**Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования**

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

Кафедра Математической кибернетики и информационных технологий

**Лабораторная работа № 3**

«Разработка нейросетевых функций. Операция Convolution Transpose»

по дисциплине «Системы машинного зрения»

Выполнила студентка группы БВТ2001:

Рыжкова Ульяна

Москва 2023

**Цель работы**

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

**Задание**

При выполнении лабораторной работы необходимо:

1. Ознакомиться с описанием операции библиотеки PyTorch;
2. Используя язык программирования Python написать алгоритм, реализующий алгоритм свертки;
3. Составить отчет по лабораторной работе.

**Выполнение**

Рассмотрим функцию conv\_transpose2d библиотеки PyTotch:

torch.nn.functional.conv\_transpose2d(input, weight, bias=None, stride=1, padding=0, output\_padding=0, groups=1, dilation=1) → Tensor

Как мы видим на вход функции подается ряд параметров, а на выходе мы получаем тензор. Для реализации аналогичной функции необходимо разобраться за что отвечает каждый параметр, как он влияет на алгоритм транспонированной 2D-свертки.

* **input** – входной тензор размером (*minibatch, in\_channels*, *iH*, *iW*);
* **weight –** фильтры размером (*in\_channels,,kH*,*kW*);
* **bias –** опциональный тензор смещения размером (*out\_channels*);
* **stride –** шаг свертки;
* **padding –** дополнительные значения пикселей по краям входных данных, имеет несколько видов, в зависимости от которых меняются значения дополнительных пикселей;
* **output\_padding –** дополнительный размер, добавленный к одной стороне каждого измерения в размере выходных данных;
* **dilation –** расстояние между элементами ядра;
* **groups –** количество групп;

Несмотря на то, что существует 4 вида paddingа ('constant', 'reflect', 'replicate' or 'circular'), в данной лабораторной значение дополнительных пикселей будет 0, а меняться будет их количество (размер paddingа).

В данной реализации не предусмотрена работа с groups, но работа с bias, stride, padding, output\_ padding и dilation присутствует.

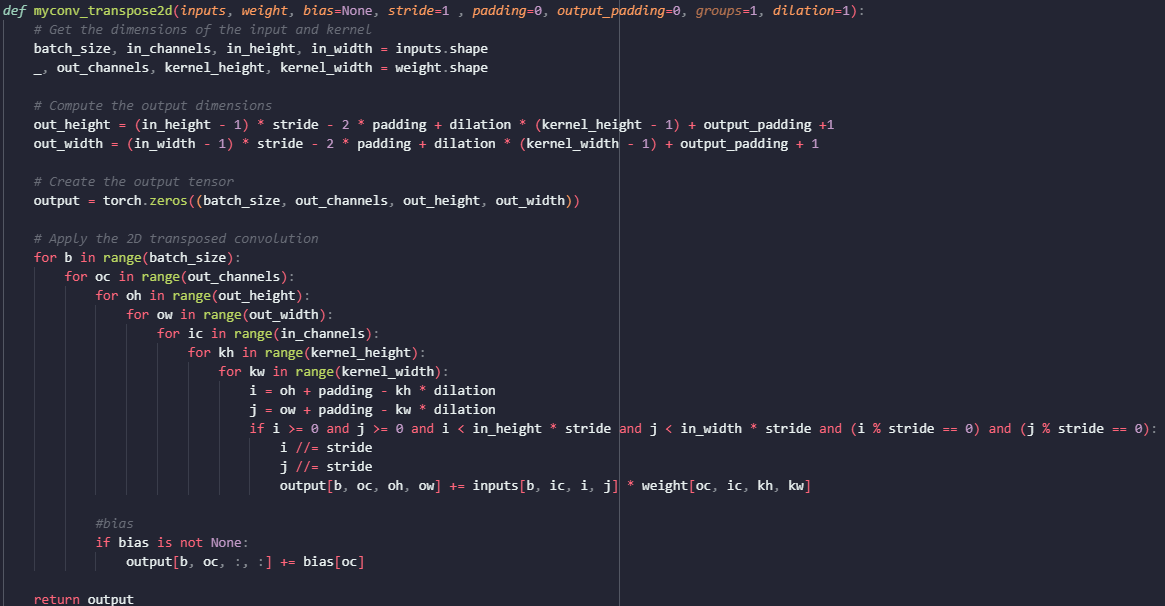


Рисунок 1 – Реализация функции транспонированной 2D-свертки

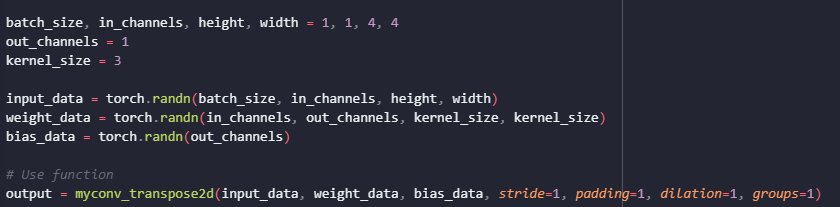


Рисунок 2 – Пример вызова функции и создания входных данных

Также по заданию было необходимо реализовать проверку правильности работы созданной функции с помощью тестов. Код тестов представлен на рисунке ниже.



Рисунок 3 – Тесты

Функция успешно прошла 4 теста, в которых были задействованы разные параметры функции.



Рисунок 4 – Результат тестов

**Вывод**

В данной лабораторной работе мы изучили принцип транспонированной 2D-свертки, познакомились с функцией torch.nn.functional.conv\_transpose2d, разобрались в том, за что отвечают ее входные параметры и реализовали собственную функцию свертки.

Ссылка на место хранения: https://github.com/Uliana2303/SMV