

تقسیم بندی تصویر پزشکی با Local Context Perception شبکه عصبی عمیق

فهرست مطالب

2	مقدمه
2	هدف
3	آنالیز داده
3	شبکه عصبی کانولوشن
3	شبکه عصبی کانولوشن چیست و چگونه از آن در دسته بندی تصاویر استفاده می شود
3	شیوه عملکرد شبکه عصبی کانولوشن چیست؟
4	لایه اول در شبکه عصبی کانولوشن مفاهیم ریاضیاتی
4	لایه اول در شبکه عصبی کانولوشن
4	لایه های عمیق تر شبکه عصبی کانولوشن
5	لایه آخر در شبکه عصبی پیچشی
5	نحوه عملکرد شبکه عصبی کانولوشن چیست؟
5	تست شبکه عصبی CNN
5	نیاز به بخش بندی تصویر
6	Image Segmentation انواع مختلف
7	Parallel Dilated convolution
8	Local Context Embedding
8	معیارهای ارزیابی
9	دقت Accuracy
9	Precision
9	حساسیت Sensitivity
10	Specificity
10	Mean Intersection over Union = mIoU
10	تابع هزینه
10	تابع هزینه عمومی
11	تابع هزینه Focal Loss (FL)
11	Cross-Entropy Loss (CE)
11	فاصله انحراف (Distance Deviation Cross Loss Function: DDCLoss)
11	نتیجه گیری
12	Reference

مقدمه

تقسیم بندی خودکار تصویر یک مرحله ضروری در تجزیه و تحلیل تصویر پزشکی است و نقش مهمی در رادیوتراپی با کمک رایانه ، تشخیص بیماری و ارزیابی اثر درمان دارد. دشواری تقسیم تصویر پزشکی به دلیل تار بودن تصویر پزشکی ، شکل پیچیده اشیا و وجود نویز بسیار افزایش می یابد. در سال های اخیر ، روش های تقسیم بندی مبتنی بر یادگیری عمیق ، به ویژه شبکه عصبی کانولوشن ، پیشرفت زیادی در بهبود دقت تقسیم بندی تصویر پزشکی داشته است. با این حال ، این روش ها همچنین به دلیل عدم استفاده کافی از اطلاعات زمینه محلی تصاویر در طول فرآیند استخراج ویژگی ، توانایی کمی در تشخیص اشیا مشابه در محیط های مختلف دارند.

برای رفع این مشکل ، این تحقیق یک شبکه عصبی عمیق (LCP-Net) را پیشنهاد می کند که می تواند اطلاعات زمینه ای مقیاس چند عکس را درک کند. LCP-Net از Parallel Dilated Convolution (PDC) و Local Context Embedding (LCE) ، که برای دستیابی به نقشه ویژگی که غنی از اطلاعات محیطی است استفاده می کند ، استفاده از اطلاعات زمینه رمزگذارهای ویژگی را بهبود می بخشد.

علاوه بر این ، برای بهبود دقت تقسیم بندی مدل برای اشیا کوچک و کاهش میزان پیچیدگی در حین آموزش ، ما یک کاهش متقابل (DDCLoss) جدید را پیشنهاد می دهیم ، که می تواند وزن را با توجه به قطعیت و فاصله انحراف مقدار پیکسل را پیش بینی کرده و مدل را قادر می سازد تا با اطمینان روی بهینه سازی نقاط نمونه تمرکز کند نقاطی که دارای برچسب غلط باشد.

هدف

تقسیم بندی تصاویر پزشکی با شبکه عصبی عمیق (LCP-Net) که متشکل از:

Local Context Embedding (LCE) + Parallel Dilated Donvolution (PDC)

توابع برای بالا بردن سطح اطلاعات استخراج شده از تصاویر. بهبود دادن میزان وزن ها در شبکه عصبی مصنوعی مستلزم استفاده از Loss Function ، هزینه کارآمد است. در اینجا از سه نوع تابع هزینه استفاده شده است.

1. تابع هزینه **Focal Loss (FL)**

2. **Cross-Entropy Loss**

3. فاصله انحراف (**Distance Deviation Cross Loss Function: DDCLoss**)

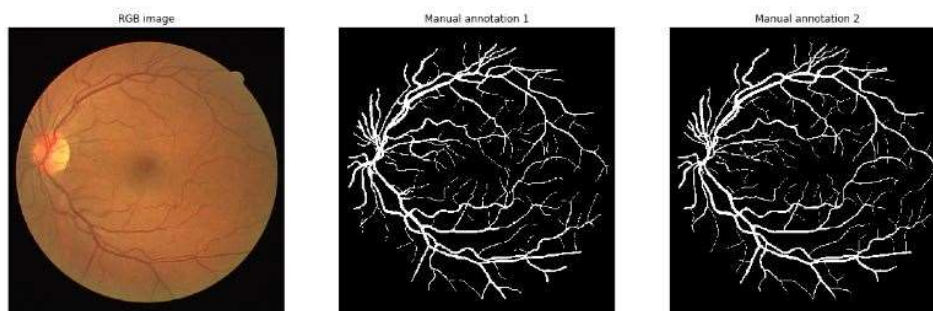
تابع هزینه **DDCLoss**

یک نسخه بهبود یافته از آنتروپی ، بر اساس اطمینان از نتایج پیش بینی و فاصله انحراف است و به عنوان یکی از توابع هزینه کارآمد برای بهینه سازی مدل پیشنهاد می کنیم. عملکرد این تابع با در نظر گرفتن اطمینان از پیکسل های پیش بینی شده و فاصله انحراف از برچسب به عنوان عوامل تنظیم کننده ، وزن پیکسل های نمونه را به صورت کارآمدی تنظیم می کند ، که این عمل ، مدل را قادر می سازد تمرکز بیشتری در بهینه سازی پیکسل های نمونه داشته باشد که دارای برچسب نامناسب یا نامشخص در حین آموزش هستند.

آنالیز داده

مجموعه داده های DRIVE توسط یک بیمارستان محلی تهیه شده است که از توالی MRI شریان کاروتید بیمار مبتلا به تنگی شریان کاروتید گرفته شده است. این مجموعه داده از 196 نمونه تشکیل شده است و هر نمونه شامل یک تصویر ساژیتال از شریان کاروتید و تصویر واقعی مربوط به آن است که به طور دستی توسط پزشکان تصویربرداری حرفه ای تقسیم بندی می شود، هر دو به اندازه 512×512 . منطقه تقسیم بندی شامل لومن شریانی و دیواره خارجی شریان کاروتید که در آن تنگی شریان کاروتید اتفاق می افتد. برای جلوگیری از overfitting بیش از حد مدل به دلیل کمبود نمونه، روش زیر را برای تقویت مجموعه داده اتخاذ می کنیم:

بعد از اینکه یک پیکسل به طور تصادفی بر روی تصویر اصلی انتخاب شد، یک تصویر فرعی به ابعاد 96×96 از تصویر اصلی با پیکسل به عنوان مرکز بریده می شود، و یک منطقه با همان اندازه در موقعیت مربوط به تصویر واقعی تقسیم می شود، تصویر به عنوان زیر تصویر. سپس، تکرار عملیات فوق 250 بار برای هر نمونه، که برابر است با گسترش مجموعه داده اصلی به 250 برابر، رخ خواهد داد. ما به طور تصادفی 80٪ از نمونه های مجموعه داده گسترده را به عنوان مجموعه آموزش و 20٪ باقیمانده را به عنوان مجموعه آزمون انتخاب می کنیم، و در این مجموعه داده، پس از تأیید آزمایشی، همان نرخ های متنوع را به عنوان مجموعه داده کانتور سلول اتخاذ کردیم. ما از مجموعه داده های DRIVE و STARE استفاده می کنیم که به یکی از معیارهای استاندارد در مطالعات تقسیم عروق شبکه تبدیل شده است.



شبکه عصبی کانولوشن

شبکه عصبی پیچشی یا شبکه عصبی کانولوشن، از مهم ترین نوآوری ها در حوزه ی بینایی کامپیوتر به حساب می آیند؛ و بیش ترین استفاده ی شبکه عصبی در پردازش تصویر است.

شبکه عصبی کانولوشن چیست و چگونه از آن در دسته بندی تصاویر استفاده می شود

دسته بندی تصاویر (image classification) در حقیقت فرایندی است که در آن تعدادی تصویر از ورودی می گیریم و در خروجی، کلاس آن ها مثلا (بازال، ملانوما، نووس، درماتوفیروما، کراتوز آکتینیک، کراتوز خوش خیم، ضایعه های عروقی و ...) یا درصد احتمال تعلق به هر کلاس را مشخص می کنیم. وقتی یک CNN تصویری را به عنوان ورودی دریافت می کند، آن را به صورت آرایه ای از اعداد می بیند. تعداد آرایه ها به ساین تصویر (بر اساس پیکسل) بستگی دارد. برای مثال اگر یک تصویر رنگی با فرمت JPG و اندازه 80×80 پیکسل را به CNN دهیم، آرایه جانشین آن دارای $80 \times 80 \times 3$ خانه خواهد بود (عدد 3 به RGB برمی گردد). (هر کدام از خانه های نیز عددی بین 0 تا 255 را می گیرند. این عدد شدت پیکسلی را نشان می دهد. این اعداد در دسته بندی تصاویر با استفاده از شبکه عصبی کانولوشن، نماینگر تصویر به زبان آرایه ها است. ایده اصلی آن است که به CNN آرایه ای از اعداد را می دهیم و CNN نیز در خروجی نام بیمارها را مشخص می کند.

شیوه عملکرد شبکه عصبی کانولوشن چیست؟

برای درک و تشخیص تصویرهای پیچیده ای مثل تصویر یک بیماری پوستی، ابتدا ویژگی های (feature) ساده تر آن تصویر مانند لبه ها و خم ها را تشخیص می دهد. در یک شبکه عصبی، لایه های متعددی وجود دارند؛ در هر یک از این لایه ها، ویژگی های خاصی تشخیص داده می شوند و در نهایت، در لایه ی آخر، تصویر به طور کامل شناسایی می شود. روندی که توضیح دادیم، فرایند کلی نحوه کار یک شبکه عصبی کانولوشن بود؛ حال به جزئیات بیش تری می پردازیم.

لایه اول در شبکه عصبی کانولوشن | مفاهیم ریاضیاتی

لایه اول در یک شبکه عصبی پیچشی، همیشه یک لایه کانولوشن است. همان طور که قبلاً اشاره کردیم، ورودی این لایه یک آرایه از اعداد است. لایه اول در شبکه عصبی مانند یک چراغ قوه کار می کند. در یک اتاق تاریک، چراغ قوه ای را تصور کنید که بر گوشه ای بالا و سمت چپ تصویر می اندازیم و محدوده ای از تصویر روشن می شود و آن قسمت را می بینیم. سپس چراغ قوه را بر روی قسمت های دیگر تصویر می گیریم تا کم کم کل تصویر را ببینیم. در شبکه عصبی کانولوشن، به این (یا نورون یا کرنل) می گوئیم. آن قسمتی از تصویر که چراغ قوه به آن نور می تاباند، محدود پذیرش (filter) چراغ قوه، فیلتر یا (weight) نام دارد. فیلترها نیز خود آرایه هایی از اعداد هستند. به اعداد موجود در فیلتر، وزن (receptive field) گفته می شود. لازم به ذکر است که عمق این فیلتر باید با عمق تصویر برابر باشد. مثلاً اگر تصویر (parameter) پارامتر یک آرایه عددی $5 \times 5 \times 3$ (عمق ۳) است، فیلتر نیز باید چنین باشد. فیلتر در هر نگاه، یک قسمت از تصویر را می بیند. سپس بر روی تصویر حرکت می کند تا قسمت های دیگر را هم اسکن کند. به این حرکت فیلتر بر روی تصویر، پیچیدن می گویند. همین طور که فیلتر از تصویر عبور می کند، اعداد موجود در فیلتر با آرایه عددی پیکسل های (convolve) تصویر ضرب می شود. در نهایت نیز تمام حاصل ضرب ها با یکدیگر جمع می شوند و به یک عدد می رسیم. فرض کنید می خواهیم یک تصویر با ابعاد $32 \times 32 \times 3$ را با استفاده از یک فیلتر $5 \times 5 \times 3$ ببینیم. این فیلتر با عملیاتی که توضیح دادیم در نهایت یک آرایه عددی با ابعاد $28 \times 28 \times 1$ تولید می کند (علت 28×28 بودن آن است که به 784 حالت می توان یک تصویر 32×32 را با استفاده از فیلتر 5×5 دید). به ماتریس $28 \times 28 \times 1$ که در نهایت به دست می آید نقشه فعال سازی می گوئیم. اگر به جای یک فیلتر، از دو فیلتر استفاده کنیم، در (feature map) یا نقشه ویژگی (activation map) نهایت به یک ماتریس با ابعاد $28 \times 28 \times 2$ می رسیم. این کار می تواند دقت ما را در ابعاد بالاتر افزایش دهد.

لایه اول در شبکه عصبی کانولوشن

هر یک از فیلترهایی را که در قسمت قبلی به آن ها اشاره کردیم، می توان به عنوان یک شناساگر ویژگی (feature identifier) در نظر گرفت. منظور از ویژگی (feature) در این جا، چیزهایی مانند خط صاف، یک رنگ ساده یا یک انحناست. فرض کنیم فیلتر اول، یک فیلتر با ابعاد $7 \times 7 \times 3$ و یک شناساگر انحناست. این فیلتر در حقیقت یک ماتریس عددی مانند تصویر زیر است که درایه های این ماتریس در محل هایی که انحنا وجود دارد، مقادیر عددی بالاتری دارند. حال این فیلتر را بر روی قسمتی از تصویر مد نظرمان قرار می دهیم. پس از آن مانند تصویر زیر، درایه به درایه اعداد موجود در خانه ها را با هم ضرب و حاصل ضرب ها را با یکدیگر جمع می کنیم.

همانطور که مشاهده می کنید، حاصل به دست آمده، یک عدد بزرگ است. بزرگ بودن این عدد نشانگر آن است که در این ناحیه یک انحنا مانند انحنای فیلتر وجود دارد.

سپس حاصل ضرب عدد کوچکی می شود؛ علت آن است که فیلتر با تصویر ورودی تطابق ندارد. همان طور که اشاره کردیم، ما در پی آن هستیم که یک نقشه فعال سازی به دست بیاوریم؛ یعنی یک آرایه از اعداد با ابعاد $26 \times 26 \times 1$ (فرض کنیم فقط از یک فیلتر شناساگر منحنی استفاده کنیم). قسمت بالا و سمت چپ این نقشه فعال سازی، مقدار 660 را خواهد داشت. این عدد بزرگ نشان دهنده ای آن است که در ناحیه ای خاصی از تصویر، با احتمال زیاد یک منحنی وجود دارد. توجه کنید که ما در این جا تنها از یک فیلتر استفاده کردیم. برای آن که اطلاعات بیشتری از تصویر استخراج کنیم، نیاز داریم تا از فیلترهای بیش تری استفاده کنیم؛ استفاده از فیلترهای بیش تر یعنی ابعاد بالاتر.

لایه های عمیق تر شبکه عصبی کانولوشن

در یک شبکه عصبی، علاوه بر لایه ای توضیح داده شده، لایه های دیگری نیز وجود دارند. این لایه ها وظایف و عملکردهای گوناگونی دارند. به طور کلی، لایه های داخلی، مسئول نگهداری و حفظ ابعاد و امور غیرخطی هستند. آخرین لایه در شبکه عصبی کانولوشن نیز از اهمیت خاصی برخوردار است.

لایه آخر در شبکه عصبی پیچشی

در لایه آخر یک شبکه عصبی کانولوشن، خروجی سایر لایه‌ها، به عنوان ورودی دریافت می‌شود. خروجی لایه آخر هم یک تعداد کلاس‌های موجود است. به عنوان مثال اگر شبکه‌ی شما، یک شبکه برای شناسایی اعداد N . بعدی است N بردار بعدی، هر مولفه، احتمال وقوع یک کلاس را نشان می‌دهد. N باشد، تعداد کلاس‌ها ده‌تاست؛ چون ده رقم داریم. در بردار کاری که لایه‌ی آخر یک شبکه عصبی کانولوشن می‌کند آن است که به ویژگی‌های لایه‌های سطح بالا نگاه می‌کند و میزان مطابقت این ویژگی‌ها را با هر کلاس مقایسه می‌کند؛ هر چه این مطابقت بیش‌تر باشد، احتمال وقوع آن کلاس، بالاتر معرفی می‌شود.

نحوه عملکرد شبکه عصبی کانولوشن چیست؟

CNN تست شبکه عصبی

پس از آن که مدل ما نهایی و آماده شد، وقت تست کردن فرا می‌رسد. برای تست مدل از تعدادی تصویر که می‌دانیم محتویات آن چیست، استفاده می‌شود. تصویر را به ورودی مدل می‌دهیم تا خروجی را به ما نشان دهد؛ سپس خروجی را بررسی می‌کنیم تا ببینیم درست عمل شده است یا نه.

جستجو در تصویر برای یافتن تنها یک الگو منجر به نتایج خوبی نمی‌شود و باعث می‌شود شبکه از لحاظ کارایی محدود باشد. برای حل این مشکل، نیاز است که لایه کانولوشنی چندین فیلتر داشته باشد. **هریک از فیلترها به تنهایی یک الگوی خاص داشته باشند و خروجی لایه کانولوشنی مجموعه‌ای از الگوهای مختلف باشد.** تصویر با فیلترهای مختلفی کانوالو شده و هر فیلتر هم یک خروجی جداگانه برای خود تولید کرده است. همه تصاویر مربوط به یک چهره با ماسک هست؛ اما دقت کنید که در هر تصویری یک مجموعه ویژگی خاص مورد توجه بوده است. مثلاً در تصویری، لبه‌های افقی، در دیگری لبه‌های عمودی و غیره.

اندازه $stride$ را می‌توان گام هم تعریف کرد. منظور از گام این است که فیلتر بعد از محاسبه در یک پنجره از ورودی، چند درایه یا خانه باید جلو برود تا دوباره محاسبات را انجام دهد. اگر اندازه گام را 1 در نظر بگیریم، یعنی فیلتر باید درایه به درایه در ورودی به سمت راست و پایین برود و اسکن را انجام دهد. یعنی، همان صحبت‌هایی که در بخش قبل (کانولوشن) داشتیم. اما اگر مثلاً $stride$ عدد 2 در نظر گرفته شود.

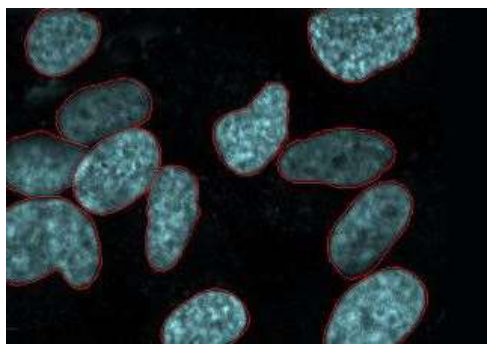
لایه پولینگ یکی دیگر از لایه‌های مهم در شبکه عصبی کانولوشن است. هدف لایه پولینگ کاهش اندازه مکانی فیچر مپ بدست آمده با استفاده از لایه کانولوشنی است. **لایه پولینگ پارامتر قابل آموزش ندارد.** صرفاً یک نمونه برداری ساده و موثر انجام می‌دهد. پولینگ عملکردی شبیه کانولوشن دارد و یک پنجره روی تصویر حرکت می‌کند. رایج‌ترین نمونه پولینگ **max pooling** و **average pooling** است. ماکس پولینگ شامل یک پنجره از پیش تعریف شده است (مثلاً 3×3) که روی تصویر حرکت می‌کند (تصویر را اسکن می‌کند) و در هر پنجره مقدار ماکزیمم را انتخاب می‌کند و بقیه را دور می‌ریزد. مشابه با لایه کانولوشنی، اندازه فیلتر، پدینگ و استراید در اینجا هم صدق می‌کند. **در پولینگ، معمولاً استراید 2 و اندازه فیلتر 3 در نظر گرفته می‌شود.** برای اینکه ساینز ویژگی به $2/1$ کاهش پیدا کند.

معمولاً آخرین لایه‌های یک شبکه عصبی کانولوشن برای طبقه بندی را لایه‌های **فولی کانکتد** تشکیل می‌دهند. این لایه‌ها همان لایه‌هایی هستند که در شبکه عصبی MLP دیده‌اید. برای آشنایی بیشتر این مقاله را بخوانید. یکی از کاربردهای اصلی لایه فولی کانکتد در شبکه کانولوشن، استفاده به عنوان طبقه بند یا **کلاسیفایر (Classifier)** است. یعنی مجموعه ویژگی‌های استخراج شده با استفاده از لایه‌های کانولوشنی در نهایت تبدیل به یک بردار می‌شوند. در نهایت این بردار ویژگی به یک کلاسیفایر فولی کانکتد داده می‌شود تا کلاس درست را شناسایی کند.

نیاز به بخش بندی تصویر

شکل سلول‌های سرطانی در تعیین شدت سرطان نقش اساسی دارد. برای شناسایی سلول‌های سرطانی، شناسایی اشیاء موجود در تصویر خیلی مفید نخواهد بود. در اینجا اگر ما فقط **Bounding Box** ها را شنایی کنیم، نمی‌توانیم شکل سلول‌ها را شناسایی کنیم. تکنیک‌های تقسیم تصویر در اینجا تأثیر بسیار زیادی را برای شناسایی

سلول‌های مورد نظر ایجاد می‌کند. آنها به ما کمک می‌کنند نتایج معنی‌دارتری کسب کنیم و به خواسته خود که شناسایی سلول‌های سرطانی است، دست پیدا کنیم.



در اینجا به وضوح می‌توان شکل تمام سلول‌های سرطانی را مشاهده کرد.

انواع مختلف Image Segmentation



در عکس 1، هر پیکسل متعلق به یک کلاس خاص (یا پس زمینه یا اسب) است. همچنین، تمام پیکسل‌های متعلق به یک کلاس خاص با همان رنگ نمایش داده می‌شوند به عنوان مثال اسب‌ها با رنگ سفید مشخص شده‌اند. این نمونه‌ای از بخش‌بندی معنایی است.

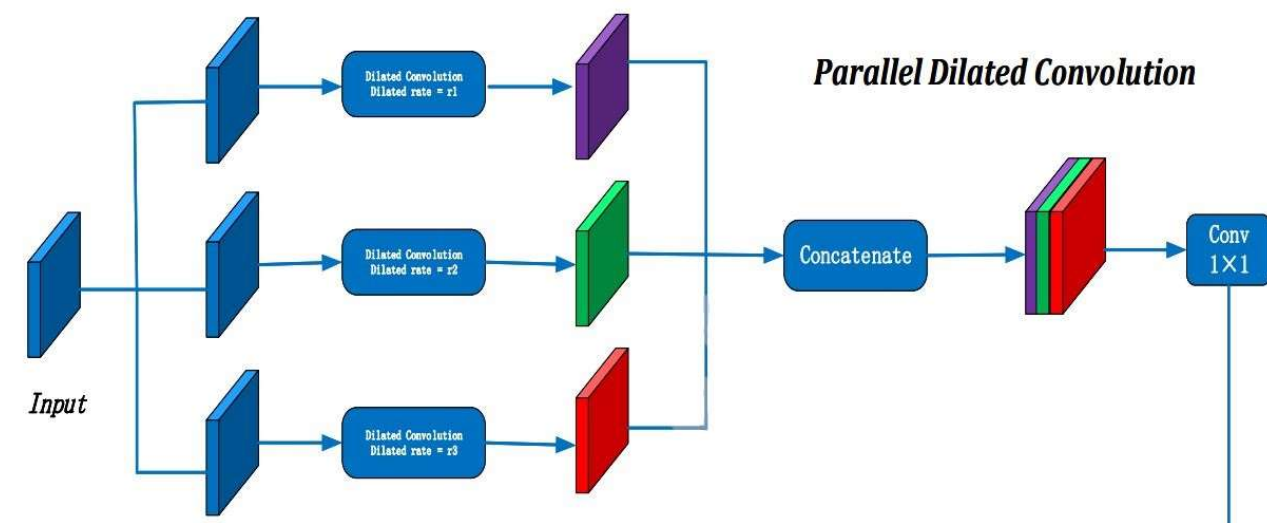
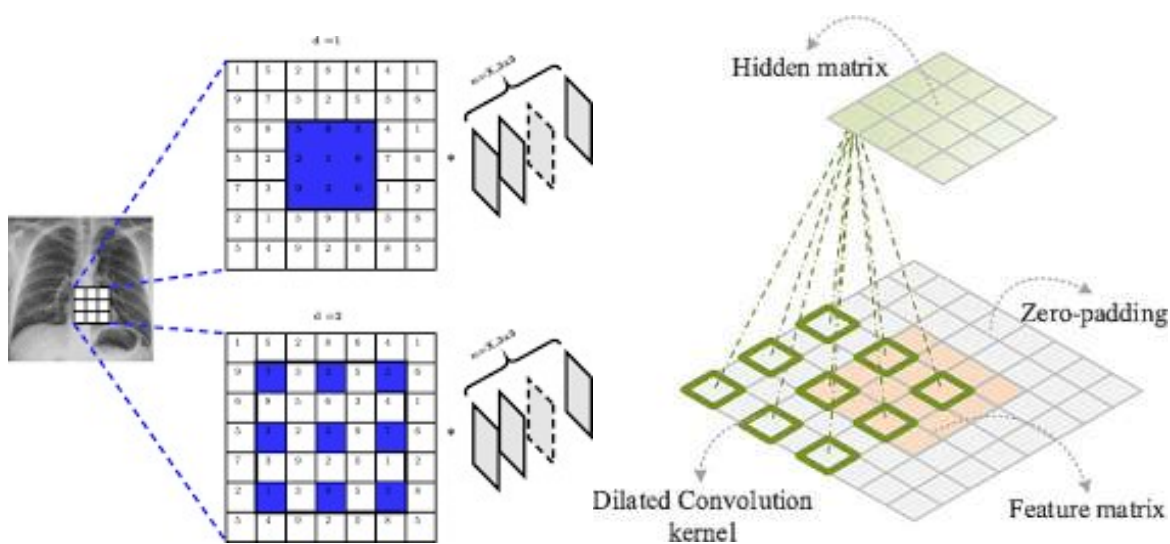
در تصویر 2 کلاس خاصی به هر پیکسل تصویر اختصاص داده شده است. با این حال، اشیاء مختلف در همان کلاس از رنگ‌های مختلف برخوردار هستند. اسب 1 با رنگ قرمز، اسب 2 به رنگ سبز و اسب 3 با رنگ آبی قابل مشاهده است. این نمونه‌ای از تقسیم‌بندی نمونه است.

الگوریتم‌های بخش‌بندی؟

- آستانه‌گیری روشنایی یا بخش‌بندی دامنه
- روش‌های فازی
- روش واترشید یا تبدیل آب پخش
- الگوریتم ژنتیک
- بخش‌بندی تصویر به کمک مینیمم درخت پوشا
- روش‌های مبتنی بر گراف

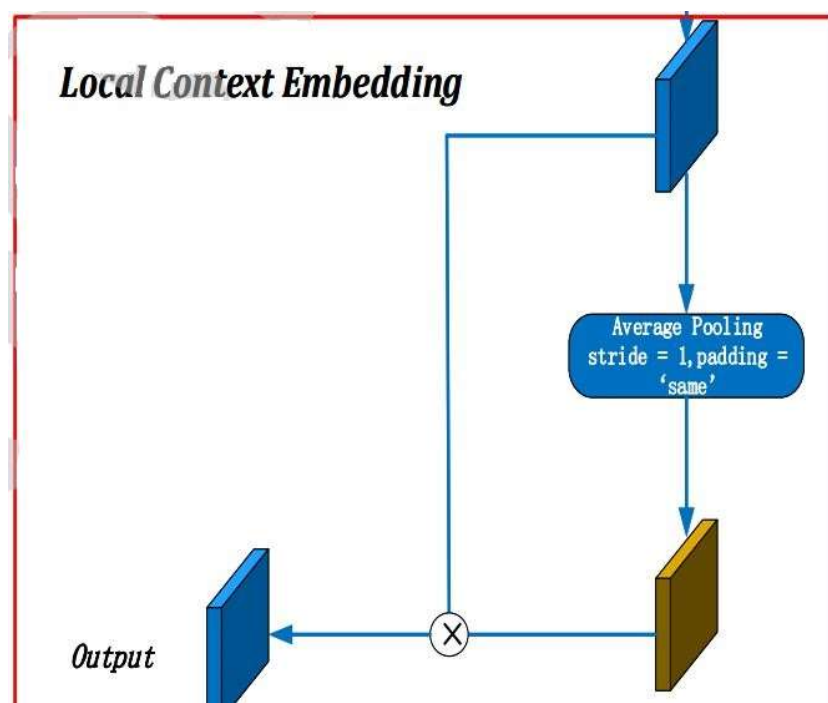
Parallel Dilated convolution

لایه های متسع شده موازی: در این لایه ها ، چرخش های متسع شده برای استخراج معناشناسی با رشد نمایی در اندازه میدان مورد قبول و بدون احتساب ابعاد ، اجرا می شوند.



Local Context Embedding

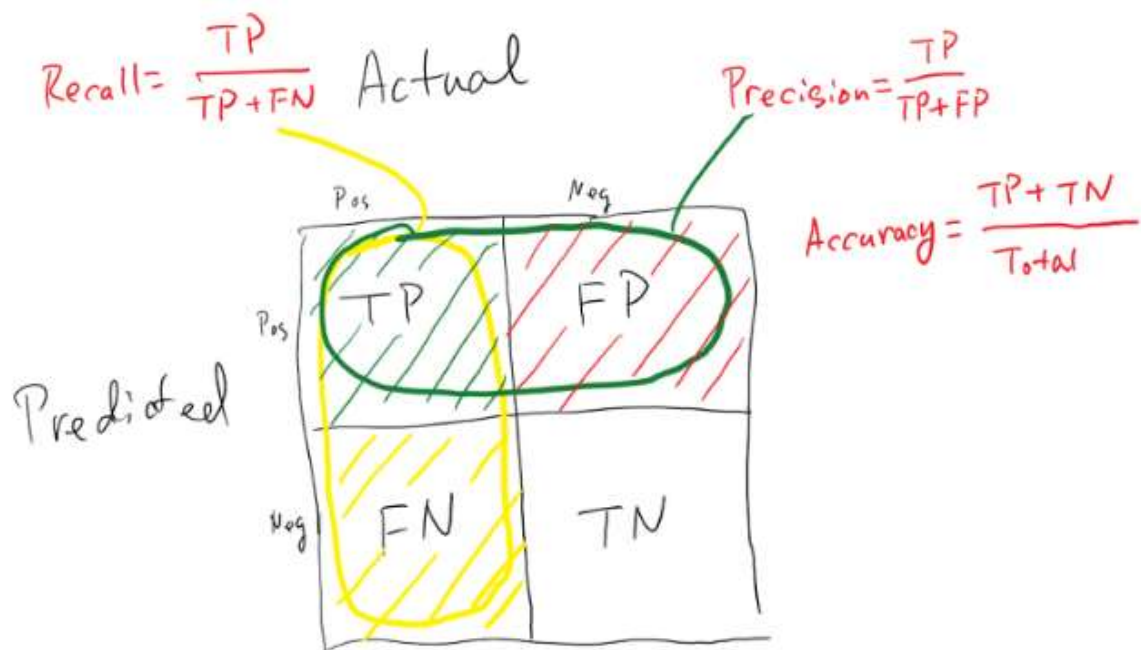
LCE مقادیر دریافتی از PDC را در بعنوان اطلاعات کلی در منطقه ای که پیکسل ها در آن قرار دارند ، جاسازی می کند.



معیارهای ارزیابی

1. معیارهایی که ما برای ارزیابی عملکرد مدل استفاده می کنیم ، که به ترتیب به شرح زیر تعریف شده اند:
2. شاخص تاس (Dice) ،
3. میانگین تقاطع (mIoU) ،
4. دقت (Acc) ،
5. ویژگی (Spe) ،
6. دقت (Pre) و
7. حساسیت (Sen) ،
8. Mean Intersection over Union = mIoU

در نهایت بعد از هر تحلیلی نتایج به دست آمده را باید در چهار گروهی که در بالا توصیف شد، دسته بندی کرد تا بتوان کیفیت تحلیل را مورد ارزیابی قرار داد. همچنین کارایی الگوریتم را برای موارد استفاده مختلف تفسیر و توصیف نمود. در ارزیابی نتایج معمولاً معیارهای زیر متصور است و بیشترین استفاده را نسبت به معیارهای ارزیابی دیگر دارد



دقت Accuracy

به طور کلی، دقت به این معناست که مدل تا چه اندازه خروجی را درست پیش‌بینی می‌کند. با نگاه کردن به دقت، بلافاصله می‌توان دریافت که آیا مدل درست آموزش دیده است یا خیر و کارایی آن به طور کلی چگونه است. اما این معیار اطلاعات جزئی در مورد کارایی مدل ارائه نمی‌دهد.

$$Accuracy = (TP+TN) / (TP+FN+FP+TN)$$

Precision

وقتی که مدل نتیجه را مثبت (positive) پیش‌بینی می‌کند، این نتیجه تا چه اندازه درست است؟ زمانی که ارزش false positives بالا باشد، معیار صحت، معیار مناسبی خواهد بود. فرض کنید، مدلی برای تشخیص سرطان داشته باشیم و این مدل Precision پایینی داشته باشد. نتیجه این امر این است که این مدل، بیماری بسیاری از افراد را به اشتباه سرطان تشخیص می‌دهد. نتیجه این امر استرس زیاد، آزمایش‌های فراوان و هزینه‌های گزافی را برای بیمار به دنبال خواهد داشت.

$$Precision = TP / (TP+FP)$$

حساسیت Sensitivity

در واقع، «حساسیت» معیاری است که مشخص می‌کند دسته‌بند، به چه اندازه در تشخیص تمام افراد مبتلا به بیماری موفق بوده است. همانگونه که از رابطه فوق مشخص است، تعداد افراد سالمی که توسط دسته‌بند به اشتباه به عنوان فرد بیمار تشخیص داده شده‌اند، هیچ تاثیری در محاسبه این پارامتر ندارد و در واقع زمانی که پژوهشگر از این پارامتر به عنوان پارامتر ارزیابی برای دسته‌بند خود استفاده می‌کند، هدفش دستیابی به نهایت دقت در تشخیص نمونه‌های کلاس مثبت است.

Specificity

در نقطه مقابل این پارامتر، ممکن است در مواقعی دقت تشخیص کلاس منفی حائز اهمیت باشد. از متداول‌ترین پارامترها که معمولاً در کنار حساسیت بررسی می‌شود، پارامتر خاصیت (Specificity)، است که به آن «نرخ پاسخ‌های منفی درست» (True Negative Rate) نیز می‌گویند. خاصیت به معنی نسبتی از موارد منفی است که آزمایش آن‌ها را به درستی به عنوان نمونه منفی تشخیص داده است. این پارامتر به صورت زیر محاسبه می‌شود. زمانی که ارزش false negatives بالا باشد، معیار Recall، معیار مناسبی خواهد بود. فرض کنیم مدلی برای تشخیص بیماری کشنده ابولا داشته باشیم. اگر این مدل Recall پایینی داشته باشد چه اتفاقی خواهد افتاد؟ این مدل افراد زیادی که آلوده به این بیماری هستند را سالم در نظر می‌گیرد و این فاجعه است. نسبت مقداری موارد صحیح طبقه‌بندی شده توسط الگوریتم از یک کلاس به تعداد موارد حاضر در کلاس مذکور که به صورت زیر محاسبه می‌شود:

$$\text{Recall} = \text{Sensitivity} = (\text{TPR}) = \text{TP} / (\text{TP} + \text{FN})$$

$$\text{Specificity (TNR)} = \text{TN} / (\text{TN} + \text{FP})$$

Mean Intersection over Union = mIoU

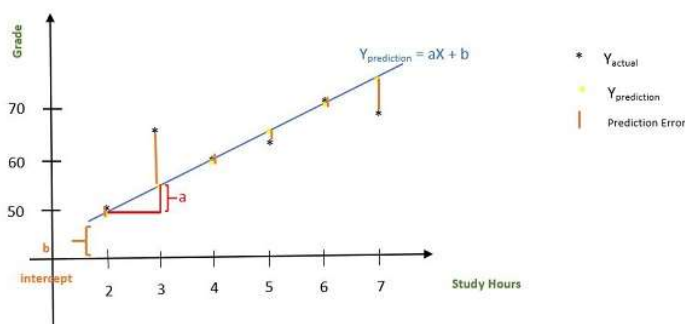
میانگین نقاط مشترک در سطوح متقاطع

$$mIoU = \frac{1}{N} \sum_{i=1}^N \frac{p_{ii}}{\sum_{j=1}^N p_{ij} + \sum_{j=1}^K p_{ji} - p_{ii}}$$

تابع هزینه

تابع هزینه عمومی

هر چقدر هم دقیق باشیم، ترسیم خط پیش بینی بدون خطا گاهی غیرممکن یا دشوار است. میزان خطا در یادگیری ماشین اجتناب ناپذیر است. اما ما باید بهترین، a و b را پیدا کنیم و میزان خطا را به حداقل برسانیم تا خروجی دقیق داشته باشیم. فاصله عمودی بین مقدار واقعی که یک ستاره سیاه است و مقدار پیش بینی شده روی خط آبی که یک ستاره زرد است "خطای پیش بینی" یا "هزینه" نامیده می‌شود. به منظور محاسبه خطای پیش بینی؛ ما باید منهای خطای پیش بینی = واقعی - پیش بینی پیش بینی شده $aX + b$



$$\text{Cost Function} = \frac{1}{2m} * \sum (Y - Y_p)^2 \quad (m \text{ is record number})$$

تابع هزینه (FL) Focal Loss

به عبارت ساده ، Focal Loss (FL) نسخه بهبود یافته Cross-Entropy Loss (CE) است که سعی می کند با تعیین وزنه های بیشتر به مثال های سخت یا طبقه بندی اشتباه (به عنوان مثال پس زمینه با بافت noisy) به مثالهای آسان کاهش دهد.

Cross-Entropy Loss (CE)

عدم نظم یا پیش بینی کاهش تدریجی به بی نظمی.

Cross Entropy Loss:

$$L(\Theta) = - \sum_{i=1}^k y_i \log(\hat{y}_i)$$

$$\text{CELoss} = \{ \sum_{h=0}^{H-1} \sum_{w=0}^{W-1} \} (\text{CELoss} / H \cdot W)$$

فاصله انحراف (Distance Deviation Cross Loss Function: DDCLoss)

$$DDC\text{Loss}_{p_{h,w}} = - \sum_{k=0}^N (2d_k)^\beta (1 - c_k)^\alpha \log(1 - d_k)$$

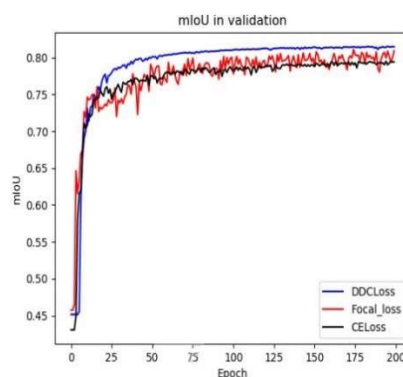
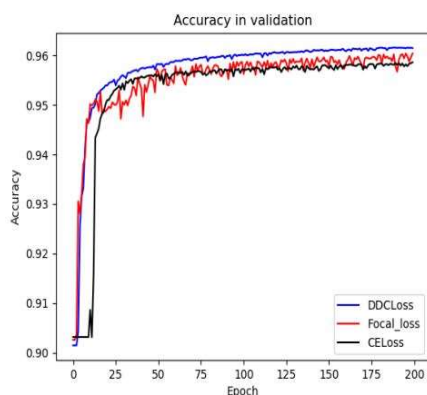
$$DDC\text{Loss} = \frac{\sum_{h=0}^{H-1} \sum_{w=0}^{W-1} DDC\text{Loss}_{p_{h,w}}}{H \cdot W}$$

نتیجه گیری

به منظور ارتقا توانایی مدل در استخراج اطلاعات محلی چند مقیاسی در تصاویر پزشکی ، ما یک شبکه عصبی تقسیم بندی تصویر پزشکی ، به نام LCP-Net پیشنهاد می کنیم. ایده اصلی LCP-Net معرفی یک ماژول جدید درک مقیاس چند منظوره (LCP) در رمزگذارهای ویژگی است. LCP از چندین لایه کانولوشن متسع شده موازی، به استخراج اطلاعات زمینه در مقیاس چند منظوره تصاویر می پردازد. در همین حال ، در LCP ، مکانیسم تعبیه محلی بافت (LCE) برای افزایش درک اطلاعات زمینه در نقشه های ویژگی استفاده می شود. علاوه بر این ، ما DDCLoss را بعنوان تابع هزینه ارائه می دهیم ، که در واقع یک عملکرد از آنتروپی بهبود یافته بر اساس قطعیت و فاصله انحراف است.

نتایج تجربی در مورد معیارهای ارزیابی چندگانه نشان می دهد که ، در مقایسه با پیشرفته ترین روش های تقسیم تصویر پزشکی ، LCP-Net پیشنهادی ما می تواند با افزایش درک متنی از استخراج کننده ویژگی ، به طور موثری دقت تقسیم بندی تصویر پزشکی را بهبود بخشد. آزمایشات مقایسه ای با آنتروپی متقابل و از دست دادن کانونی نشان می دهد که DDCLoss پیشنهادی می تواند به خوبی توانایی مدل را برای تقسیم به اشیا کوچک بهبود بخشد، و بازگرداندن اطلاعات دقیق از تصویر اصلی ، و استفاده از DDCLoss به عنوان یک تابع Loss می تواند روند آموزش را با ثبات تر کند.

<i>Loss Function</i>	<i>Acc(%)</i>	<i>mIoU(%)</i>	<i>Sen(%)</i>	<i>Spe(%)</i>
LCP-Net+CELoss	96.62	81.13	78.49	98.53
LCP-Net+Focal loss	96.83	81.74	78.52	98.67
LCP-Net+DDCLoss	97.04	82.14	78.72	99.13



Reference

[1] mathworks.com

[2] bigdata-ir.com

[3] howsam.org/convolutional-neural-network

[4] fanology.ir/convolutional-neural-network

[5] isic-archive.com

[6] ostovae.ir

[7] roboticshell.com