

هدف	ادغام پیشینه	ادغام میانگین
هر عمل ادغام مقدار بیشینه‌ی نمای فعلی را انتخاب می‌کند	هر عمل ادغام مقدار میانگین نمای فعلی را انتخاب می‌کند	
نگاره		
توضیحات	– ویژگی‌های شناسایی شده را حفظ می‌کند – اغلب مورد استفاده قرار می‌گیرد	– کاستن نگاشت ویژگی – در (معماری) LeNet استفاده شده است

راهنمای کوتاه شبکه‌های عصبی پیچش

اقتین عمیدی و شروین عمیدی

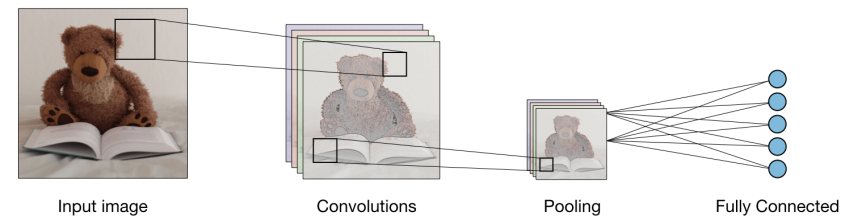
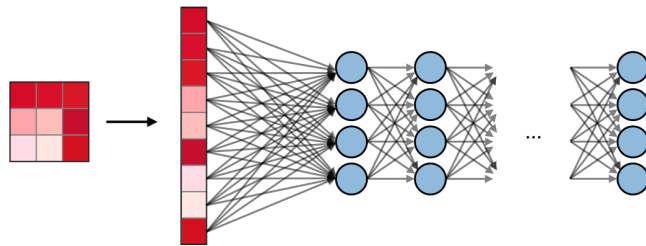
۱۵ شهریور ۱۳۹۸

ترجمه به فارسی توسط الیستر و احسان کرمانی. بازبینی شده توسط عرفان نوری.

نمای کلی

□ **تمام متصل (FC)** – لایه‌ی تمام‌متصل (FC) بر روی یک ورودی سطح به طوری که هر ورودی به تمامی نرون‌ها متصل است، عمل می‌کند. در صورت وجود، لایه‌های FC معمولاً در انتهای معماری‌های CNN یافت می‌شوند و می‌توان آن‌ها را برای بهینه‌سازی اهدافی مثل امتیازات کلاس به کار برد.

□ **معماری یک CNN سنتی** – شبکه‌های عصبی مصنوعی پیچشی، که همچنین با عنوان CNN شناخته می‌شوند، یک نوع خاص از شبکه‌های عصبی هستند که عموماً از لایه‌های زیر تشکیل شده‌اند:

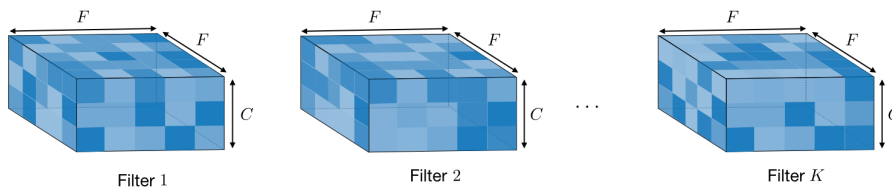


ابرفراسنج‌های فیلتر

لایه‌ی کانولوشنی و لایه‌ی ادغام می‌توانند به نسبت ابرفراسنج‌هایی که در بخش‌های بعدی بیان شده‌اند تنظیم و تعدیل شوند.

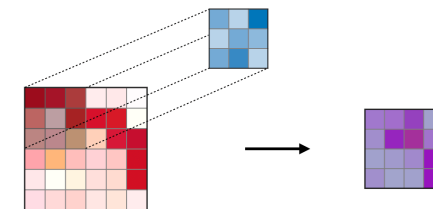
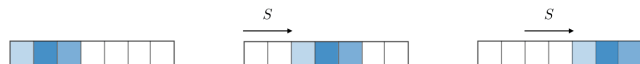
لایه کانولوشنی شامل فیلترهایی است که دانستن مفهوم نهفته در فراسنج‌های آن اهمیت دارد.

□ **ابعاد یک فیلتر (filter)** – یک فیلتر به اندازه $F \times F$ اعمال شده بر روی یک ورودی حاوی C کانال، یک توده $F \times F \times C$ است (عملیات) پیچشی بر روی یک ورودی به اندازه $I \times I \times C$ اعمال می‌کند و یک نگاشت ویژگی خروجی (که همچنین نگاشت فعال‌سازی نامیده می‌شود) به اندازه $O \times O \times 1$ تولید می‌کند.



نکته: اعمال K فیلتر به اندازه‌ی $F \times F$ ، منتج به یک نگاشت ویژگی خروجی به اندازه $O \times O \times K$ می‌شود.

□ **گام (stride)** – در یک عملیات ادغام یا پیچشی، اندازه گام S به تعداد پیکسل‌هایی که پنجره بعد از هر عملیات جابه‌جا می‌شود، اشاره دارد.



نکته: مرحله کانولوشنی همچنین می‌تواند به موارد یک بُعدی و سه بُعدی تعمیم داده شود.

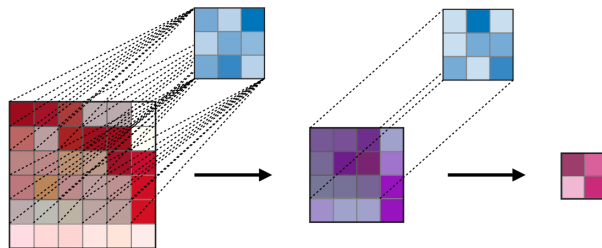
□ **لایه ادغام (POOL)** – لایه ادغام (POOL) یک عمل نمونه‌گاهی است، که معمولاً بعد از یک لایه کانولوشنی اعمال می‌شود، که تا حدی منجر به ناوردایی مکانی می‌شود. به طور خاص، ادغام بیشینه و میانگین انواع خاص ادغام هستند که به ترتیب مقدار بیشینه و میانگین گرفته می‌شود.

FC	POOL	CONV	نگاره
N_{in}	$I \times I \times C$	$I \times I \times C$	اندازه ورودی
N_{out}	$O \times O \times C$	$O \times O \times K$	اندازه خروجی
$(N_{in} + 1) \times N_{out}$	0	$(F \times F \times C + 1) \cdot K$	تعداد فراسنج‌ها
<ul style="list-style-type: none"> ورودی مسطح شده است یک پیش‌قدر به ازای هر نورون تعداد نورون‌های FC فاقد محدودیت‌های ساختاری است 	<ul style="list-style-type: none"> عملیات ادغام به صورت کانال به‌کانال انجام می‌شود در بیشتر موارد $S = F$ است 	<ul style="list-style-type: none"> یک پیش‌قدر به ازای هر فیلتر در بیشتر موارد $S < F$ یک انتخاب رایج برای $K, 2C$ است 	ملاحظات

□ **ناحیه تاثیر (receptive field)** – ناحیه تاثیر در لایه k محدوده‌ای از ورودی $R_k \times R_k$ است که هر پیکسل k – ام نگاشت ویژگی می‌تواند 'ببیند'. با ذکر F_j به عنوان اندازه فیلتر لایه j و S_j مقدار گام لایه i و با این توافق که $S_0 = 1$ است، ناحیه تاثیر در لایه k فرمول زیر محاسبه می‌شود:

$$R_k = 1 + \sum_{j=1}^k (F_j - 1) \prod_{i=0}^{j-1} S_i$$

در مثال زیر داریم، $F_1 = F_2 = 3$ و $S_1 = S_2 = 1$ که منتج به $R_2 = 1 + 2 \cdot 1 + 2 \cdot 1 = 5$ می‌شود.



توابع فعال‌سازی پرکاربرد

□ **تابع یکسوساز خطی (Rectified Linear Unit)** – تابع یکسوساز خطی (ReLU) یک تابع فعال‌سازی g است که بر روی تمامی عناصر توده اعمال می‌شود. هدف آن ارائه (رفتار) غیرخطی به شبکه است. انواع آن در جدول زیر به‌صورت خلاصه آمده‌اند:

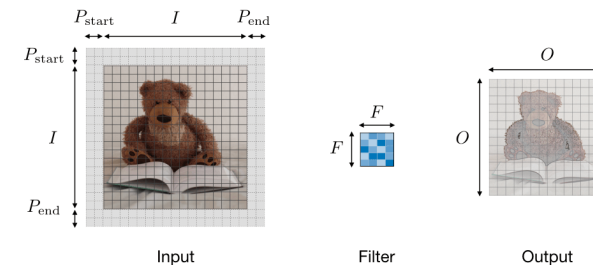
□ **حاشیه صفر (zero-padding)** – حاشیه صفر به فرآیند افزودن P صفر به هر طرف از کرانه‌های ورودی اشاره دارد. این مقدار می‌تواند به طور دستی مشخص شود یا به طور خودکار به سه روش زیر تعیین گردد:

Full	Same	Valid	مقدار
$P_{start} \in [0, F - 1]$ $P_{end} = F - 1$	$P_{start} = \left\lfloor \frac{S \lceil \frac{I}{S} \rceil - I + F - S}{2} \right\rfloor$ $P_{end} = \left\lceil \frac{S \lceil \frac{I}{S} \rceil - I + F - S}{2} \right\rceil$	$P = 0$	
			نگاره
<ul style="list-style-type: none"> – (اعمال) حاشیه به طوری که اندازه نگاشت ویژگی $\lceil \frac{I}{S} \rceil$ باشد – (محاسبه) اندازه خروجی به لحاظ ریاضیاتی آسان است – همچنین حاشیه‌ی 'نیمه' نامیده می‌شود 	<ul style="list-style-type: none"> – فاقد حاشیه – اگر ابعاد مطابقت ندارند آخرین کانولوشنی را رها کن 		هدف

تنظیم ابرفراسنج‌ها

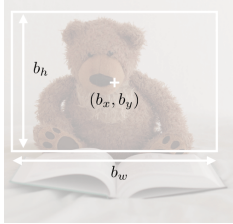
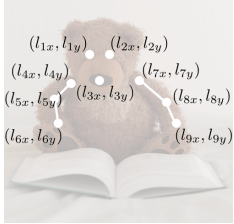
□ **سازش‌پذیری فراسنج در لایه کانولوشنی** – با ذکر I به عنوان طول اندازه توده ورودی، F طول فیلتر، P میزان حاشیه صفر، S گام، اندازه خروجی نگاشت ویژگی O در امتداد ابعاد خواهد بود:

$$O = \frac{I - F + P_{start} + P_{end}}{S} + 1$$



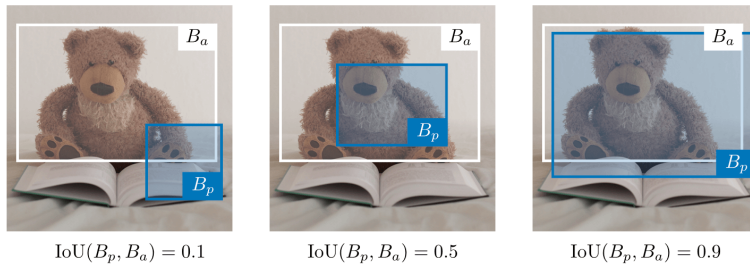
نکته: اغلب $P_{start} = P_{end} \triangleq P$ است، در این صورت $P_{start} + P_{end}$ را می‌توان با $2P$ در فرمول بالا جایگزین کرد.

□ **درک پیچیدگی مدل** – برای برآورد پیچیدگی مدل، اغلب تعیین تعداد فراسنج‌هایی که معماری آن می‌تواند داشته باشد، مفید است. در یک لایه مفروض شبکه پیچشی عصبی این امر به صورت زیر انجام می‌شود:

پیش‌بینی کادر محصورکننده	شناسایی نقاط (برجسته)
بخشی از تصویر که شیء در آن قرار گرفته را شناسایی می‌کند	– یک شکل یا مشخصات یک شیء (مثل چشم‌ها) را شناسایی می‌کند – موشکافانه‌تر
	
مرکز کادر (b_x, b_y) ، ارتفاع b_h و عرض b_w	نقاط مرجع $(l_{1x}, l_{1y}), \dots, (l_{nx}, l_{ny})$

□ **نسبت هم‌پوشانی اشتراک به اجتماع (Intersection over Union)** – نسبت هم‌پوشانی اشتراک به اجتماع، همچنین به عنوان IoU شناخته می‌شود، تابعی است که میزان موقعیت دقیق کادر محصورکننده B_p نسبت به کادر محصورکننده حقیقی B_a را می‌سنجد. این تابع به‌صورت زیر تعریف می‌شود:

$$\text{IoU}(B_p, B_a) = \frac{B_p \cap B_a}{B_p \cup B_a}$$

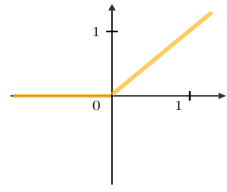
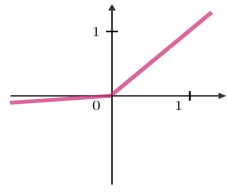
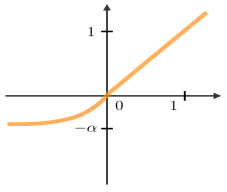


نکته: همواره داریم $\text{IoU} \in [0, 1]$. به صورت قرارداد، یک کادر محصورکننده B_p را می‌توان نسبتاً خوب در نظر گرفت اگر $\text{IoU}(B_p, B_a) \geq 0.5$ باشد.

□ **کادرهای محوری (anchor boxes)** – کادر بندی محوری روشی است که برای پیش‌بینی کادرهای محصورکننده هم‌پوشان استفاده می‌شود. در عمل، شبکه این اجازه را دارد که بیش از یک کادر به‌صورت هم‌زمان پیش‌بینی کند جایی‌که هر پیش‌بینی کادر مقید به داشتن یک مجموعه خصوصیات هندسی مفروض است. به عنوان مثال، اولین پیش‌بینی می‌تواند یک کادر مستطیلی با قالب خاص باشد حال آنکه کادر دوم، یک کادر مستطیلی محوری با قالب هندسی متفاوتی خواهد بود.

□ **فروداشت غیربیشینه (non-max suppression)** – هدف روش فروداشت غیربیشینه، حذف کادرهای محصورکننده هم‌پوشان تکراری دسته یکسان با انتخاب معرف‌ترین‌ها است. بعد از حذف همه کادرهایی که احتمال پیش‌بینی پایین‌تر از 0.6 دارند، مراحل زیر با وجود آنکه کادرهایی باقی می‌مانند، تکرار می‌شوند:

- **گام اول:** کادر با بالاترین احتمال پیش‌بینی را انتخاب کن
- **گام دوم:** هر کادری که $\text{IoU} \geq 0.5$ نسبت به کادر پیشین دارد را رها کن

ReLU	Leaky ReLU	ELU
$g(z) = \max(0, z)$	$g(z) = \max(\epsilon z, z)$ $\epsilon \ll 1$ یا	$g(z) = \max(\alpha(e^z - 1), z)$ $\alpha \ll 1$ یا
		
پهچیدگی‌های غیر خطی که از دیدگاه زیبستی قابل تفسیر هستند	مسئله افول ReLU برای مقادیر منفی را مهار می‌کند	در تمامی نقاط مشتق‌پذیر است

□ **بیشینه‌ی هموار (softmax)** – مرحله بیشینه‌ی هموار را می‌توان به عنوان یک تابع لجستیکی تعمیم داده شده که یک بردار $x \in \mathbb{R}^n$ را از ورودی می‌گیرد و یک بردار خروجی احتمال $p \in \mathbb{R}^n$ ، به‌واسطه‌ی تابع بیشینه‌ی هموار در انتهای معماری، تولید می‌کند. این تابع به‌صورت زیر تعریف می‌شود:

$$p = \begin{pmatrix} p_1 \\ \vdots \\ p_n \end{pmatrix} \quad \text{با} \quad p_i = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}}$$

شناسایی شیء

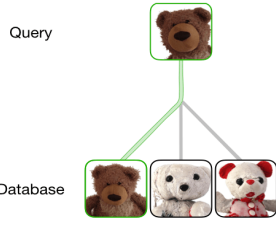
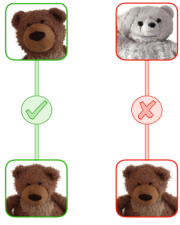
□ **انواع مدل** – سه نوع اصلی از الگوریتم‌های بازشناسایی وجود دارد، که ماهیت آنچه‌که شناسایی شده متفاوت است. این الگوریتم‌ها در جدول زیر توضیح داده شده‌اند:

دسته‌بندی تصویر	دسته‌بندی با موقعیت‌یابی	شناسایی
		
یک عکس را دسته‌بندی می‌کند احتمال شیء را پیش‌بینی می‌کند	– یک شیء را در یک عکس شناسایی می‌کند – احتمال یک شیء و موقعیت آن را پیش‌بینی می‌کند	– چندین شیء در یک عکس را شناسایی می‌کند – احتمال اشیاء و موقعیت آنها را پیش‌بینی می‌کند
CNN سنتی	YOLO ساده شده، R-CNN	R-CNN، YOLO

□ **شناسایی (detection)** – در مضمون شناسایی شیء، روشهای مختلفی بسته به اینکه آیا فقط می‌خواهیم موقعیت قرارگیری شیء را پیدا کنیم یا شکل پیچیده‌تری در تصویر را شناسایی کنیم، استفاده می‌شوند. دو مورد از اصلی‌ترین آنها در جدول زیر به‌صورت خلاصه آورده شده‌اند:

تایید چهره و بازشناسایی

□ انواع مدل – دو نوع اصلی از مدل در جدول زیر به صورت خلاصه آورده شده اند :

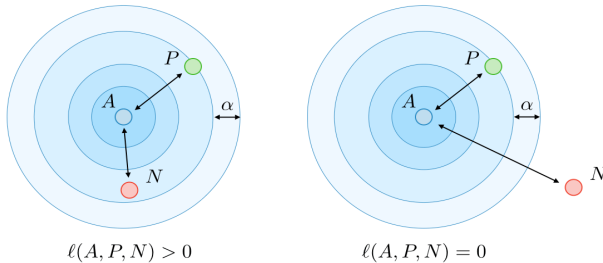
بازشناسایی چهره	تایید چهره
– این فرد یکی از K فرد پایگاه داده است؟ – جستجوی یک به چند	– فرد مورد نظر است؟ – جستجوی یک به یک
<p>Query</p>  <p>Database</p>	<p>Query</p>  <p>Reference</p>

□ **یادگیری یک باره‌ای (One Shot Learning)** – یادگیری یک باره‌ای یک الگوریتم تایید چهره است که از یک مجموعه آموزشی محدود برای یادگیری یک تابع مشابهت که میزان اختلاف دو تصویر مفروض را تعیین می‌کند، بهره می‌برد. تابع مشابهت اعمال شده بر روی دو تصویر اغلب با نماد $d(\text{image 1}, \text{image 2})$ نمایش داده می‌شود.

□ **شبکه‌ی Siamese** – هدف شبکه‌ی Siamese یادگیری طریقه رمزنگاری تصاویر و سپس تعیین اختلاف دو تصویر است. برای یک تصویر مفروض ورودی $x^{(i)}$ ، خروجی رمزنگاری شده اغلب با نماد $f(x^{(i)})$ نمایش داده می‌شود.

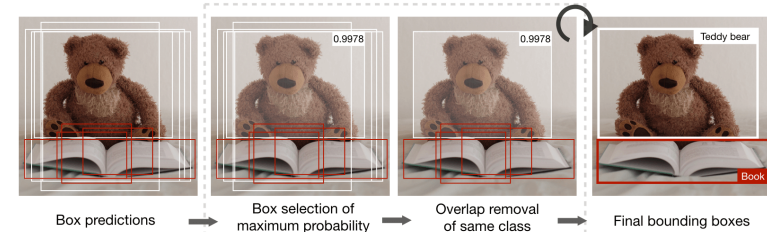
□ **خطای سه گانه (triplet loss)** – خطای سه گانه ℓ یک تابع خطا است که بر روی بازنمایی تعبیه‌ی سه گانه‌ی تصاویر A محور (anchor) مثبت P ، (anchor) منفی N و محاسبه می‌شود. نمونه‌های محور (anchor) و مثبت به دسته یکسانی تعلق دارند، حال آنکه نمونه منفی به دسته دیگری تعلق دارد. با نامیدن $\alpha \in \mathbb{R}^+$ (به عنوان) فراسنج حاشیه، این خطا به صورت زیر تعریف می‌شود :

$$\ell(A, P, N) = \max(d(A, P) - d(A, N) + \alpha, 0)$$



انتقال سبک عصبی

□ **انگیزه** – هدف انتقال سبک عصبی تولید یک تصویر G بر مبنای یک محتوای مفروض C و سبک مفروض S است.



□ **YOLO** – « شما فقط یک بار نگاه می‌کنید » (You Only Look Once, YOLO) یک الگوریتم شناسایی شیء است که مراحل زیر را اجرا می‌کند :

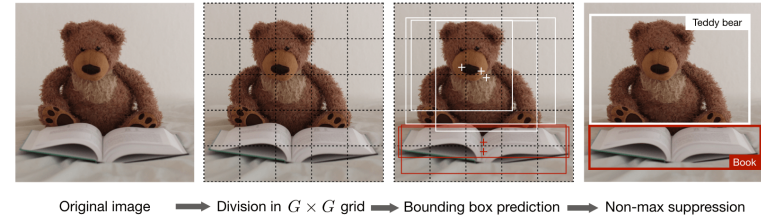
• **گام اول** : تصویر ورودی را به یک مشبک $G \times G$ تقسیم کن

• **گام دوم** : برای هر سلول مشبک، یک CNN که y را به شکل زیر پیش‌بینی می‌کند، اجرا کن :

$$y = \left[\underbrace{p_c, b_x, b_y, b_h, b_w, c_1, c_2, \dots, c_p}_{\text{repeated } k \text{ times}} \right]^T \in \mathbb{R}^{G \times G \times k \times (5+p)}$$

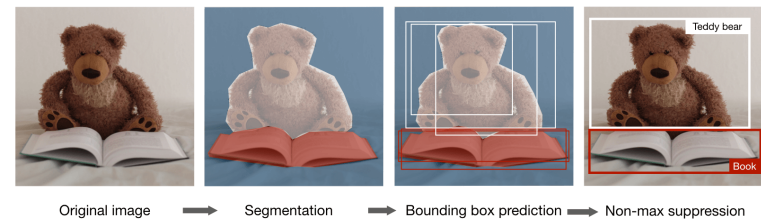
که p_c احتمال شناسایی یک شیء است، b_x, b_y, b_h, b_w اندازه‌های نسبی کادر محیطی شناسایی شده است، c_1, \dots, c_p نمایش «تک‌فعال» یک دسته از p دسته که تشخیص داده شده است، و k تعداد کادرهای محوری است.

• **گام سوم** : الگوریتم فروداشت غیربیشینه را برای حذف هر کادر محصورکننده هم‌پوشان تکراری بالقوه، اجرا کن.

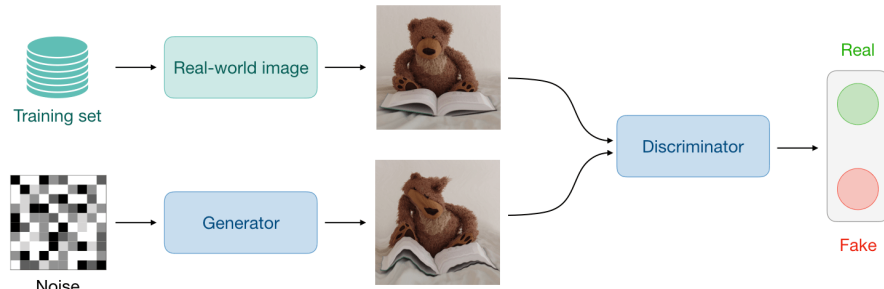


نکته : زمانی که $p_c = 0$ است، شبکه هیچ شیئی را شناسایی نمی‌کند. در چنین حالتی، پیش‌بینی‌های متناظر b_x, \dots, c_p بایستی نادیده گرفته شوند.

□ **R-CNN** – ناحیه با شبکه‌های عصبی پیچشی (Region with Convolutional Neural Networks, R-CNN) یک الگوریتم شناسایی شیء است که ابتدا تصویر را برای یافتن کادرهای محصورکننده مربوط بالقوه قطعه‌بندی می‌کند و سپس الگوریتم شناسایی را برای یافتن محتمل‌ترین اشیاء در این کادرهای محصور کننده اجرا می‌کند.



نکته : هرچند الگوریتم اصلی به لحاظ محاسباتی پرهزینه و کند است، معماری‌های جدید از قبیل *Fast R-CNN* و *Faster R-CNN* باعث شدند که الگوریتم سریعتر اجرا شود.



نکته : موارد استفاده متنوع GAN ها شامل تبدیل متن به تصویر، تولید موسیقی و تلفیقی از آنهاست.

ResNet – معماری شبکه‌ی پسماند (همچنین با عنوان ResNet شناخته می‌شود) از بلاک‌های پسماند با تعداد لایه‌های زیاد به منظور کاهش خطای آموزش استفاده می‌کند. بلاک پسماند معادله‌ای با خصوصیات زیر دارد :

$$a^{[l+2]} = g(a^{[l]} + z^{[l+2]})$$

شبکه‌ی Inception – این معماری از ماژول‌های inception استفاده می‌کند و هدفش فرصت دادن به (عملیات) کانولوشنی مختلف برای افزایش کارایی از طریق تنوع بخشی ویژگی‌ها است. به طور خاص، این معماری از ترفند کانولوشنی 1×1 برای محدود سازی بار محاسباتی استفاده می‌کند.

★ ★ ★



فعال سازی (activation) – در یک لایه مفروض l ، فعال سازی با $a^{[l]}$ نمایش داده می‌شود و به ابعاد $n_H \times n_w \times n_c$ است

تابع هزینه‌ی محتوا (content cost function) – تابع هزینه‌ی محتوا $J_{\text{content}}(C, G)$ برای تعیین میزان اختلاف تصویر تولیدشده G از تصویر اصلی C استفاده می‌شود. این تابع به صورت زیر تعریف می‌شود :

$$J_{\text{content}}(C, G) = \frac{1}{2} \|a^{[l](C)} - a^{[l](G)}\|^2$$

ماتریس سبک (style matrix) – ماتریس سبک $G^{[l]}$ یک لایه مفروض l ، یک ماتریس گرم (Gram) است که هر کدام از عناصر $G^{[l]}_{kk'}$ میزان همبستگی کانال‌های k و k' را می‌سنجند. این ماتریس نسبت به فعال سازی‌های $a^{[l]}$ به صورت زیر محاسبه می‌شود :

$$G^{[l]}_{kk'} = \sum_{i=1}^{n_H} \sum_{j=1}^{n_w} a^{[l]}_{ijk} a^{[l]}_{ijk'}$$

نکته : ماتریس سبک برای تصویر سبک و تصویر تولید شده، به ترتیب با $G^{[l]}(S)$ و $G^{[l]}(G)$ نمایش داده می‌شوند.

تابع هزینه‌ی سبک (style cost function) – تابع هزینه‌ی سبک $J_{\text{style}}(S, G)$ برای تعیین میزان اختلاف تصویر تولیدشده G و سبک S استفاده می‌شود. این تابع به صورت زیر تعریف می‌شود :

$$J_{\text{style}}^{[l]}(S, G) = \frac{1}{(2n_H n_w n_c)^2} \|G^{[l]}(S) - G^{[l]}(G)\|_F^2 = \frac{1}{(2n_H n_w n_c)^2} \sum_{k, k'=1}^{n_c} \left(G^{[l]}_{kk'}(S) - G^{[l]}_{kk'}(G) \right)^2$$

تابع هزینه‌ی کل – تابع هزینه‌ی کل به صورت ترکیبی از توابع هزینه‌ی سبک و محتوا تعریف شده است که با فراسنج‌های α, β ، به شکل زیر وزن دار شده است :

$$J(G) = \alpha J_{\text{content}}(C, G) + \beta J_{\text{style}}(S, G)$$

نکته : مقدار بیشتر α مدل را به توجه بیشتر به محتوا و مقدار بیشتر β مدل را به توجه بیشتر به سبک وا می‌دارد.

معماری‌هایی که از ترفندهای محاسباتی استفاده می‌کنند

شبکه‌ی هم‌آورد مولد (Generative Adversarial Network) – شبکه‌ی هم‌آورد مولد، همچنین با نام GAN شناخته می‌شوند، ترکیبی از یک مدل مولد و تمیزدهنده هستند، جایی که مدل مولد هدفش تولید واقعی‌ترین خروجی است که به (مدل) تمیزدهنده تغذیه می‌شود و این (مدل) هدفش تفکیک بین تصویر تولیدشده و واقعی است.