

(۱)

(الف)

شبکه های عصبی کانولوشنی مستعد خراب شدن با نویز هستند، مقدار کمی نویز در تصویر ورودی میتواند تاثیر شدیدی در خروجی داشته باشد. برای خنثی کردن اثر منفی نویز میتوان از تبدیل DWT به جای Max-pooling، strided-convolution، average-pooling استفاده کرد، زیرا این تبدیل هم برای ما DownSampling انجام میدهد و هم فرکانس های بالای تصویر را از فرکانس های پایین آن جدا میکند و چون نویز ها اغلب در فرکانس های بالا حضور دارند، این کار باعث حذف نویز هم میشود؛ در صورتی که روش های اشاره شده فقط DownSampling انجام میدهند و denoising ندارند.

(ب)

چالش های پیاده سازی این روش، پیش خور داده ها و پس انتشار خطا در لایه های DWT و IDWT است. در شبکه های WaveCNet از تبدیل wavelet به منظور denoising و همچنین downsampling به جای لایه های pooling و یا strided-convolution استفاده میکنیم به این طریق که مولفه های فرکانس بالای سیگنال ورودی به منظور حذف نویز، حذف میشوند و مولفه فرکانس پایین سیگنال (approximation) عبور داده میشود.

(ج)

در این روش، یکی از معماری های معروف طبقه بندی تصاویر را در نظر میگیریم و به جای لایه های pooling و یا strided-convolution آن، از DWT و IDWT استفاده میکنیم زیرا این تبدیل های علاوه بر downsampling میتوانند denoising هم انجام دهند.

ابتدا از سیگنال تبدیل DWT میگیریم و آن را به مولفه های تقریب و جزییات میشکنیم، سپس مولفه های جزییات را به منظور کاهش نویز، حذف میکنیم، حال تبدیل معکوس میگیریم تا سیگنال را به فضای قبلی برگردانیم.

۲-

از روابط زیر استفاده میکنیم

$$W_2 = \frac{W_1 - F + 2P}{S} + 1 \quad H_2 = \frac{H_1 - F + 2P}{S} + 1 \quad D_2 = K$$

K: تعداد فیلتر ها

F: اندازه فیلتر ها

S: اندازه گام

P: مقدار گسترش مرز ها (padding)

(الف)

$$\frac{(7 - 7 + 2 \times 0)}{1} + 1 = 1$$

$$\frac{(7 - 7 + 2 \times 0)}{1} + 1 = 1$$

خروجی یک وکتور $1 \times 1 \times 1$ خواهد بود.

(ب)

در هر مرحله عمق کرنل فیلتر با عمق ورودی برابر است

در مرحله اول کرنل فیلتر یک ماتریس $3 \times 3 \times 3$ است

$$W_1 = \frac{(7) - 3 + 2(0)}{1} + 1 = 5$$

$$H_1 = \frac{(7) - 3 + 2(0)}{1} + 1 = 5$$

$$D_1 = 1$$

در مرحله دوم، کرنل فیلتر یک ماتریس $3 \times 3 \times 1$ است

$$W_2 = \frac{(5) - 3 + 2(0)}{1} + 1 = 3$$

$$H_2 = \frac{(5) - 3 + 2(0)}{1} + 1 = 3$$

$$D_2 = 1$$

در مرحله دوم، کرنل فیلتر یک ماتریس $3 \times 3 \times 1$ است

$$W_3 = \frac{(3) - 3 + 2(0)}{1} + 1 = 1$$

$$H_3 = \frac{(3) - 3 + 2(0)}{1} + 1 = 1$$

$$D_3 = 1$$

در روش اول تعداد پارامترها برابر است با:

$$7 * 7 * 3 * 1 + 1 = 148$$

در روش دوم تعداد پارامترها برابر است با:

$$3 * 3 * 3 * 1 + 1 = 28$$

$$3 * 3 * 1 * 1 + 1 = 10$$

$$3 * 3 * 1 * 1 + 1 = 10$$

مجموع پارامترهای روش دوم برابر است با:

$$28 + 10 + 10 = 48$$

روش دوم نسبت به روش اول تعداد کمتری پارامتر دارد که باعث کاهش محاسبات و مصرف کمتر منابع سخت افزاری میشود.

کانولوشن یک عملگر خطی است و ویژگی های خطی را استخراج میکند.
در روش دوم میتوانیم بین هر لایه از یک تابع خیر خطی استفاده کنیم و از این طریق خاصیت غیر خطی شبکه را تقویت کنیم و ویژگی های عمیق تری را استخراج کنیم.

در عملیات کانولوشن ابتدا باید کرنل آن را معکوس کنیم و سپس روی ماتریس تصویر اعمال کنیم.

0	0	2
0	4	1
0	1	10



10	1	0
1	4	0
2	0	0

12	52	10	54	56
104	235	74	96	72
12	12	52	222	220
201	32	36	35	35
82	35	35	35	35

$$P1 = 10 * 12 + 1 * 52 + 0 * 10 + 1 * 104 + 4 * 235 + 0 * 72 + 2 * 12 + 0 * 12 + 0 * 52 = 1240$$

خروجی:

1240		

10	1	0
1	4	0
2	0	0

12	52	10	54	56
104	235	74	96	72
12	12	52	222	220
201	32	36	35	35
82	35	35	35	35

$$P1 = 10 * 52 + 1 * 10 + 0 * 54 + 1 * 235 + 4 * 74 + 0 * 96 + 2 * 12 + 0 * 52 + 0 * 222 = 1085$$

خروجی:

1240	1085	

10	1	0
1	4	0
2	0	0

12	52	10	54	56
104	235	74	96	72
12	12	52	222	220
201	32	36	35	35
82	35	35	35	35

$$P1 = 10 * 10 + 1 * 54 + 0 * 56 + 1 * 74 + 4 * 96 + 0 * 72 + 2 * 52 + 0 * 222 + 0 * 220 = 716$$

خروجی:

1240	1085	716

اگر به همین ترتیب تا پایان ادامه دهیم خروجی به صورت زیر خواهد بود:

1240	1085	756
1737	2708	1848
625	418	988

(ب)

مشابه قبل ابتدا کرنل را معکوس میکنیم:

0	0	2
0	4	1
0	1	10



10	1	0
1	4	0
2	0	0

12	52	10	54	56
104	235	74	96	72
12	12	52	222	220
201	32	36	35	35
82	35	35	35	35

$$P1 = 10 * 12 + 1 * 52 + 0 * 10 + 1 * 104 + 4 * 235 + 0 * 72 + 2 * 12 + 0 * 12 + 0 * 12 = 1240$$

10	1	0
1	4	0
2	0	0

12	52	10	54	56
104	235	74	96	72
12	12	52	222	220
201	32	36	35	35
82	35	35	35	35

$$P1 = 10 * 10 + 1 * 54 + 0 * 56 + 1 * 74 + 4 * 96 + 0 * 72 + 2 * 52 + 0 * 222 + 0 * 220 = 716$$

10	1	0
1	4	0
2	0	0

12	52	10	54	56
104	235	74	96	72
12	12	52	222	220
201	32	36	35	35
82	35	35	35	35

$$P1 = 10 * 12 + 1 * 12 + 0 * 52 + 1 * 201 + 4 * 32 + 0 * 36 + 2 * 82 + 0 * 35 + 0 * 35 = 625$$

خروجی نهایی برابر است با:

1238	756
625	1188

(ج)
از کرنل ۲ در ۲ استفاده میکنیم.

12	52	10	54
104	235	74	96
12	12	52	222
201	32	36	35

$$Max(12,52,104,235) = 235$$

12	52	10	54
104	235	74	96
12	12	52	222
201	32	36	35

$Max(10,54,74,96) = 96$

12	52	10	54
104	235	74	96
12	12	52	222
201	32	36	35

$Max(12,12,201,32) = 201$

خروجی نهایی:

235	96
201	222

(۵)

12	52	10	54
104	235	74	96
12	12	52	222
201	32	36	35

$\frac{12 + 52 + 104 + 235}{4} = 100.75$

12	52	10	54
104	235	74	96
12	12	52	222
201	32	36	35

$\frac{10 + 54 + 74 + 96}{4} = 58.5$

12	52	10	54
104	235	74	96
12	12	52	222
201	32	36	35

$\frac{12 + 12 + 201 + 32}{4} = 64.25$

خروجی نهایی:

100.25	58.5
64.25	86.25

12	52	10	54	56
104	235	74	96	72
12	12	52	222	220
201	32	36	35	35
82	35	35	35	35

$$\frac{12 + 52 + 10 + 54 + 56 + 104 + 235 + 74 + 96 + 72 + 12 + 12 + 52 + 222 + 220 + 201 + 32 + 36 + 35 + 35 + 82 + 35 + 35 + 35 + 35}{25} = 73.76$$

12	52	10	54
104	235	74	96
12	12	52	222
201	32	36	35

$$\frac{12 + 52 + 10 + 54 + 104 + 235 + 74 + 96 + 12 + 12 + 52 + 222 + 201 + 32 + 36 + 35}{16} = 77.43$$

$$W_2 = \frac{W_1 - F + 2P}{S} + 1 \quad H_2 = \frac{H_1 - F + 2P}{S} + 1 \quad D_2 = K$$

K: تعداد فیلتر ها

F: اندازه فیلتر ها

S: اندازه گام

P: مقدار گسترش مرز ها (padding)

با استفاده از روابط بالا سایز خروجی را محاسبه میکنیم.

لایه اول: Input

ابعاد ورودی: $128 \times 128 \times 3$

ابعاد خروجی: $128 \times 128 \times 3$

لایه دوم: Con2D

ابعاد ورودی: $128 \times 128 \times 3$

محاسبه ابعاد خروجی:

$$K=32, F=5, S=1, P=0$$

$$W_1 = \frac{(128) - 5 + 2(0)}{1} + 1 = 124$$

$$H_1 = \frac{(128) - 5 + 2(0)}{1} + 1 = 124$$

$$D_1 = 32$$

ابعاد خروجی: $124 \times 124 \times 32$

لایه سوم: Con2D

ابعاد ورودی: $124 \times 124 \times 32$

محاسبه ابعاد خروجی:

$$K=64, F=5, S=2, P=0$$

$$W_2 = \frac{(124) - 5 + 2(0)}{2} + 1 = 59.5 \approx 60$$

$$H_2 = \frac{(124) - 5 + 2(0)}{2} + 1 = 59.5 \approx 60$$

$$D_2 = 64$$

ابعاد خروجی: $60 \times 60 \times 64$

برای لایه pooling از روابط مشابهی با لایه کانولوشنی استفاده میشود:

$$W_2 = \frac{W_1 - F + 2P}{S} + 1 \quad H_2 = \frac{H_1 - F + 2P}{S} + 1 \quad D_2 = D_1$$

K: تعداد فیلتر ها

F: اندازه فیلتر ها

S: اندازه گام

P: مقدار گسترش مرز ها (padding)

لایه چهارم: average pooling

ابعاد ورودی: $60 \times 60 \times 64$

$$D_2=64, F=5, S=2, P=0$$

$$W_3 = \frac{(60) - 10 + 2(0)}{10} + 1 = 6$$

$$H_3 = \frac{(60) - 10 + 2(0)}{10} + 1 = 6$$

$$D_3 = 64$$

ابعاد خروجی: $6 \times 6 \times 64$

لایه Flatten:

ابعاد ورودی: $6 \times 6 \times 64$

ابعاد خروجی: 2304

لایه Dense:

ابعاد ورودی: 2304

ابعاد خروجی: 32

لایه Dense:

ابعاد ورودی: 32

ابعاد خروجی: 10

(ب)

لایه های کانولوشنی:

$$F.F.D_1.K + K$$

$$5 * 5 * 32 * 3 + 32 = 2432$$

$$5 * 5 * 64 * 32 + 64 = 51264$$

لایه های pooling، پارامتری برای یادگیری ندارند:

0

0

در لایه های dense، هر نورون به تعداد خروجی های لایه قبلی وزن دارد و هر لایه یک بایاس هم دارد:

$$2304 * 32 + 32 = 73760$$

$$32 * 10 + 10 = 330$$

