۱ تحلیل معماری‌های نوین در برابر Faster R-CNN

معماری Faster R-CNN به عنوان یکی از پیشگامان تشخیص‌دهنده‌های دومرحله‌ای (Two-Stage Detector)، استاندارد بالایی را در دقت تشخیص اشیاء بنا نهاده است. این معماری از یک شبکه پیشنهاد ناحیه (RPN) برای استخراج کادرهای احتمالی و سپس از یک شبکه مجزا برای دسته‌بندی و تنظیم دقیق این کادرها استفاده می‌کند. با این حال، نیازهای روزافزون به پردازش سریع‌تر منجر به ظهور معماری‌های یک‌مرحله‌ای (Single-Stage) نظیر خانواده YOLO و رویکردهای مبتنی بر ترانسفورمر نظیر DETR شده است. در ادامه، این معماری‌ها را از سه منظر اساسی تحلیل و مقایسه می‌کنیم:

۱.۱ سرعت استنتاج (Inference Speed) و پیاده‌سازی بلادرنگ

در سیستم‌های نیازمند پردازش ویدیوی بلادرنگ (نظیر خودروهای خودران یا دوربین‌های نظارتی)، سرعت استنتاج یک عامل حیاتی است:

- معماری Faster R-CNN: به دلیل ماهیت دومرحله‌ای و متوالی بودن عملیات (ابتدا استخراج پیشنهادات توسط RPN و سپس اعمال RoI Pooling و طبقه‌بندی روی هر ناحیه)، سرعت استنتاج نسبتاً پایینی دارد. این ساختار معمولاً به سختی می‌تواند به نرخ فریم‌های بالا (High FPS) برای پردازش ویدیوی بلادرنگ دست یابد و بیشتر برای پردازش‌های آفلاین یا کاربردهایی که دقت بر سرعت ارجحیت دارد، مناسب است.
- خانواده YOLO (You Only Look Once): این مدل‌ها مسئله تشخیص را به عنوان یک رگرسیون پیوسته (Single Regression Problem) می‌بینند و مختصات کادرها (Bounding Boxes) و کلاس‌ها را در یک عبور (Forward Pass) از شبکه پیش‌بینی می‌کنند. به همین دلیل، سرعت استنتاج در YOLO بسیار بالاست و گزینه‌ای بی‌رقیب برای پیاده‌سازی‌های بلادرنگ روی دستگاه‌های لبه (Edge Devices) محسوب می‌شود.
- مدل DETR (DEtection TRansformer): این معماری با حذف عملیات دستی و مبتنی بر قانون نظیر NMS (Non-Maximum Suppression) و Anchor Generation، پایپ‌لاین تشخیص را ساده کرده است. با این حال، به دلیل پیچیدگی محاسباتی مربعیِ مکانیزم توجه (Self-Attention) در ترانسفورمرها، سرعت استنتاج آن در نسخه‌های پایه کمتر از YOLO است، اما می‌تواند با سخت‌افزار مناسب به سرعت‌های قابل قبولی دست یابد.

۲.۱ دقت در تشخیص اشیاء کوچک (Small Object Detection)

یافتن اشیاء کوچک و متراکم در تصویر یکی از چالش‌برانگیزترین بخش‌های بینایی ماشین است:

- معماری Faster R-CNN: در این زمینه عملکرد فوق‌العاده‌ای دارد. استفاده از Anchorهای متنوع در اندازه‌ها و نسبت‌های مختلف و همچنین ارزیابیِ مجدد و متمرکز در مرحله دوم، باعث می‌شود این مدل روی اشیاء کوچک با دقت بالایی عمل کند (به ویژه اگر با Feature Pyramid Network یا FPN ترکیب شود).
- خانواده YOLO: در نسخه‌های اولیه (مثل YOLOv۱ تا v۳) به شدت در تشخیص اشیاء ریز مشکل داشتند، زیرا پیش‌بینی‌ها به یک شبکه گرید (Grid Cell) محدود می‌شد. اگرچه در نسخه‌های نوین (مانند YOLOv۸ یا YOLO۱۰) با استفاده از معماری‌های چندمقیاسه (Multi-scale) این ضعف تا حد زیادی پوشش داده شده است، اما در سناریوهای با چگالیِ بالا از اشیاء ریز، همچنان ممکن است رقابت را به Faster R-CNN واگذار کنند.

- مدل DETR: ترانسفورمرهای استاندارد به دلیل هزینه بالای محاسبات، معمولاً روی نقشه‌های ویژگی (Feature Maps) با رزولوشن پایین کار می‌کنند. این موضوع باعث می‌شود مدل پایه DETR در تشخیص اشیاء کوچک ضعیف‌تر از مدل‌های کانولوشنی عمل کند. (البته این مشکل در نسخه‌های پیشرفته‌تر نظیر Deformable DETR با تمرکز توجه روی نقاط خاص حل شده است). با این حال، DETR در درک زمینه سراسری تصویر (Global Context) و تشخیص اشیاء بزرگ بسیار توانمند است.

۳.۱ جمع‌بندی و نتیجه‌گیری راهبردی

انتخاب میان این سه معماری کاملاً وابسته به سناریوی پیاده‌سازی است:

۱. اگر کاربرد شما نیازمند دقت مطلق به ویژه در اشیاء ریز است (مانند تصویربرداری پزشکی یا تحلیل تصاویر ماهواره‌ای) و سرعت دغدغه اصلی نیست، Faster R-CNN همچنان یک استاندارد طلایی است.

۲. اگر هدف سیستم‌های بلادرنگ، ردیابی ویدیویی و پیاده‌سازی روی سخت‌افزارهای محدود است، خانواده YOLO بهترین توازن را بین سرعت استثنایی و دقت قابل قبول ارائه می‌دهد.

۳. اگر به دنبال حذف پیش‌پردازش‌ها و پس‌پردازش‌های دستی (مانند NMS) و یافتن روابط پیچیده بین اشیاء در تصویر هستید، DETR و معماری‌های مبتنی بر ترانسفورمر آینده‌ی این حوزه را شکل می‌دهند.

۲ تحلیل چالش‌های تغییر توزیع داده (Domain Shift) و رویکردهای سازگاری دامنه

هنگامی که یک مدل تشخیص‌دهنده شیء روی مجموعه داده‌ای ساختاریافته نظیر PROPS آموزش می‌بیند و سپس در محیط واقعی (با نویز سنسور، تغییرات شدید روشنایی، و شرایط جوی متفاوت) مورد استفاده قرار می‌گیرد، با پدیده‌ای به نام تغییر توزیع داده (Domain Shift) مواجه می‌شود. این تغییرات باعث می‌شوند شبکه‌های استخراج ویژگی (Backbone) نقشه‌های ویژگیِ متفاوتی نسبت به فاز آموزش تولید کنند. در نتیجه، شبکه تولید پیشنهاد ناحیه (RPN) که مستقیماً از این نقشه‌ها تغذیه می‌کند، در یافتن کادرهای اولیه (Object Proposals) دچار افت شدید عملکرد می‌شود.

برای رفع این مشکل و حفظ پایداری RPN، رویکردهای سازگاری دامنه (Domain Adaptation) در سطح ویژگی (Feature-level) زیر پیشنهاد می‌گردند:

۱.۲ ۱. هم‌ترازی ویژگی‌ها مبتنی بر یادگیری تخاصمی (Adversarial Feature Alignment)

یکی از قدرتمندترین روش‌ها برای غلبه بر Domain Shift، استفاده از شبکه‌های عصبی متخاصم (DANN) در سطح استخراج‌گر ویژگی است.

- نحوه عملکرد: یک طبقه‌بند دامنه (Domain Classifier) به موازات RPN به خروجیِ شبکه استخراج ویژگی متصل می‌شود. وظیفه این طبقه‌بند تشخیص این است که ویژگی استخراج‌شده متعلق به دامنه مبدأ (PROPS) است یا دامنه مقصد (دنیای واقعی). با استفاده از یک لایه معکوس‌کننده گرادیان (Gradient Reversal Layer - GRL)، شبکه استخراج‌گر ویژگی جریمه می‌شود تا ویژگی‌هایی تولید کند که طبقه‌بند نتواند دامنه آن‌ها را تشخیص دهد.
- تأثیر روی RPN: این فرآیند شبکه را مجبور می‌کند ویژگی‌های مستقل از دامنه (Domain-Invariant Features) استخراج کند (مثلاً تمرکز بر شکل هندسی اشیاء به جای بافت یا نورپردازی محیط). در نتیجه، RPN همیشه با یک توزیع استاندارد از ویژگی‌ها روبرو شده و از افت عملکرد مصون می‌ماند.

۲.۲ ۲. نرمال‌سازی و تطبیق آماری ویژگی‌ها (Feature Statistics Matching)

تغییرات روشنایی و نویز سنسور عموماً خود را در میانگین (Mean) و واریانس (Variance) نقشه‌های ویژگی نشان می‌دهند.

- نحوه عملکرد: می‌توان از تکنیک‌هایی نظیر نرمال‌سازی نمونه‌ای تطبیقی (AdaIN - Adaptive Instance Normalization) یا به حداقل رساندن فاصله حداکثر میانگین (MMD - Maximum Mean Discrepancy) در سطح لایه‌های عمیقِ شبکه استفاده کرد. این روش‌ها توزیع آماریِ ویژگی‌های دامنه واقعی را به توزیع ویژگی‌های مجموعه داده PROPS نزدیک می‌کنند.
- تأثیر روی RPN: با هم‌تراز کردن آمار ویژگی‌ها، حساسیت RPN به تغییرات کنتراست و روشنایی محیط از بین رفته و Anchorها با دقت بیشتری روی سوژه‌ها متمرکز می‌شوند.

۳.۲ ۳. استفاده از مکانیزم‌های توجه برای استخراج ویژگی‌های مقاوم (Attention-based Robust Features)

نویز سنسور و شرایط محیطی معمولاً بخش‌های خاصی از تصویر (پس‌زمینه یا بافت‌ها) را بیشتر تحت تأثیر قرار می‌دهند.

- نحوه عملکرد: با ادغام ماژول‌های توجه فضایی و کانالی (مانند CBAM یا مکانیزم‌های Self-Attention) در ساختار Backbone، شبکه یاد می‌گیرد که وزن بیشتری به لبه‌های ساختاری و الگوهای معنایی بدهد و کانال‌های حساس به نویز را سرکوب کند.
- تأثیر روی RPN: این مکانیزم‌ها باعث می‌شوند سیگنال‌های ورودی به RPN از نویزهای فرکانس بالا پاکسازی شوند و پیشنهادهای ناحیه (Proposals) منحصراً روی مناطق دارای محتوای غنی معنایی ایجاد گردند.

۴.۲ ۴. سازگاری مبتنی بر انسجام (معلم - دانش‌آموز) در سطح ویژگی (Consistency Regularization)

در شرایطی که داده‌های دنیای واقعی برچسب ندارند، یادگیری نیمه‌نظارتی (Semi-Supervised) راهگشا است.

- نحوه عملکرد: از یک معماری Mean Teacher استفاده می‌شود. به این صورت که نسخه‌های مختلفی از یک تصویر دنیای واقعی (با نویزهای مصنوعی یا تغییرات نوری افزوده شده) به شبکه داده می‌شود و از مدل خواسته می‌شود تا در فضای ویژگی (Feature Space) و همچنین خروجی‌های RPN پیش‌بینی‌های یکسانی و منسجمی داشته باشد.
- تأثیر روی RPN: این کار باعث می‌شود RPN به مرور زمان نسبت به اغتشاشات اعمال شده (پرتوربیشن‌ها) در سطح ویژگی مقاوم (Robust) شود و ثبات عملکرد خود را در شرایط پیش‌بینی‌نشده حفظ کند.

۵.۲ جمع‌بندی

برای جلوگیری از افت عملکرد RPN در مواجهه با داده‌های نویزی و شرایط متفاوت محیطی، صرفاً تغییر ابعاد Anchorها یا تنظیمات سطحی کافی نیست. پیاده‌سازی سازگاری دامنه در سطح ویژگی (از طریق هم‌ترازی تخاصمی توزیع‌ها یا اصلاح آماری نقشه‌های ویژگی)، پایه‌ای‌ترین روش برای مصون‌سازی RPN در برابر پدیده Domain Shift است که منجر به تعمیم‌پذیری بالای مدل در سناریوهای دنیای واقعی می‌شود.

۳ تحلیل مدیریت عدم توازن پیش‌زمینه و پس‌زمینه و ارزیابی Focal Loss

در معماری‌های تشخیص شیء، به‌ویژه شبکه‌های یک‌مرحله‌ای (Single-Stage Detectors)، شبکه در هر تصویر با ده‌ها هزار کادر پیشنهادی روبرو می‌شود که اکثریت مطلق آن‌ها (بیش از ۹۹ درصد) متعلق به پس‌زمینه (Background) هستند و تنها کسر کوچکی از آن‌ها اشیاء هدف (Foreground) را در بر می‌گیرند. این عدم توازن شدید کلاس‌ها (Foreground-Background Imbalance)، فرآیند آموزش شبکه را با چالش‌های اساسی در محاسبه خطا و به‌روزرسانی وزن‌ها مواجه می‌کند.

۱.۳ مقایسه تحلیلی Cross-Entropy استاندارد در برابر Focal Loss

برای درک چرایی برتری Focal Loss، باید رفتار تابع خطای استاندارد را در مواجهه با داده‌های نامتوازن بررسی کرد:

- تابع خطای Cross-Entropy (CE) استاندارد: در این تابع ($CE(p_t) = -\log(p_t)$)، تمامی نمونه‌ها با وزن برابر در محاسبه خطای کل مشارکت می‌کنند. یک نمونه پس‌زمینه که طبقه‌بند به راحتی و با اطمینان بالا (مثلاً $p_t = 0.95$) آن را تشخیص می‌دهد، خطای بسیار کوچکی تولید می‌کند. اما به دلیل تعداد بسیار زیاد (مثلاً صدهزار عدد) از این نمونه‌های آسان (Easy Examples)، مجموع خطای آن‌ها بر خطای تولید شده توسط تعداد معدود اشیاء هدف غلبه می‌کند. این انباشت خطا باعث می‌شود گرادیان‌های ارسالی به شبکه، به جای تمرکز بر یادگیری ویژگی‌های اشیاء، صرفاً به سمت پیش‌بینی «پس‌زمینه» متمایل شوند.
- تابع خطای Focal Loss: این تابع با افزودن یک عامل تنظیم‌کننده (Modulating Factor) به فرمول استاندارد، فرمت $FL(p_t) = -(1-p_t)^\gamma \log(p_t)$ را شکل می‌دهد. این نوآوری به صورت پویا وزن نمونه‌های آسان را کاهش (Down-weight) می‌دهد. بدین ترتیب، حتی اگر ده‌ها هزار نمونه پس‌زمینه با قطعیت بالا تشخیص داده شوند، سهم آن‌ها در خطای کل و گرادیان نهایی نزدیک به صفر خواهد بود و شبکه مجبور می‌شود به نمونه‌هایی توجه کند که پیش‌بینی آن‌ها دشوار است.

۲.۳ تحلیل عملکرد پارامترهای α و γ بر گرادیانِ نمونه‌های دشوار (Hard Examples)

فرم کامل تابع Focal Loss به صورت $FL(p_t) = -\alpha_t(1-p_t)^\gamma \log(p_t)$ تعریف می‌شود که در آن دو پارامتر α و γ به صورت مکمل عمل کرده و مسیر گرادیان‌ها را مهندسی می‌کنند:

۱. پارامتر α (ضریب متوازن‌کننده کلاس - Class Balancing Parameter):

- نقش تحلیلی: این پارامتر یک ضریب ثابت است که عدم توازن کمیِ کلاس‌ها را جبران می‌کند. معمولاً مقدار α برای کلاس نادر (پیش‌زمینه) بالا و برای کلاس پرشمار (پس‌زمینه) پایین در نظر گرفته می‌شود.
- تأثیر بر گرادیان: پارامتر α به طور خطی دامنه گرادیان‌های مربوط به کلاس‌های مختلف را مقیاس‌دهی می‌کند. این امر تضمین می‌کند که سیگنال‌های آموزشیِ کلاس پیش‌زمینه در اقیانوس سیگنال‌های پس‌زمینه محو نشوند، اما تفاوتی بین نمونه‌های آسان و دشوار در یک کلاس قائل نمی‌شود.

۲. پارامتر γ (ضریب تمرکز - Focusing Parameter):

- نقش تحلیلی: این پارامتر دقیقاً مشکل نمونه‌های آسان در برابر نمونه‌های دشوار (Hard Examples) را هدف قرار می‌دهد. عامل $(1-p_t)^\gamma$ نرخ جریمه را تعیین می‌کند.

- تأثیر بر گرادیانِ نمونه‌های دشوار (Hard Examples): هنگامی که شبکه با یک نمونه دشوار روبرو می‌شود (مثلاً بخشی از پس‌زمینه که شبیه به یک شیء است و شبکه آن را اشتباه تشخیص می‌دهد، در نتیجه p_t کوچک است)، مقدار $(1-p_t)^\gamma$ نزدیک به ۱ خواهد بود. بنابراین، گرادیان این نمونه تقریباً دست‌نخورده باقی مانده و با تمام قدرت برای آپدیت وزن‌ها ارسال می‌شود. در مقابل، برای نمونه‌های آسان ($p_t \approx 1$)، این عامل به شدت به صفر میل می‌کند و گرادیان آن‌ها را سرکوب می‌نماید.

۳.۳ جمع‌بندی تأثیرات در شبکه‌های تشخیص‌دهنده

با ترکیب α و γ، تابع Focal Loss ساختار فضای گرادیان‌ها را بازنویسی می‌کند. با سرکوب شدن گرادیان‌های بی‌ارزشِ ناشی از میلیون‌ها نمونه آسان پس‌زمینه (توسط γ) و متوازن شدن سهم کلاس‌ها (توسط α)، گرادیان‌های ناشی از نمونه‌های دشوار (Hard Negatives و اشیاء پنهان یا مبهم) به حاکم مطلق فرآیند Backpropagation تبدیل می‌شوند. این مهندسی هوشمندانه به شبکه‌های یک‌مرحله‌ای اجازه می‌دهد تا بدون نیاز به مکانیزم‌های پیچیده نمونه‌برداری مانند OHEM (Online Hard Example Mining)، به دقتی هم‌تراز با شبکه‌های دومرحله‌ای دست یابند.

۴ تحلیل کارایی Depthwise Separable Convolutions در دستگاه‌های لبه (Edge Devices)

در سیستم‌های تشخیص شیء (Object Detection) که قرار است روی دستگاه‌های لبه با محدودیت‌های شدید حافظه، باتری و توان پردازشی (مانند موبایل‌ها، پهپادها و دوربین‌های هوشمند) پیاده‌سازی شوند، معماری شبکه استخراج‌کننده ویژگی (Backbone) نقش حیاتی دارد. جایگزینی کانولوشن‌های استاندارد با کانولوشن‌های جداپذیر عمقی (Depthwise Separable Convolutions) که پایه و اساس شبکه‌هایی نظیر MobileNet است، این چالش را با تجزیه یک کانولوشن استاندارد به دو مرحلهِ مجزا (یک کانولوشن مکانیِ Depthwise و یک کانولوشن ترکیبِ کانالی Pointwise یا 1×1) حل می‌کند.

در ادامه، تأثیر این جایگزینی را بر سه فاکتور اساسیِ مطرح‌شده تحلیل می‌کنیم:

۱.۴ ۱. تأثیر بر تعداد پارامترها (Number of Parameters)

- در کانولوشن استاندارد: یک فیلتر مکانی (مثلاً $D_k \times D_k$) همزمان روی تمام کانال‌های ورودی (M) اعمال می‌شود تا کانال‌های خروجی (N) را تولید کند. تعداد پارامترها برابر است با: $D_k \times D_k \times M \times N$.
- در Depthwise Separable Convolution: عملیات به دو بخش تجزیه می‌شود. ابتدا فیلترهای مکانی فقط روی هر کانال به صورت جداگانه اعمال می‌شوند ($D_k \times D_k \times M$) و سپس ترکیب خطی آن‌ها توسط فیلترهای 1×1 انجام می‌گیرد ($1 \times 1 \times M \times N$).
- تحلیل میزان کاهش: نسبت پارامترهای جدید به قدیم برابر است با $\frac{1}{N} + \frac{1}{D_k^2}$. اگر از یک فیلتر استاندارد 3×3 استفاده کنیم، تعداد پارامترها تقریباً به $\frac{1}{9}$ تقلیل می‌یابد (کاهش نزدیک به ۹۰ درصدی). این امر باعث کاهش چشمگیر حجم مدل در حافظه (Model Size) می‌شود که برای دستگاه‌های لبه بسیار حیاتی است.

۲.۴ ۲. تأثیر بر حجم عملیات ممیز شناور (FLOPs)

شاخص FLOPs مستقیماً با مصرف انرژی و سرعت استنتاج (Inference Speed) روی پردازنده‌های ضعیف‌تر در ارتباط است.

- تحلیل عملیاتی: از آنجا که تعداد عملیات ضرب و جمع (MACs) در لایه‌های کانولوشنی رابطه‌ای مستقیم با ابعاد تصویر خروجی ($D_F \times D_F$) و تعداد پارامترها دارد، محاسبه FLOPs نیز با همان ضریبِ کاهشِ پارامترها افت می‌کند.
- تأثیر نهایی: استفاده از این معماری حجم محاسبات را ۸ تا ۹ برابر کاهش می‌دهد. این افت شدیدِ بار پردازشی به سیستم‌های لبه اجازه می‌دهد تا شبکه‌های عمیق‌تری را در زمان واقعی (Real-time) پردازش کنند، بدون آنکه دچار افت فریم (Frame Drop) یا تولید حرارت بیش از حد در پردازنده شوند.

۳.۴ ۳. تأثیر بر میدان دید پذیرنده (Receptive Field)

یکی از مهم‌ترین نگرانی‌ها در فشرده‌سازی شبکه‌های تشخیص شیء، از دست رفتن توانایی شبکه در «دیدن» فضای کافی از تصویر برای تشخیص اشیاء بزرگ است.

- نحوه محاسبه میدان دید: میدان دید پذیرنده به اندازه فیلترهای مکانی ($D_k \times D_k$)، تعداد لایه‌ها و گام‌های حرکت (Strides) بستگی دارد.
- تأثیر نهایی: از آنجا که در Depthwise Separable Convolutions مرحله استخراج ویژگی‌های مکانی (Depthwise) همچنان از فیلترهایی با همان ابعاد (مثلاً 3×3) استفاده می‌کند و استرایدها نیز تغییری نمی‌کنند، میدان دید پذیرنده شبکه (نظری و مؤثر) کاملاً دست‌نخورده باقی می‌ماند و برابر با معماری استاندارد است. این ویژگی استثنایی تضمین می‌کند که شبکه همچنان می‌تواند اطلاعات متنی (Contextual Information) را در مقیاس‌های بزرگ برای یافتن اشیاء حفظ کند.

۴.۴ جمع‌بندی

جایگزینی کانولوشن‌های استاندارد با Depthwise Separable Convolutions یک مبادله (Trade-off) بسیار هوشمندانه است. در ازای افت بسیار اندکی در ظرفیت بازنمایی شبکه (Representational Capacity - که معمولاً با افزایش جزئی عرض شبکه جبران می‌شود)، ما به کاهش ۹۰ درصدی در مصرف حافظه (پارامترها) و بار محاسباتی (FLOPs) دست می‌یابیم، در حالی که میدان دید پذیرنده (Receptive Field) شبکه کاملاً حفظ می‌شود. این مهندسی دقیق، پایه‌گذار اصلی حضور مدل‌های پیشرفته بینایی ماشین در گوشی‌های هوشمند و اینترنت اشیاء (IoT) است.