ششمین همایش بین‌المللی افق های نویـن در
مهندسی برق، کامپیوتـر و مکانیک
6th International Conference on the New Horizons in
Electrical Engineering, Computer and Mechanical
www.mhconf.ir

تشخیص اشیا با استفاده از شبکه های عصبی کانولوشن

حجت رئیسی(نویسنده مسئول)[۱]، شیوا بنی هاشمی[۲]، مسعود کاوسی[۳]، فاطمه حسن نژادیان فرد شیرازی[۴]، محمد لهراسبی[۵]

[۱] رئیس اداره طراحی و توسعه سامانه های هوشمند شهرداری شیراز ، شیراز hojjat.raisee@gmail.com

[۲] کارشناس حمل و نقل شهرداری شیراز ، شیراز shivabanihashemi@gmail.com

[۳] کارشناس اداره طراحی و توسعه سامانه های هوشمند شهرداری شیراز ، شیراز masoud@live.com

[۴] کارشناس منطقه شش شهرداری شیراز ، شیراز fatemeh.shirazi68@yahoo.com

[۵] کارشناس اداره طراحی و توسعه سامانه‌های هوشمند شهرداری شیراز، شیراز mohammad.lohrasbi69@gmail.com

چکیده

در این مقاله، نشان می دهیم که تغییر الگوریتم (طرح های محاسباتی با استفاده از شبکه عمیق) منجر به یک حل موثر و قابل توجهی می شود که محاسبه طرح در محاسبه شبکه تشخیص، تقریبا بدون هزینه (زمان پردازش اندک) می باشد. در انتها، شبکه های طرح منطقه جدیدی (RPNs) را معرفی می کنیم که لایه های کانولوشن را با شبکه های تشخیص اشیای جدید و پیشترفته، به اشتراک می گذارد. با اشتراک لایه ها در زمان آزمایش (زمان تست)، زمان محاسبه طرح ها، کم می باشد (درحدود ۱۰ میلی ثانیه بر تصویر).

مشاهدات ما نشان می دهد که ویژگی کانولوشن استفاده شده با استفاده از تشخیص دهندگان مبتنی بر منطقه، مانند R-CNN سریع، می تواند در ایجاد طرح های منطقه مورد استفاده قرار گیرد. در بالاترین لایه این ویژگی های کانولوشن، ما RPNs را با اضافه کردن دو لایه کانولوشن اضافی ایجاد می کنیم. این لایه ها عبارتند از: لایه ای که هر موقعیت کانولوشن را در بردارهای کوتاه تر رمزگذاری می کند و لایه ای که در هر موقعیت کانولوشن، امتیاز اشیا و مرزهای مربوطه شان را برای طرح های منطقه k مربوط به مقیاس های متنوع و ضریب منظرهایی (زاویه های دید) در آن مکان (موقعیت) به عنوان خروجی تعیین می کند یک مقدار معمول است.

بنابراین RPNs پیشنهادی نوعی از شبکه کانولوشن کامل است که به خصوص برای ایجاد طرح های تشخیص می تواند بصورت انتها به انتها آموزش ببیند. به منظور یکپارچه کردن شبکه های تشخیص اشیای RPNs با R-CNN سریع، روش آموزش ساده ای را پیشنهاد می دهیم که بین تنظیم بهینه طرح منطقه و سپس تنظیم بهینه تشخیص اشیا تغییر می کند؛ این امر درحالی اتفاق می افتد که طرح ها را بصورت ثابت حفظ می کند. این روش بطور سریع همگرا می شود و شبکه ای یکنواخت با ویژگی هایی که بین هر دو روش به اشتراک گذاشته می شود را ایجاد می کند.

واژه‌های کلیدی

شبکه عمیق، تشخیص اشیا، شبکه کانولوشن

۱. مقدمه

اخیرا مزایای بدست آمده در تشخیص اشیا بر اساس موفقیت روش های طرح های منطقه (نظیر مقاله[22]) و شبکه های عصبی کانولوشن مبتنی بر منطقه [6] (R-CNNs) بوده است. اگرچهCNN های مبتنی بر منطقه بطور گسترده و وسیعی در مقاله [۶] توسعه یافته اند، اما هزینه (زمان راه اندازی) شان بطور قابل توجهی با استفاده از اشتراک کانولوشن در میان طرح ها [۷,۵] کاهش یافته است. پیشترفته ترین روش، R-CNNسریع [۵]، به نرخ هایی نزدیک به زمان واقعی با استفاده از شبکه های بسیار عمیق، (وقتی که زمان گذرانیده شده در طرح ها نادیده گرفته می شود) دست یافته است .

روش های طرح منطقه معمولا به روش های استنتاجی اقتصادی و ویژگی های معقول و کم هزینه وابسته است. روش جستوجوی انتخابی[22] (SS) ، یکی از محبوبترین روش هایی است که بطور حریصانه ای سوپرپیکسل های مبتنی بر ویژگی های سطح پایین مهندسی شده را استخراج می نماید. هنوز هم در مقایسه با شبکه های موثر تشخیص ، سرعت روش جستوجوی انتخابی (SS) به اندازه ۲ ثانیه بر تصویر در بکارگیری زمان پردازش CPU کمتر است. اخیرا باکسهای لبه [۲۴] شرایط بهینه ای را بین کیفیت و سرعت (۰,۲ ثانیه بر تصویر) ارائه می دهند. با این وجود، هر گام روش طرح منطقه، زمان پردازشی به اندازه شبکه تشخیص صرف می نماید.

به منظور ارزیابی روش پیشنهادی، از پایگاه داده PASCAL VOC [۴] استفاده نموده ایم که نشان می دهد روش ترکیبی RPNs با R-CNNs دقت تشخیص بالاتری را نسبت به روش ترکیبی جستوجوی تصادفی با R-CNNs سریع ارائه می دهد. ضمنا روش این پیشنهادی تقریبا تمام موانع محاسباتی روش SS را در مرحله آزمایش (تست) رفع می کند. در روش پیشنهادی، زمان راه اندازی موثر برای طرح ها تنها برابر با ۱۰ میلی ثانیه است. با استفاده از مدل های بسیار بسیار عمیق [۱۹]، روش تشخیص پیشنهادی مان دارای نرخ فریم ۵ fps (شامل تمام گام ها) در GPU بوده که بنابراین از دو نقطه نظر سرعت و دقت، یک سیستم تشخیص اشیای عملی و مناسب محسوب می گردد.

۲. سوابق تحقیق

برخی از مقالات اخیر در زمینه روش های ارائه شده مبتنی بر شبکه های عمیق تعیین مکان یک کلاس خاص یا باکس های مرزی کلاس های مخصوص بوده است [21.18.3.20]. در روش OverFast [18]، لایه اتصال کامل (fc[1]) برای پیش بینی مختصات باکس آموزش داده می شود. سپس لایه fc به یک لایه کانولوشن برای نشخیص اشیای کلاس مخصوص چندگانه تبدیل می شود. مدلهای چند باکسی [3.20] ، طرح های منطقه را از شبکه ای که آخرین لایه fc را بطور همزمان باکس های چندگانه (که برای تشخیص اشیای R-CNN [6] استفاده می شود)، پیش بینی میکند را ایجاد می کنند. شبکه پیشنهادی آنها برای یک تصویر تکی یا برش های تصویر بزرگ چندتایی نظیر ۲۲۴*۲۲۴ بکار می رود [20]. در این مقاله به بررسی دقیق روش پیشنهادیمان در مقایسه با روش های OverFeat و مالتی‌باکس[2] می پردازیم.

روش محاسبه اشتراکی کاتولوشن [18.7.2.5] بطور قابل ملاحظه ای در تشخیص بصری دقیق و موثر مورد توجه قرار گرفته است. مقاله Overfeat [18] ویژگی های کانولوشن را از تصویر اولیه به منظور طبقه بندی، مکانیابی و تشخیص محاسبه می کند. روش SPP[3] [7] بروی ویژگی های کانولوشن به اشتراک گذاشته شده به منظور تشخیص بهینه اشیای مبتنی بر منطقه [7.16] و بخش بندی معنایی [2] پیشنهاد شده است. روش R-CNN سریع [5] آموزش انتها به انتهای ویژگی های کانولوشن به اشتراک گذاشته را مهیا کرده و سرعت و دقت محاسباتی مناسبی را ارائه می دهد.

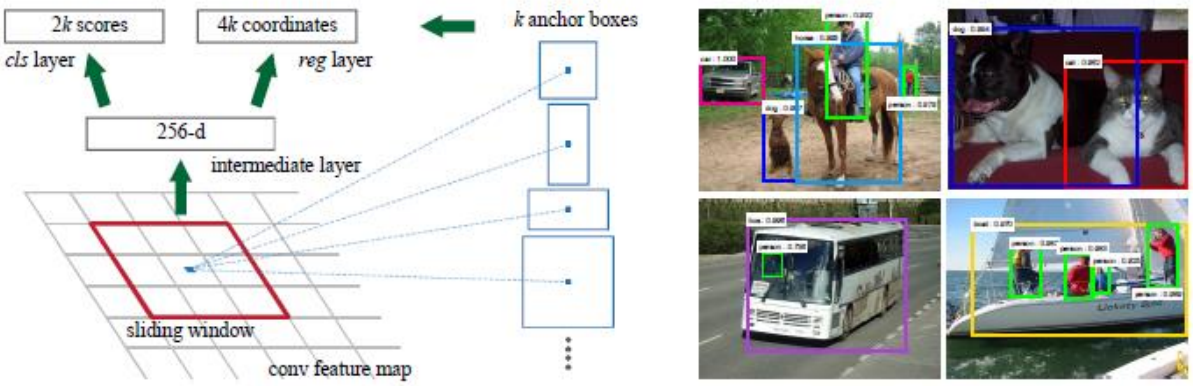
---

[1] fully-connected
[2] MultiBox
[3] Adaptively-sized pooling

۳. شبکه های طرح منطقه

یک شبکه طرح منطقه (RPN) یک تصویر در هر اندازه ای را به عنوان ورودی می گیرد و مجموعه ای از طرح های مستطیلی شیء را با امتیازات مربوط به هر شیء به عنوان خروجی ارائه می دهد. ما این فرآیند را با استفاده از شبکه کانولوشن کامل [] که در این بخش به توضیح آن می پردازیم را مدلسازی می کنیم. از آنجا که خدف نامحدود ما، اشتراک گذاری محاسبات با شبکه تشخیص اشیای R-CNN سریع می باشد، فرض می کنیم که هر دو شبکه مجموعه مشترکی از لایه های کانولوشن را به اشتراک می گذارند. در ازمایشات ما، مدل زِیلِر و فِرگوس [4](ZF) [23] را که دارای ۵ لایه قابل اشتراک و مدل سیمونیان و زیسِرمَن (VGG)[5] [19] که دارای ۱۳ لایه کانولوشن قابل اشتراک می باشند را مورد مطالعه قرار می دهیم.

به منظور ایجاد طرح های منطقه، شبکه کوچکی بروی ویژگی کانولوشن خروجی توسط آخرین لایه کانولوشن به اشتراک گذاشته شده را تولید میکنیم. این شبکه از نوع اتصال کامل[6] است و بروی پنجره فضایی n*n خروجی توسط آخرین لایه کانولوشن به اشتراک گذاشته شده ایجاد می شود. هر پنجره لغزنده[7] به یک بردار کم بُعدتر (256-d برای ZF و 512-d برای VGG) نسبت داده می شود. این بردار در دو لایه اتصال یافته کامل (یک لایه رگرسیون (reg) و یک لایه طبقه بندی (cls)) تغذیه می شود. ما در این مقاله از n=3 استفاده می کنیم؛ توجه شود که میدان پذیرنده موثر بروی تصویر ورودی بسیار بزرگ است (۱۷۱ پیکسل برای ZF و ۲۲۸ پیکسل برای VGG). این شبکه کوچک به عنوان یک موقعیت تکی در شکل ۱ سمت چپ ارائه شده است. توجه شود که از آنجا که شبکه کوچک در یک مُد پنجره لغزنده عمل می کند، لایه هایی که بطور کامل متصل شده اند، در میان تمام مکان های فضایی به اشتراک گذاشته می شوند. این ساختار معمولا بصورت n*n لایه کانولوشن که توسط دو لایه ۱*۱ کاتولوشنی برای لایه های reg و cls تشکیل می شوند، بکار برده می شود. روش ReLUs [15] برای خروجی n*n لایه کانولوشنی بکار برده می شود.



شکل ۱: سمت چپ: شبکه طرح منطقه (RPN). سمت راست: نمونه ای از تشخیص با استفاده از طرح های RPN بروی داده های تست PASCAL VOC 2007. روش مان اشیاء را در طیف گسترده ای از مقیاس ها و ضریب های منظری تشخیص می دهد.

---

[4] Zeiler and Fergus
[5] Simonyan and Zisserman
[6] fully connected
[7] sliding window

۴.لنگرهای تبدیل ثابت [8]

در مکان هر پنجره لغزنده، بطور همزمان طرح های منطقه k را پیش بینی می کنیم، بنابراین لایه reg دارای خروجی های 4k است که مختصات باکس k را پیش بینی می کند. لایه cls امتیازات 2k را که احتمال شیء بودن یا نبودن برای هر طرح را پیش بینی می کند. طرح های k نسبت به باکس های مرجع k پارامتریک می شوند که به آن ها لنگر گفته می شود. هر لنگر در پنجره لغزنده متمرکز شده و بر اساس مقیاس و ضریب منظر، سازماندهی می گردد. ما از ۳ مقیاس و ۳ ضریب منظر استفاده می کنیم که منجر به k=9 لنگر در هر موقعیت لغزنده می شود. برای یک ویژگی کانولوشن با اندازه W*H (معمولا ۲,۴۰۰)، بطور کلی W H لنگر k وجود دارد. نسبت مهمی که روش ما به آن متکی است، "تبدیل ثابت" می باشد که هم در لنگرها و هم در توابعی که طرح های نسبی را نسبت به لنگرها محاسبه می کند، برقرار می باشد.

به منظور مقایسه، روش مالتی‌باکس [20] از روش کا-مینز[9] برای تولید ۸۰۰ لنگر استفاده می کند که "تبدیل ثابت" نیستند. اگر هر لنگر، یک شیء را در یک تصویر شناسایی کند، آن طرح و تابع مشابه باید قادر به پیش بینی طرح در هر مکان باشد. علاوه بر این، از آنجا که لنگرهای مالتی‌باکس بصورت تبدیل ثابت و غیر متنوعی نیستند، این روش به یک لایه خروجی 800*(4+1) بُعدی نیاز دارد؛ این درحالی است که روش پیشنهادی ما نیازمند 9*(4+2) بُعد لایه خروجی می باشد. لایه های پیشنهادی ما دارای پارامترهای کمتر (۲۷ میلیون برای مالتی‌باکس با استفاده از گوگل‌نت[10] [20] و ۲,۴ میلیون برای RPN با استفاده از VGG-16) می باشد، بنابراین دارای ریسک کمتری در زمینه عملکرد مناسب بروی پایگاه داده های کوچک نظیر PASCAL VOC می باشد.

۵.تابع زیان برای آموزش طرح های منطقه

برای آموزش RPNs، یک برچسب کلاس باینری (که بیان کننده شیء بودن یا نبودن می باشد) به هر لنگر اختصاص می دهیم. بدین منظور برچسب مثبت را به دو نوع از لنگرها اختصاص می دهیم: (i) لنگر/لنگرهایی با بالاترین همپوشانی IoU[11] با هر باکس درست؛ یا (ii) هر لنگری که دارای IoU همپوشانی بالاتر از ۰,۷ با هر باکس درست. توجه شود که یک باکس درست تکی ممکن است برچسب های مثبتی را به لنگرهای چندتایی اختصاص دهد. همچنین، برچسب منفی را به لنگری غیر مثبتی اختصاص می دهیم که نسبت IoU اش برای تمام باکس ها کمتر از ۰,۳ باشد. لنگرهایی که نه مثبت هستند و نه منفی، مربوط به اهداف آموزشی نمی باشند.

با این تعاریف، تابع هدف را بر اساس تابع زیان چند منظوره در R-CNN سریع [5] کمینه می کنیم. تابع زیان برای یک تصویر بصورت زیر تعریف می شود:

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}(p_i, p_i^*) + \lambda\frac{1}{N_{reg}}\sum_i p_i^* L_{reg}(t_i, t_i^*). \qquad (1)$$

که در این رابطه i بیانگر شاخص لنگر در یک بخش کوچک و p_i بیان کننده احتمالی است که یک لنگر نشان دهنده یک شیء می باشد. درصورتی که برچسب p_i^* برابر با ۱ باشد، لنگر مثبت است و درصورتی که ۰ باشد، لنگر منفی می باشد. t_i بیان کننده برداری است که ۴ مختصات پارامتریک شده باکس مرزی پیش بینی شده می باشد و t_i^* باکس مرتبط با لنگر مثبت است. در زیان[12] طبقه بندی

[8] Translation-Invariant Anchors
[9] K-Means
[10] GoogleNet
[11] Intersection over-Union
[12] Loss

بندی L_{cls} بیان کننده اختلاف دو کلاس (شیء بودن یا نبودن) می باشد. برای زیان رگرسیون، از $L_{reg}(t_i - t_i^*) = R(t_i - t_i^*)$ که R تابع زیان رُباست تعریف شده در [5] است. عبارت $p_i^* L_{reg}$ بیان کننده زیان رگرسیون است که تنها برای لنگرهای مثبت $(p_i^* = 1)$ بصورت فعال و در غیر اینصورت $(p_i^* = 0)$ غیر فعال می باشد. خروجی های لایه های cls و reg به ترتیب شامل $\{p_i\}$ و $\{t_i\}$ می باشد. این دو عبارت با N_{cls} و N_{reg} نرمال شده و با وزن λ^3 متعادل می شوند.

برای رگرسیون، پارامتریک کردن ۴ مختصات بصورت زیر انجام می شود [6]:

$$t_{\mathrm{x}} = (x - x_{\mathrm{a}})/w_{\mathrm{a}}, \quad t_{\mathrm{y}} = (y - y_{\mathrm{a}})/h_{\mathrm{a}}, \quad t_{\mathrm{w}} = \log(w/w_{\mathrm{a}}), \quad t_{\mathrm{h}} = \log(h/h_{\mathrm{a}}),$$
$$t_{\mathrm{x}}^* = (x^* - x_{\mathrm{a}})/w_{\mathrm{a}}, \quad t_{\mathrm{y}}^* = (y^* - y_{\mathrm{a}})/h_{\mathrm{a}}, \quad t_{\mathrm{w}}^* = \log(w^*/w_{\mathrm{a}}), \quad t_{\mathrm{h}}^* = \log(h^*/h_{\mathrm{a}}),$$

که $x.y.w$ و h بیان کننده دو مختصات مرکز، عرض و ارتفاع باکس می باشند. متغیرهای x x_a و x^* به ترتیب برای باکس، باکس لنگر و باکس ground-truth می باشند. این موضوع می تواند به عنوان رگرسیون باکس مرزی از یک باکس لنگر به نزدیک ترین باکس ground-truth درنظر گرفته شود.

با این وجود، روش ما رگرسیون باکس مرزی را با استفاده از روشی متفاوت از روش های پیشین مبتنی بر ویژگی انجام میدهد[7.5]. در مقالات [7.5]، رگرسیون باکس مرزی بر اساس ویژگی های حاصل شده از مناطق سایزبندی شده اختیاری است و وزن های رگرسیون توسط تمام این اندازه های مناطق به اشتراک گذاشته می شوند. در فرمول بندی مان، ویژگی های استفاده شده برای رگرسیون دارای اندازه فضایی یکسانی (n*n) بروی ویژگی ها هستند. برای درنظرگیری اندازه های متغیر، مجموعه ای از رگرسیون های باکس مرزی k آموزش داده می شوند. هر رگرسیون، برای یک مقیاس و یک ضریب منظر (زاویه دید) درنظر گرفته می شود و رگرسیون های k ، وزنشان را به اشتراک نمی گذارند. با این حال، حتی اگر ویژگی ها دارای مقیاس/اندازه ثابت نیز باشند، همچنان پیش بینی باکس ها با اندازه های مختلف امکان پذیر می باشد.

۶. بهینه سازی

روش RPN که بطور طبیعی به عنوان یک شبکه کانولوشن کامل بکار برده می شود[14]، با استفاده از روش پس انتشار و SGD[۱۳] می تواند بصورت انتها به انتها آموزش ببیند[12]. همچنین می توان از استراتژی نمونه برداری "imagecentric" که از [5] تعیین می شود، این شبکه را آموزش دهیم. هر بخش کوچک، از تصویر جداگانه ای حاصل می شود که شامل تعداد زیادی لنگرهای مثبت و منفی می باشد. بهینه سازی تابع زیان تمام لنگرها نیز امکان پذیر است اما بسته به نمونه های منفی ای دارد که نسبت به نمونه های دیگر غالب هستند. بجای آن، ما از ۲۵۶ لنگر نمونه بصورت تصادفی در یک تصویر به منظور محاسبه تابع زیان بخش کوچک استفاده می کنیم که لنگرهای مثبت و منفی نمونه دارای نسبت ۱:۱ هستند. اگر مقداری کمتر از ۱۲۸ نمونه مثبت در یک تصویر وجود داشته باشد، بخش کوچکی را با نمونه های منفی درنظر می گیریم.

حال تمام لایه های جدید را بصورت تصادفی با ایجاد وزونی از توزیع گوسین حدود صفر با انحراف استاندارد ۰,۰۱ آماده سازی می کنیم. تمام لایه های دیگر (یعنی لایه های کانولوشن به اشتراک گذاشته شده) توسط آموزش مدل برای طبقه بندی ImageNet [17] به

[13] stochastic gradient descent

عنوان یک عمل استاندارد [6] آماده سازی می شوند. ما تمام لایه ها ی شبکه ZF ، conv3_1 و تا شبکه VGG را به منظور حفظ حافظه تنظیم می کنیم[5]. برای این کار از نرخ آموزش ۰,۰۰۱ برای 60 k بخش کوچک، و ۰,۰۰۰۱ برای 20 k نمونه کوچک بعدی بروی پایگاه داده PASCAL استفاده می کنیم. همچنین از مومنتوم ۰,۹ و وزن کاهشی ۰,۰۰۰۵ [11] استفاده می کنیم. راه اندازی نرم افزاری ما از Caffe استفاده می کند [10].

۷. اشتراک گذاری ویژگی های کانولوشن برای تشخیص اشیا و طرح منطقه

بنابراین، جدا از اینکه چگونه یک شبکه برای تولید طرح منطقه آموزش می بیند، بدون درنظرگیری تشخیص شیء مبتنی بر منطقه، روش CNN این طرح ها را آماده سازی می کند. برای شبکه تشخیص، ابتدا R-CNN سریع [5] را پیاده کرده و سپس الگوریتمی را که لایه های کانولوشن را بین RPN و R-CNN سریع به اشتراک میگذارد، توصیف می نماییم.

هر دو روش RPN و R-CNN سریع بطور مستقل آموزش می بینند و لایه های کانولوشن خودشان را از طریق روش های مختلف، اصلاح می نمایند. بنابراین نیاز به توسعه تکنیک و روشی داریم که بجای اینکه هر دو روش بصورت جداگانه آموزش ببینند، بتواند مجال تبادل لایه های کانولوشن بین هر دو شبکه را مهیا نماید. توجه شود که این کار به اندازه تعریف هر شبکه بصورت مجزا که شامل هر دو روش RPN و R-CNN سریع باشد و سپس با استفاده از روش پس انتشار بهینه سازی گردد، آسان نیست. دلیل این موضوع این است که آموزش R-CNN سریع به طرح های ثابت شیء بستگی دارد. در صورتی که آموزش R-CNN سریع بطور همزمان تغییراتی را ایجاد نماید، این امر منجر به همگرایی مکانیزم طرح خواهد شد. از آنجا که این نقطه بهینه یکی از موضوعات مورد توجه برای تحقیقات آینده است، ما از یک الگوریتم آموزشی ۴ مرحله ای برای آموزش ویژگی های به اشتراک گذاشته شده از طریق بهینه سازی جایگزینی استفاده کرده ایم.

در گام اول، همانند آنچه که در بالا توضیح داده شد، RPN را آموزش می دهیم. این شبکه با مدل از پیش آموزش داده شده ImageNet آماده سازی شده و برای استفاده در طرح منطقه بصورت انتها به انتها تنظیم می گرددو در گام دوم، شبکه تشخیص را بصورت جداگانه توسط R-CNN سریع با استفاده از طرح های تولید شده توسط گام اول RPN آموزش می دهیم. این شبکه تشخیص نیز توسط مدل از پیش آموزش دیده شده ImageNet آماده سازی می شود. در این مرحله، این دو شبکه لایه های کاتولوشن را به اشتراک نمی گذارند. در گام سوم، از شبکه تشخیص دهنده برای آماده سازی آموزش RPN استفاده می کنیم، که لایه های اشتراک گذاشته شده را ثابت و تنها لایه هایی که مختص RPN می باشند را تنظیم بهینه می کند. حال، هر دو شبکه لایه های کانولوشن را به اشتراک می گذارند. در نهایت، با حفظ لایه های کاتولوشن به اشتراک گذاشته شده بطور ثابت، لایه های fc روش R-CNN سریع را تنظیم بهینه می کنیم. به همین ترتیب، هر دو شبکه لایه های کانولوشن مشابهی را به اشتراک می گذارند و یک شبکه یکنواخت را تشکیل می دهند.

۸.جزئیات پیاده سازی

برای اینکار، هر دو شبکه های تشخیص شیء و طرح منطقه را برای تصاویر مقیاس تکی آموزش و تست می کنیم [7.5]. همچنین مقیاس تصاویر را به نحوی تغییر می دهیم که کوچکترین بخش کناریشان برابر با s=600 پیکسل شود [5]. اگرچه ممکن است که استخراج ویژگی چند مقیاسه منجر به بهبود دقت و صحت تشخیص گردد، اما هیچگونه بهبودی در زمینه سرعت استخراج را به همراه نخواهد داشت [5]. همچنین باید توجه داشت که در شبکه های ZF و VGG ، گام کلی در آخرین لایه کانولوشن برابر با ۱۶ پیکسل بر اساس تغییر مقیاس تصاویر می باشد، و بنابراین تقریبا برابر با ۱۰ پیکسل بروی تصاویر پایگاه داده PASCAL معمولی $(\sim 500 * 375)$ خواهد بود. اگرچه که دقت تشخیص در گام های کمتر بهتر خواهد بود، اما این گام بزرگتر نیز نتایج مطلوبی را به همراه خواهد داشت.

برای لنگرها، از ۳ مقیاس با مساحت های باکس برابر با 128^2 ,256^2 و 512^2 پیکسل و ۳ ضریب منظر (زاویه دید) برابر با ۱:۱ ، ۱:۲ و ۲:۱ استفاده می کنیم. توجه شود الگوریتم پیشنهادیمان اجازه استفاده از باکس های لنگری را که بزرگتر از میدان پذیرنده اصلی در زمان پیش بینی طرح های بزرگتر هستند را می دهد. این پیش بینی ها غیر ممکن بوده و خیلی دقیق نیستند و تنها ممکن است محدوده ای از شی را تنها در صورتی که وسط شیء قابل روئیت باشد را شناسایی کنند. با این طراحی، راه حل پیشنهادی به ویژگی های چند مقیاسه یا پنجره های لغزنده چند مقیاسه برای پیش بینی مناطق بزرگ نیاز داشته که این امر سبب کاهش قابل توجهی از زمان راه اندازی و پردازش می گردد. شکل ۱ سمت راست، قابلیت روش پیشنهادیمان را برای طیف گسترده ای از مقیاس ها و زوایای دید را نشان می دهد. جدول زیر اندازه طرح میانگین اموزش داده شده را برای هر لنگر با استفاده از شبکه ZF را نشان می دهد. (اعداد برای s=600 می باشد).

anchor	128^2, 2:1	128^2, 1:1	128^2, 1:2	256^2, 2:1	256^2, 1:1	256^2, 1:2	512^2, 2:1	512^2, 1:1	512^2, 1:2
proposal	188×111	113×114	70×92	416×229	261×284	174×332	768×437	499×501	355×715

باکس های لنگر که مرزهای تصویر را برش می دهند نیازمند این می باشند که به درستی مدیریت شوند. در طول مرحله آموزش، تمام لنگرهای برش مرز را به نحوی نادیده می گیریم که منجر به ایجاد زیان در شبکه نشوند. برای یک تصویر معمولی ۶۰۰*۱۰۰۰، معمولا بطور کلی 20k لنگر $(\sim 60 * 40 * 9)$ وجود خواهد داشت. با صرف نظر از لنگرهای برش مرز، حدودا 6k لنگر بر اصویر برای آموزش وجود خواهد داشت. اگر لایه های خروجی برش های مرزی در مرحله آموزش نادیده گرفته شوند، دشواری زیادی در تصحیح خطا بوجود خواهد آمد و مرحله آموزش، همگرا نخواهد شد. در طول مرحله آزمایش نیز، روشRPN کانولوشن کامل را به کل تصویر اعمال می کنیم. این موضوع باکس های طرح برش مرزی ای را ایجاد خواهد نمود که آن را در مرز تصویر در نظر می گیریم.

تعدادی از طرح های RPN بشدت بروی یکدیگر همپوشانی ایجاد می کنند. به منظور کاهش این افزونگی، از روش NMS[۱۴] بروی طرح های منطقه بر اساس امتیازات cls شان استفاده می کنیم. همچنین حد آستانه IoU را برای NMS برابر با ۰,۷ درنظر می گیریم که حدود 2k طرح منطقه بر تصویر را به همراه خواهد داشت. همانطور که نشان خواهیم داد، روش NMS به دقت تشخیص نهایی صدمه وارد نخواهد کرد اما منجر به کاهش تعداد طرح ها خواهد شد. پس از NMS، از تعداد بالای N طرح منطقه برای تشخیص استفاده می کنیم. در ادامه، از روش R-CNN سریع با استفاده از 2kطرح RPN استفاده خواهیم نمود اما تعدادزیادی از طرح ها را بطور همزمان مورد ارزیابی قرار می دهیم.

منابع

[1] N. Chavali, H. Agrawal, A. Mahendru, and D. Batra. Object-Proposal Evaluation Protocol is 'Gameable'.arXiv: 1505.05836, 2015.
[2] J. Dai, K. He, and J. Sun. Convolutional feature masking for joint object and stuff segmentation. In CVPR, 2015.
[3] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. In CVPR, 2014.
[4] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results, 2007.

[14] Non maximum suppression

[5] R. Girshick. Fast R-CNN. arXiv:1504.08083, 2015.
[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, 2014.
[7] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In ECCV. 2014.
[8] J. Hosang, R. Benenson, P. Dollar, and B. Schiele. What makes for effective detection proposals? ´arXiv:1502.05082, 2015.
[9] J. Hosang, R. Benenson, and B. Schiele. How good are detection proposals, really? In BMVC, 2014.
[10] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. arXiv:1408.5093, 2014.
[11] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural net works. In NIPS, 2012.
[12] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backprop agation applied to handwritten zip code recognition. Neural computation, 1989.
[13] K. Lenc and A. Vedaldi. R-CNN minus R. arXiv:1506.06981, 2015.
[14] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In CVPR, 2015.
[15] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In ICML, 2010.
[16] S. Ren, K. He, R. Girshick, X. Zhang, and J. Sun. Object detection networks on convolutional feature maps. arXiv:1504.06066, 2015.
[17] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. arXiv:1409.0575, 2014.
[18] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In ICLR, 2014.
[19] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In ICLR, 2015.
[20] C. Szegedy, S. Reed, D. Erhan, and D. Anguelov. Scalable, high-quality object detection. arXiv:1412.1441v2, 2015.
[21] C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. In NIPS, 2013.
[22] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. IJCV, 2013.
[23] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional neural networks. In ECCV, 2014.
[24] C. L. Zitnick and P. Dollar. Edge boxes: Locating object proposals from edges. In ´ ECCV, 2014.