоритм, используя язык python, реализующий работу операции двумерной свертки.

**Задание**

При выполнении лабораторной работы необходимо:

* ознакомиться с описанием операции библиотеки PyTorch (https://pytorch.org/docs/stable/generated/torch.nn.Conv2d);
* используя язык программирования Python написать алгоритм, реализующий алгоритм свертки;
* составить отчет по лабораторной работе.

**Теоретические сведения**

Свертка 2D (2D-convolution) – это один из базовых инструментов, применяемых при разработке функций нейросетей в контексте глубокого обучения и искусственного интеллекта. Она заключается в проведении математических операций над двумерными массивами данных (к примеру, изображениями) с целью извлечения из них значимых характеристик.

В рамках нейросетей операция свертки 2D предполагает фильтрацию входного изображения при помощи матриц фильтров, обучаемых на извлечение определенных характеристик изображения. Таким образом, сеть способна самостоятельно выявлять значимые признаки, не полагаясь на предопределенные функции.

Операция 2D свертки состоит из следующих шагов:

1. Загрузка входных данных: двумерное изображение преобразуется в одномерный массив (вектор), который затем используется как входной сигнал для операции свертки.
2. Определение фильтров: матрица фильтров представляет собой двумерный массив чисел, который используется для фильтрации входного изображения. Матрица фильтров обучается на основе набора обучающих данных, чтобы находить определенные особенности в изображениях.
3. Выполнение свертки: каждый фильтр применяется к входному изображению путем перемещения его по всему изображению. Каждый раз, когда фильтр накладывается на определенную область изображения, выполняется операция умножения между значениями фильтра и соответствующими значениями входного изображения, затем суммируются результаты умножения. Этот процесс повторяется для каждого фильтра, и результатом является набор карт признаков.
4. Нелинейное преобразование: после выполнения свертки для каждого фильтра обычно применяется нелинейная функция (например, сигмоид или ReLU), чтобы увеличить выразительность модели.
5. Построение выходных данных: выходные данные обычно представляют собой вектор, где каждая компонента соответствует одному фильтру. Этот вектор затем используется для последующих слоев нейронной сети, таких как полносвязные слои или слои активации.

Свертка 2D — важная функция обработки изображений и глубокого обучения. Она позволяет автоматически выделять значимые черты во входных изображениях, упрощая процесс обучения и делая модель более гибкой.

**Выполнение работы**

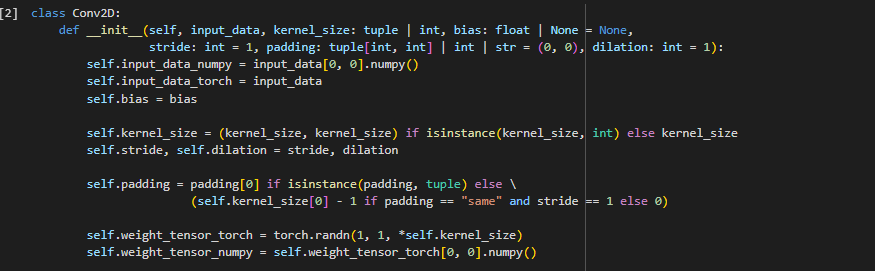
Создадим класс conv2D, который будет реализовывать операцию двумерной свертки. В этом классе опишем все необходимые функции. Начнем с создания функции для проверки соответствия входных данных (параметров) типу класса. Функция для инициализации объекта класса представлена в следующем блоке кода. 

Рисунок 1 – Инициализация объектов.

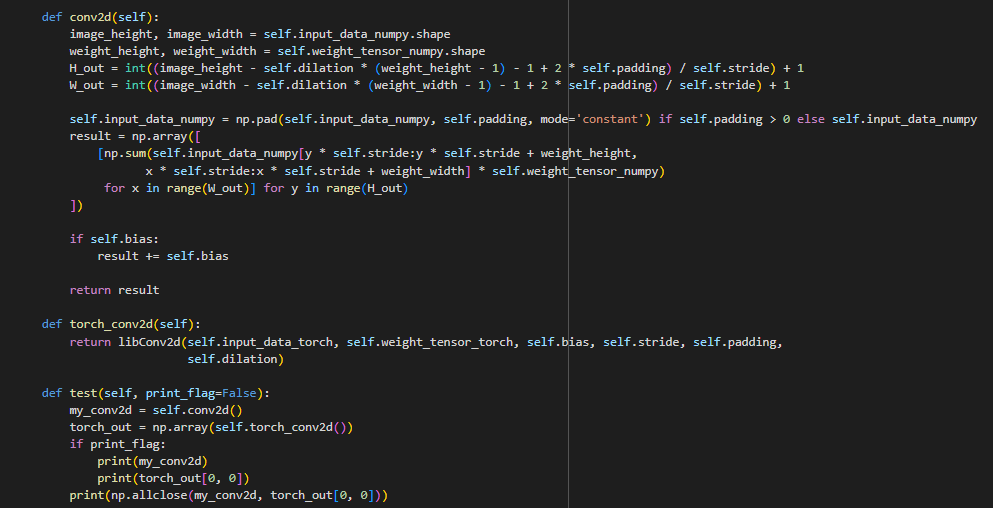
Ниже представлен сам алгоритм свертки, а также функция теста, по которой мы будем сравнивать значения нашей рукописной свертки с оригинальной сверткой из библиотеки torch: 

Рисунок 2 – Алгоритм двумерной свертки и функция тестирования

На рисунке 3 представлены результаты непосредственно самого тестирования для обычной свертки, для свертки с padding “valid”, с padding “same”, для свертки с большим размером kernel size, с добавлением dilation и с добавлением stride:

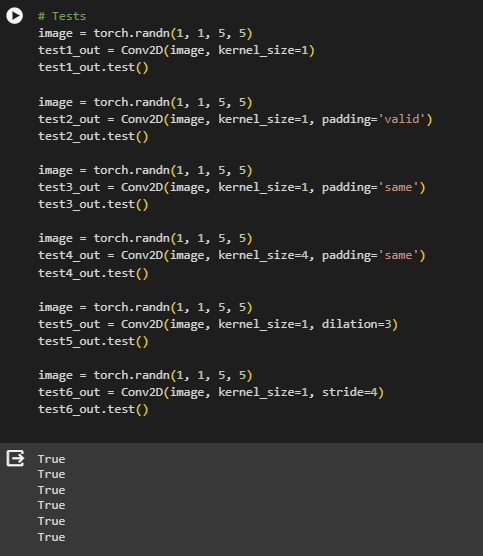
****

Рисунок 3 – Результаты всех шести тестов.

**Вывод**

В ходе выполнения данной лабораторной работы я разработал алгоритм на языке Python, который реализует операцию двумерной свертки. Также я создал тесты для проверки корректности работы алгоритма.

Ссылка на git-репозиторий: https://github.com/Sombdy021/CMZ