دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

دانشکده برق

موضوع: Mobile FaceNet

درس: هوش محاسباتی

استاد: جناب آقای دکتر طالبی

نگارش : سجاد قدیری

محمد برآبادی

مارال مرداد

محیا حقگو

پاییز 1400

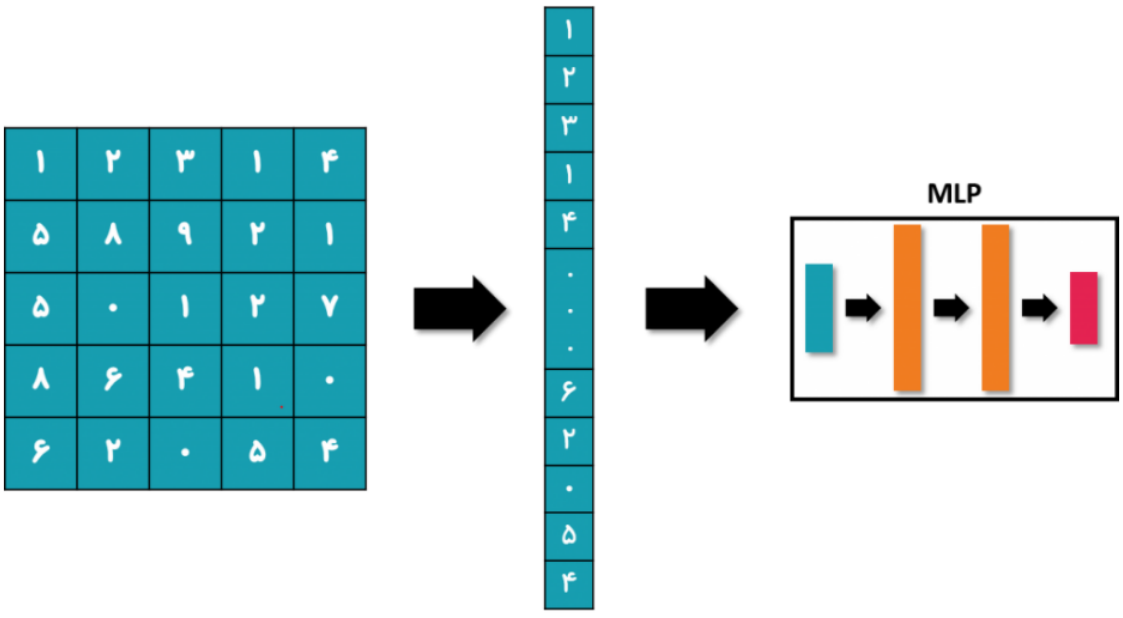
شبکه عصبی کانولوشن[1]:

شبکه عصبی کانولوشن بسیار شبیه MLP است. این شبکه ها از نورون‌هایی تشکیل شده اند که دارای وزن‌های قابل یادگیری و بایاس هستند. به هر نورون یک سری ورودی داده می‌شود و با یک ضرب داخلی و عبور از تابع فعال‌ساز خروجی را می‌سازد.

وزن‌ها در هر سری یادگیری بر اساس مینیمم سازی یک تابع هزینه آپدیت می‌شوند. پس تا اینجا شبکه MLP و CNN بسیار شبیه به هم بودند. تفاوت آن‌ها در ورودی است.

اگر در شبکه MLP یک تصویر 100*100 ورودی بدهیم در کل 10000 پیکسل دارد، شبکه برای دریافت ورودی این عکس را به یک آرایه یک بعدی 10000 تایی تبدیل می‌کند.

برای ساخت ورودی شبکه MLP نیاز است 10000 نورون در لایه ورودی قرار دهیم. که این تعداد نورون بسیار زیاد باعث زمان بر بودن فرآیند یادگیری و اتفاق افتادن بیش برازش[2] در زمان یادگیری می‌شود همچنین پارامترهای شبکه به شدت زیاد می‌شوند.



[1] CNN
[2] overfitting

شبکه CNN برخلاف MLP ساختار ورودی را تغییر نمی‌دهد. ورودی شبکه CNN می‌تواند ماتریس های یک بعدی مانند سیگنال، دو بعدی مانند تصویر و طیف صوت، سه بعدی مانند ویدئو و تصاویر حجمی وداده‌های چهاربعدی مانند تصاویر حجمی همراه با زمان باشند. این شبکه به ارتباط بین پیکسل‌های همسایه اهمیت می‌دهد.

شبکه CNN از لایه‌های مختلفی تشکیل شده است که عبارتند از:

- لایه ورودی (Input layer)
- لایه کانولوشن (Convolutional layer)
- لایه غیرخطی (Non-linear activation function)
- لایه پولینگ (Pooling layer)
- لایه فولی کانکتد (Fully connected layer)

معروف ترین ورودی شبکه CNN تصویر می‌باشد.

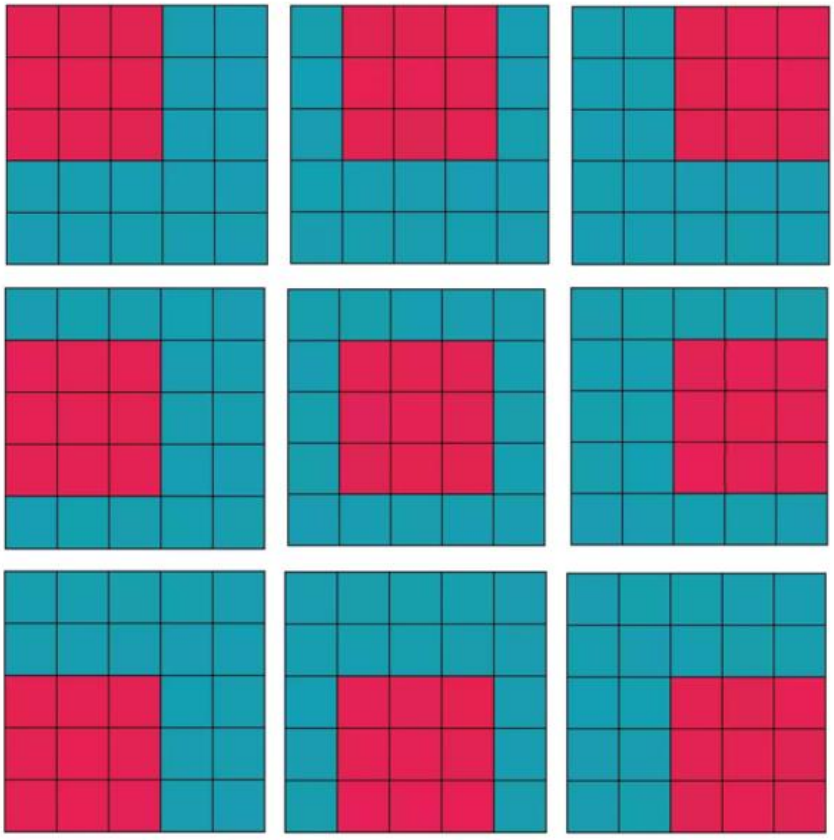
تصاویر ورودی می‌توانند خاکستری یا رنگی باشند.

تصویر رنگی از سه صفحه تشکیل شده است. این صفحات عبارتند از صفحه قرمز (R)، صفحه سبز (G) و صفحه آبی (B) به همین علت به این تصاویر، تصاویر RGB نیز گفته می‌شود.

هر یک از این صفحات یک ماتریس دوبعدی از اعداد بین 0 تا 255 می‌باشند. از ترکیب این سه صفحه یک تصویر رنگی حاصل می‌شود.

برای توضیح عملکرد کانولوشن می‌توان گفت، عملگر کانولوشن، کرنل یا فیلتر کانولوشنی را برمی‌دارد و روی تصویر یا ماتریس ورودی می‌لغزاند. فیلتر ابتدا هر سطر را ستون به ستون طی می‌کند و بعد یک سطر پایین می‌آید و دوباره ستون به ستون جلو می‌رود و این فرآیند تا آخر ادامه دارد.

برای مثال روند حرکت فیلتر بر تصویر ورودی به صورت زیر می‌باشد:



هر فیلتر کانولوشنی، شامل مجموعه ای عدد است. با قرار گرفتن فیلتر روی هر بخش از تصویر، اعداد فیلتر درایه به درایه در پیکسل تصویر متنلظر ضرب می‌شوند و در نهایت همه اعداد با هم جمع می‌شوند.

بسته به اندازه فیلتر، اندازه ماتریس خروجی تغییر می‌کند. اگر ماتریس ورودی و فیلتر مربعی باشند و ابعاد آن‌ها به ترتیب n و k باشند، بعد ماتریس خروجی(m) به صورت زیر می‌باشد:

$$m = n - (k - 1)$$

اگر بخواهیم بعد ماتریس خروجی برابر با بعد ماتریس ورودی باشد، می‌توان از روش لایه گذاری [۳] استفاده کرد. یک راه ساده و رایج لایه گذاری، اضافه کردن سطر و ستون صفر به صورت متقارن به دور ماتریس ورودی است. به این روش لایه گذاری صفر[۴] گفته می‌شود.

فیلتر به دنبال پیدا کردن نواحی مشابه خود در تصویر است. پس کانولوشن منجر به یافتن الگوهای خاص در تصویر با توجه به فیلتر می‌شود. اعداد موجود در فیلتر از طریق آموزش به دست می‌آیند.

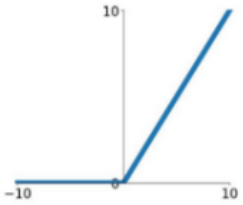
هر صفحه از تصویر رنگی به طور مجزا برای خودش یک فیلتر دارد. تصویر رنگی سه صفحه دارد بنابراین فیلتر هم باید سه صفحه داشته باشد. سه صفحه از فیلتر به طور موازی با هم روی سه صفحه تصویر حرکت می‌کنند و در نهایت نتیجه ضرب سه صفحه با هم جمع می‌شوند و خروجی نهایی یک تصویر تک صفحه است. هریک از فیلترها به تنهایی برای تشخیص یک الگوی خاص کاربرد دارد و برای تشخیص یک مجموعه الگو باید از چندین فیلتر استفاده کرد. تعداد فیلترها معمولا نمایی از 2 (بین 32 تا 4096) می‌باشد. استفاده از فیلترهای بیشتر موجب قدرتمندتر شدن شبکه عصبی می‌شود.

بنابراین هر لایه کانولوشن در شبکه عصبی شامل مجموعه ای فیلترهاست و خروجی از کانولوشن فیلترها و ورودی به دست می‌آید. به خروجی لایه کانولوشنی Feature map گفته می‌شوند.

مشابه با سایر شبکه های عصبی، شبکه عصبی کانولوشن هم از تابع تحریک غیرخطی بعد از لایه کانولوشن استفاده می‌کند. از تابع ReLU به عنوان تابع غیرخطی استفاده می‌شود که به صورت زیر می‌باشد:

ReLU

$$\max(0, x)$$



هدف لایه پولینگ کاهش اندازه مکانی Feature map به دست آمده از لایه کانولوشنی است.

لایه پولینگ پارامتر قابل آموش ندارد. عملکرد پولینگ مشابه عملکرد کانولوشن است. یک پنجره از پیش تعریف شده که روی تصویر حرکت می‌کند و در هر پنجره ماکزیمم را انتخاب می‌کند و بقیه را دور می‌ریزد.

این سه مرحله چندبار تکرار می‌شوند تا به ارور مطلوبی برسیم.

---

[3] padding
[4] Zero padding

آخرین لایه های شبکه عصبی کانولوشن، لایه فولی کانکتد است که کاربرد آن طبقه بندی (classification) می‌باشد. مجموعه ویژگی های استخراج شده با استفاده از لایه‌های کانولوشنی یک بردار هستند که در نهایت این بردار ویژگی ها به فولی کانکتد داده می‌شود تا کلاس درست را شناسایی کند.

معماری‌های متفاوت شبکه‌های CNN:

معماری‌های قدیمی‌تر به سادگی از لایه‌های کانولوشن انباشته تشکیل شده می‌شدند ولی معماری‌های مدرن راه‌های جدید و نوآورانه‌ای را برای ساخت لایه‌های کانولوشن به گونه ای که امکان یادگیری کارآمدتر شود ارائه داده‌اند. تقریبا همه این معماری‌ها بر اساس یک واحد تکرارپذیر هستند که در سراسر شبکه استفاده می‌شود.

روند اولیه برای توسعه مدل‌های CNN عمیق‌تر و پیچیده تر کردن شبکه برای رسیدن به دقت بالاتر بود.مانند VGGNet، GoogleNet، ResNetو DenseNet . اگرچه این روند بهبود مدل‌ها باعث افزایش سایز آن‌ها می‌شد. این افزایش سایز باعث افزایش محاسبات و حافظه مورد نیاز برای ذخیره سازی آنها می‌شود. برای حل این مشکل راندمان پایین ناشی از مدل‌های پیچیده محققین شبکه‌های CNN سبک وزن را پیشنهاد دادند. معماری‌های خاص و کارآمدی را برای ساخت CNN های سبک وزن پیشنهاد کردند.

SqueezeNet یک شبکه سبک وزن نسبتا اولیه است که در ICLR2017[5] پیشنهاد شده است. که می‌تواند به دقت مشابه مدل AlexNet روی داده‌های ImageNet با کاهش 50 برابری پارامترها برسد. ماژول آتش استفاده شده در SqueezNet عامل اصلی کاهش تعداد پارامترها است و از squeeze convolution module و ماژول گسترش تشکیل شده است. Squeeze convolution module به فشرده سازی تعداد کانال‌های ورودی کمک می‌کند. و فیلترهای 3*3 ماژول گسترش را با فیلترهای 1*1 جایگزین می‌کند. ShuffleNetV1 از عملیات کانولوشن گروهی نقطه‌ای و shuffle کردن کانال برای کاهش پارامترها و محاسبات و در عین حال حفظ دقت استفاده می‌کند. ShuffleNetV2 برای افزایش سرعت آموزش دیگر از تعداد زیادی کانولوشن گروهی استفاده نمی‌کند اما عملکرد شبکه را از طریق عملیات تقسیم کانال بهبود می‌بخشد. MobileNetV1 یک عملیات کانولوشن جدید را پیشنهاد کرد. اسم این عملیات کانولوشن، کانولوشن عمقی تفکیک پذیر است[6] که کانولوشن استاندارد را به دو بخش کانولوشن عمقی و کانولوشن نقطه‌ای[7] تقسیم می‌کند و محاسبات اضافی را تا حد زیادی کاهش

[5] International Conference on Learning Representations
[6] Depthwise separable convolutions
[7] pointwise

می‌دهد این عملیات کانولوشن در ادامه به صورت کامل توضیح داده شده است. MobileNetV2 برای بهبود مدل MobileNetV1 باقی‌مانده های معکوس[8] و گلوگاه خطی[9] معرفی کرد.

برای رسیدگی به مشکلات ناشی از مدل‌های پیچیده محققان چندین معماری شبکه خاص را برای تسک تشخیص چهره طراحی کردند.

MobileFaceNet یک مدل بر پایه MobileNetV2 است که برای تشخیص چهره استفاده می‌شود و به صورت تئوری به دقت 99.55٪ بر روی دادگان LFW[10] می‌رسد.

معماری Mobile facenet :

خلاصه ای از معماری Mobile facenet در جدول زیر آورده شده‌است:

Input	Operator	t	c	n	s
$112 \times 112 \times 3$	conv3 × 3	-	64	1	2
$56 \times 56 \times 64$	depthwise conv3 × 3	-	64	1	1
$56 \times 56 \times 64$	bottleneck	2	64	5	2
$28 \times 28 \times 64$	bottleneck	4	128	1	2
$14 \times 14 \times 128$	bottleneck	2	128	6	1
$14 \times 14 \times 128$	bottleneck	4	128	1	2
$7 \times 7 \times 128$	bottleneck	2	128	2	1
$7 \times 7 \times 128$	conv1 × 1	-	512	1	1
$7 \times 7 \times 512$	linear GDConv7 × 7	-	512	1	1
$1 \times 1 \times 512$	linear conv1 × 1	-	128	1	1

که پارامترهای t , c , n , s به ترتیب ضریب گسترش، تعداد کانال خروجی، تعداد تکرار و طول گام می‌باشد.

توضیح چند سطر از جدول در زیر آورده شده‌است:

سطر اول جدول کانولوشن استاندارد ورودی سه کاناله با ابعاد $112*112$ و کرنل سه کاناله با ابعاد $3*3$ می‌باشد. تعداد تکرار وگام و کرنل‌ها به ترتیب 1 و 2 و 64 می‌باشد. در نتیجه خروجی 64 کاناله است.

سطر دوم جدول کانولوشن عمقی ورودی 64 کاناله با ابعاد $56*56$ و کرنل 64 کاناله با ابعاد $3*3$ می‌باشد. تعداد تکرار وگام و کرنل‌ها به ترتیب 1 و 1 و 64 می‌باشد. در نتیجه خروجی 64 کاناله است.

[8] Inverted residuals
[9] Linear bottlenecks
[10] Labeled Faces in the Wild

سطر سوم از لایه Bottleneck استفاده شده‌است. این لایه یک کانولوشن 1*1 می‌باشد که ابعاد خروجی را به منظور جلوگیری از Overfitting و vanishing gradiant کاهش می‌دهد.

مقایسه Mobile facenet و سایر معماری‌های CNN :

Network	LFW	AgeDB-30	Params	Speed
MobileNetV1	98.63%	88.95%	3.2M	60ms
ShuffleNet (1×, g = 3)	98.70%	89.27%	**0.83M**	27ms
MobileNetV2	98.58%	88.81%	2.1M	49ms
MobileNetV2-GDConv	98.88%	90.67%	2.1M	50ms
MobileFaceNet	**99.28%**	**93.05%**	**0.99M**	24ms
MobileFaceNet (112 × 96)	99.18%	92.96%	0.99M	21ms
MobileFaceNet (96 × 96)	99.08%	92.63%	0.99M	**18ms**
MobileFaceNet-M	99.18%	92.67%	0.92M	24ms
MobileFaceNet-S	99.00%	92.48%	**0.84M**	23ms
MobileFaceNet (ReLU)	99.15%	92.83%	0.98M	23ms
MobileFaceNet (expansion factor ×2)	99.10%	92.81%	1.1M	27ms

کاربردها :

1- Google Lens :

تشخیص مکان ها و بناهای تاریخی، متن ها، حیوانات و اشیاء مختلف ترجمه متن به زبان های مختلف از روی عکس گرفته شده توسط دوربین، ذخیره شماره تلفن و آدرس از روی عکس گرفته شده توسط دوربین از کاربردهای این نرم افزار می‌باشد.

برای نمونه با استفاده از این نرم‌افزار متن های موجود در این عکس از زبان عربی به انگلیسی ترجمه شده‌است.

2- TapTap See

به منظور تشخیص اشیاء و افراد موجود در عکس برای افراد کم‌بینا و نابینا می‌توان از این نرم‌افزار استفاده کرد. این نرم‌افزار متنی شامل اشیاء و افراد تشخیص داده شده در عکس را توسط صدا به فرد نابینا اعلام می‌کند.

برای نمونه با استفاده از این نرم‌افزار اشیاء موجود در عکس تشخیص داده شده‌است: