دانشگاه صنعتی امیرکبیر

(پلی‌تکنیک تهران)

دانشکده مهندسی کامپیوتر

پروژه کارشناسی

تقسیم‌بندی معنایی برای ماشین‌های خودران با استفاده از یادگیری عمیق

نگارش

کیوان ایپچی حق

استاد راهنما

دکتر احسان ناظرفرد

فروردین ۱۴۰۳


بسم الله الرحمن الرحيم

دانشگاه صنعتی امیرکبیر
(پلی‌تکنیک تهران)

به نام خدا

تعهدنامه اصالت اثر

تاریخ: فروردین ۱۴۰۳

اینجانب **کیوان ایپچی حق** متعهد می‌شوم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظارت و راهنمایی اساتید دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مآخذ ذکر گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است.

در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان‌نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، نسخه‌برداری، ترجمه و اقتباس از این پایان نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مآخذ بلامانع است.

کیوان ایپچی حق

امضا

تقدیم به

آنان که الفبای انسانیت و چگونه زیستن را به من آموختند...

سپاس گزاری

بدین وسیله از زحمات و تلاش بی‌دریغ استاد محترم جناب دکتر احسان ناظرفرد و خانواده عزیزم صمیمانه سپاسگزاری می نمایم و همچنین از سایر همکاران و دوستانی که هر کدام به نحوی در تهیه این مجموعه با این جانب همکاری داشته اند تشکر نموده و موفقیت همه آنها را از خداوند متعال خواهانم.

کیوان اویچی حق
فروردین ۱۴۰۳

چکیده

خودروهای خودران [1] به منظور اتخاذ تصمیمات آگاهانه و مسیریابی ایمن در محیط‌های مختلف، نیازمند درک دقیقی از اشیاء اطراف خود هستند. تقسیم‌بندی معنایی [2] از ابتدایی‌ترین مراحل در فرایند تجزیه و تحلیل تصاویر و استخراج اطلاعات مفید آن به منظور تصمیم‌گیری در اینگونه سیستم‌ها است که نقش حیاتی در تشخیص اشیاء محیط دارد و این امکان را بوجود می‌آورد تا به طور دقیق اشیاء مختلف از جمله جاده‌ها، عابران پیاده، خودروهای دیگر و موانع شناسایی شوند. روش‌های یادگیری عمیق [3] بهبود قابل توجهی در تقسیم‌بندی معنایی تصاویر به وجود آورده‌اند، به گونه‌ای که از عملکرد برتری نسبت به روش‌های سنتی برخوردار هستند. این پروژه به بررسی پیشرفت‌های اخیر در زمینه تقسیم‌بندی معنایی تصاویر برای خودروهای خودران با استفاده از روش‌های یادگیری عمیق می‌پردازد. ما معماری‌های مختلف یادگیری عمیق را در مسئله تقسیم‌بندی معنایی سریع [4] مورد مطالعه قرار داده، معماری‌های مختلف را مقایسه نموده و نقاط قوت و ضعف آنها را برای مسئله خاص خودروهای خودران بررسی می‌کنیم. علاوه بر این، مجموعه‌داده‌های مورد استفاده برای آموزش و ارزیابی مدل‌های تقسیم‌بندی معنایی در این حوزه مورد بررسی قرار گرفته و از آنها برای ارزیابی مدل‌های یادگیری عمیق مختلف استفاده می‌شود. در پایان، جمع‌بندی بر روی مدل‌های مورد بررسی قرار گرفته خواهیم داشت و پیشنهاداتی برای پژوهش‌های آینده در جهت بهبود پایداری، کارایی و قابلیت عمومی سیستم‌های تقسیم‌بندی معنایی در لحظه مبتنی بر یادگیری عمیق برای خودروهای خودران ارائه می‌شود.

واژه‌های کلیدی:

هوش مصنوعی، خودروهای خودران، یادگیری عمیق، تقسیم بندی معنایی، تقسیم‌بندی معنایی سریع تصاویر

[1] Self-driving cars

[2] Semantic segmentation

[3] Deep learning

[4] Fast semantic segmentation

فهرست مطالب

عنوان صفحه

فهرست اشکال

شکل صفحه

فهرست جداول

جدول صفحه

فهرست نمادها

نماد	مفهوم
$\mathbb{R}^n$	فضای اقلیدسی با بعد n
$\mathbb{S}^n$	کره یکه n بعدی
M^m	خمینه m-بعدی M
$\mathfrak{X}(M)$	جبر میدان‌های برداری هموار روی M
$\mathfrak{X}^1(M)$	مجموعه میدان‌های برداری هموار یکه روی (M,g)
$\Omega^p(M)$	مجموعه p-فرمی‌های روی خمینه M
Q	اپراتور ریچی
$\mathcal{R}$	تانسور انحنای ریمان
ric	تانسور ریچی
L	مشتق لی
Φ	۲-فرم اساسی خمینه تماسی
∇	التصاق لوی-چویتای
Δ	لاپلاسین ناهموار
∇^*	عملگر خودالحاق صوری القا شده از التصاق لوی-چویتای
g_s	متر ساساکی
∇	التصاق لوی-چویتای وابسته به متر ساساکی
Δ	عملگر لاپلاس-بلترامی روی p-فرم‌ها

فصل اول

مقدمه

۱-۱ شرح مسأله

در دهه اخیر، پیشرفت‌های چشمگیری در زمینه هوش مصنوعی و یادگیری عمیق، به ویژه در حوزه پردازش تصویر و استفاده از آنها برای بهبود عملکرد تصمیم‌گیری در خودروهای خودران، انقلابی در روند توسعه و بهینه‌سازی فناوری در این زمینه ایجاد کرده است. شبکه‌های عصبی عمیق [۱] به دلیل قابلیت‌هایی که از طریق شبکه‌های عصبی کانولوشنی [۲] [۱] فراهم می‌آید، امکاناتی را برای خودروها فراهم می‌کند که پیش از این غیرقابل تصور بوده است.

با این وجود، یکی از چالش‌های بزرگ در مسیر توسعه خودروهای خودران، توانایی فهم و تفسیر دقیق محیط اطراف و اشیاء موجود در تصویر است. برای حل این چالش معمولاً از روش‌های متنوعی استفاده می‌شود که یکی از روش‌های مهم در این زمینه، تقسیم بندی معنایی نامیده می‌شود. در این روش، تمامی سلول‌های [۳] تصویری موجود به دسته‌هایی از پیش تعیین شده تخصیص داده می‌شوند. خودرو باید توانایی آن را داشته باشد تا اطلاعات دریافتی از محیط را با سرعت در لحظه [۴] به دسته‌های مختلف مانند خیابان، پیاده‌رو، خودروها، چراغ راهنما و غیره تقسیم بندی کرده و به هر دسته یک رنگ مخصوص که اسطلاحا به آن رنگبندی تقسیم‌بندی [۵] گفته می‌شود اختصاص دهد.

بدون شک، تقسیم‌بندی معنایی محیط برای خودروهای خودران امری بسیار اساسی و حیاتی است. اطلاعات دقیق و صحیح در مورد محیط اطراف، به سیستم‌های خودران امکان می‌دهد تا تصمیمات صحیح و ایمن را در مسیر حرکت خود اتخاذ کنند. این اطلاعات، پایه‌ای برای عملکرد امن و کارآمد این خودروهای خودران است. در عین اهمیت داشتن دقت بالا در انجام این امر، پردازش در لحظه نیز حائز اهمیت است. زیرا تنها داشتن دقت بالا بدون توانایی پردازش سریع و به موقع، در مسائلی که نیاز به پردازش آنی دارند ناکارآمد خواهد بود. به عبارتی دیگر، دقت بالا و سرعت پردازش به‌طور همزمان، می‌توانند به عنوان دو عامل اساسی و مکمل، عملکرد بهینه سیستم خودران را فراهم کنند.

۱-۲ اهداف پروژه

ادر بخش ابتدایی از پروژه، به توضیح مقدمه‌ای بر چگونگی انجام تقسیم‌بندی معنایی تصاویر پرداخته و سپس به بررسی روش‌های تقسیم‌بندی معنایی بااستفاده از یادگیری عمیق که برای استفاده در حوزه تصویربرداری پزشکی طراحی شده‌اند [۶] می‌پردازیم. تمرکز ما در این بخش بررسی مدل‌هایی است که از دقت بالایی برخوردار هستند و سرعت عمل به عنوان یک مشخصه ثانویه مطرح نمی‌شود، زیرا هدف این مدل‌ها در صنعت پزشکی، تشخیص درست و دقیق اجزای موجود در تصویر است که سرعت حائز

[1] Deep neural networks
[2] convolutional neural networks
[3] Pixel
[4] Real-time
[5] Segmentation color
[6] Medical Imaging

اهمیت چندانی نیست. در ادامه، به بررسی روش‌های یادگیری عمیق برای تقسیم‌بندی معنایی در حوزه خودروهای خودران با تمرکز بر پردازش در لحظه پرداخته و هدف ما دقت بالا و سرعت عمل بهینه متناسب با این مسئله است. با توجه به نیازهای خاص این حوزه، ما به دنبال راهکارها و معماری‌هایی هستیم که بهبود دقت و سرعت عمل سیستم‌های خودران را به هدف داشته و در نتیجه، ایمنی و کارایی این سیستم‌ها را بهبود بخشند. در پایان، به جمع‌بندی و نتیجه‌گیری مطالب بدست آمده در این پروژه پرداخته و مروری بر سیر انجام پروژه و معماری‌های مطرح شده خواهیم داشت و سپس پیشنهاداتی برای کارهای آینده که می‌تواند به بهبود وضعیت فعلی کمک کند، مطرح می‌کنیم.

۱-۳ ساختار گزارش

در فصل ابتدایی این گزارش، مقدمه‌ای بر روی مسأله مطرح شده در این پروژه و شرح کلی از اهداف و محتوای گزارش ارائه شد. در فصل دوم، به طور مفصل به مفاهیم مرتبط و چگونگی پیاده‌سازی این مسأله پرداخته و سپس به بررسی معماری رمزگذار-رمزگشا و مدل‌هایی که از این معماری استفاده می‌کنند پرداخته خواهد شد. در فصل سوم، به مدل‌های طراحی شده برای مساله خاص تقسیم‌بندی معنایی در خودروهای خودران اشاره شده و سپس مدل‌های پیشنهادی این پژوهش انتخاب و به طور مفصل توضیح داده خواهد شد. در فصل چهارم، آزمایش‌ها، نتایج و ارزیابی‌های انجام شده بر روی کیفیت و کارایی مدل‌ها مورد بحث و بررسی قرار خواهند گرفت. در فصل پنجم، به عنوان فصل پایانی، جمع‌بندی نکات گزارش و پیشنهادهایی برای کارهای آینده به منظور بهبود عملکرد و کارایی در این حوزه خواهد شد.

فصل دوم

مرور کارهای پیشین

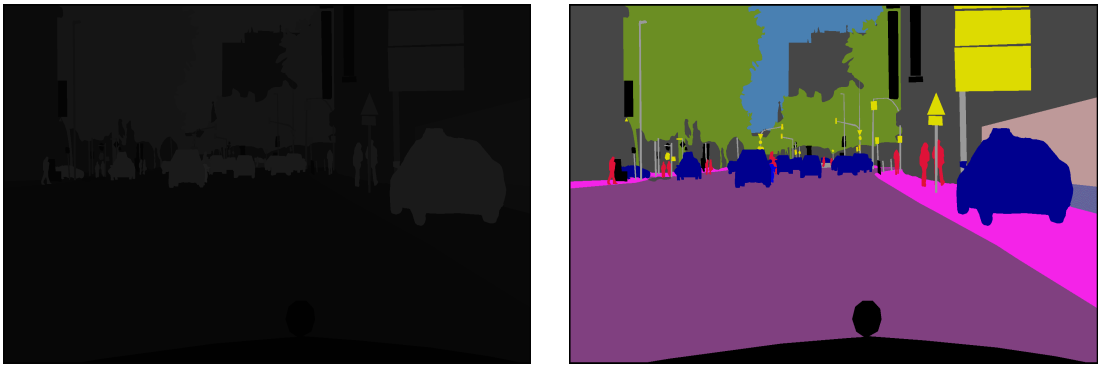
۲-۱ مقدمه‌ای تقسیم‌بندی معنایی

در مسأله تقسیم‌بندی معنایی، مدل یک نقشه تقسیم‌بندی شده [۱] از شناسه‌ها [۲] را تولید می‌کند که هر سلول تصویر به یک شناسه خاص مرتبط با دسته‌بندی هر شیء اختصاص می‌یابد. این نقشه تقسیم‌بندی شده در واقع یک تصویر خاکستری دو بعدی [۳] است، زیرا صرفا شامل شناسه ها که خود اعداد کوچک بوده است که باعث می‌شوند تصویر تیره و با تنها یک کانال رنگی تولید شود که در آن هر مقدار سلول متناظر با شناسه دسته شیء مورد نظر است که نمایانگر دسته آن شیء است. به عنوان مثال، در یک نقشه تقسیم‌بندی شده، مقادیر پیکسل ۱ ممکن است نمایانگر زمینه باشد، مقادیر ۲ ممکن است نمایانگر یک عابر پیاده و مقادیر ۳ ممکن است نمایانگر یک خودرو باشد و غیره.

تمایز بین اشیاء موجود در یک تصویر خاکستری برای چشم انسان کار دشواری است. به همین دلیل، برای تبدیل این نقشه تقسیم‌بندی خاکستری به یک تصویر تقسیم‌بندی شده رنگارنگ [۴] که به صورت بصری شیء‌های تقسیم‌بندی شده را نشان می‌دهد، از یک تبدیل بین رنگ‌ها به شناسه‌ها و برعکس آن استفاده می‌شود. این تبدیل شامل اختصاص رنگ‌های متمایز به هر دسته (شناسه اشیاء) است. رویکرد رایج‌تر، استفاده از یک پالت رنگ [۵] پیش‌تعیین شده است که هر کلاس با یک کد رنگی [۶] منحصر به فرد مرتبط است.

از این تبدیل برای تغییر تصاویر رنگارنگ به خاکستری قبل از دادن آنها به مدل و برعکس آن بر روی خروجی مدل استفاده می‌شود. به طوری که تصاویر رنگارنگ به تصاویر خاکستری، که شامل شناسه‌های اشیاء هستند تبدیل شده و وارد مدل می‌شوند. سپس، خروجی مدل که نیز تصاویر خاکستری هستند به تصاویر رنگارنگ تقسیم‌بندی شده تبدیل می‌شوند و یک تصویر بصری رنگی بوجود آورده می‌شود که در آن اشیاء مختلف با رنگ‌های متمایز مشخص شده‌اند و نمایش واضح و روشنی از نتایج تقسیم‌بندی معنایی ارائه می‌دهد. این تصویر رنگی سپس می‌تواند برای تحلیل و نمایش مورد استفاده قرار گیرد.

[1] Segmentation map
[2] Segmentation id
[3] Grayscale image
[4] Segmentation image
[5] Color palette
[6] RGBA



(آ) تصویر رنگارنگ (ب) تصویر خاکستری شناسه‌ها

شکل ۲-۱: نمونه تبدیل نقشه تقسیم‌بندی شده به تصویر رنگارنگ متناظر

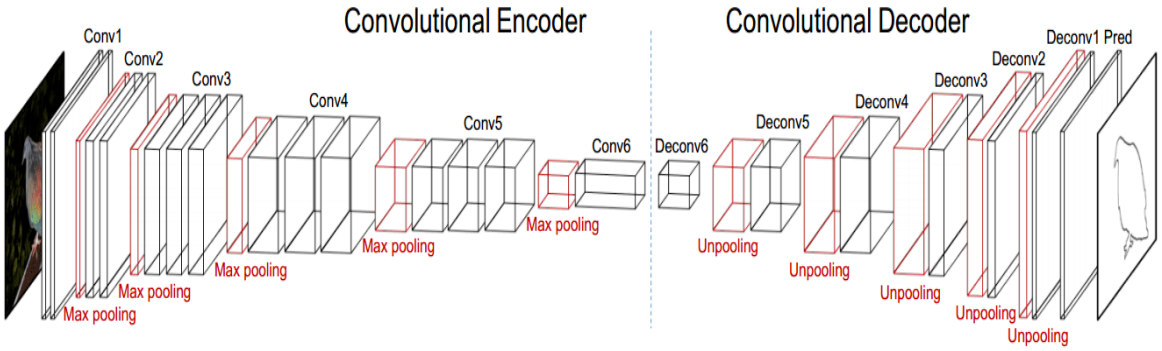
۲-۲ معماری رمزگذار-رمزگشا

معماری رمزگذار-رمزگشا [۷] یک نوع معماری شبکه عصبی است که برای یادگیری توالی به توالی [۸] [۲] مورد استفاده قرار می‌گیرد. این معماری شامل دو بخش اصلی، یعنی رمزگذار [۹] و رمزگشا [۱۰] است که در آن رمزگذار تصویر ورودی را دریافت و پردازش می‌کند تا مجموعه‌ای از بردارهای ویژگی [۱۱] برای تصویر تولید کند. سپس، این بردارهای ویژگی توسط رمزگشا برای بزرگ‌نمایی تصویر خروجی به اندازه تصویر ورودی استفاده می‌شوند. ایده اصلی پشت این معماری آن است که بتواند یک فرم از داده (در اینجا تصویر) را دریافت کرده و به فرم دیگری (مانند تصویر تقسیم‌بندی معنایی شده معادل) تبدیل کند. با انجام این کار، خودرو قادر خواهد بود که چگونگی ارتباطات بین تصاویر ورودی و خروجی را درک کند.

این معماری می‌تواند در بسیاری از حوزه‌ها مورد استفاده قرار بگیرد، از جمله پردازش تصویر، ترجمه ماشینی [۱۲] ، تولید متن توسط تصویر [۱۳] ، و غیره. در هر حالت، رمزگذار مسئول استخراج ویژگی‌های مهم از داده ورودی و تولید ویژگی‌های نهان است که اطلاعات اصلی داده را در خود جاسازی می‌کند. سپس، رمزگشا این ویژگی‌های نهان را به فرم دیگری از داده ترجمه می‌کند که معمولاً خروجی مورد نظر است. این معماری به عنوان یکی از روش‌های موثر برای یادگیری مدل‌های پیچیده از داده‌های توالی به توالی شناخته می‌شود و در مسائلی که توالی و ارتباطات بین داده‌ها مهم هستند، بسیار مفید است. در

[7] Encoder-decoder
[8] Sequence to sequence learning
[9] Encoder
[10] Decoder
[11] Feature vectors
[12] Machine translation
[13] Image to text

تصویر زیر، این معماری به صورت یک نمودار نشان داده شده است:



شکل ۲-۲: معماری رمزگذار-رمزگشا

۲-۲-۱ شبکه رمزگذار

رمزگذار، اولین بخش از معماری رمزگذار-رمزگشا است و در پردازش و استخراج اطلاعات از داده ورودی نقش اساسی دارد. این بخش وظیفه استخراج ویژگی‌های معنادار از داده را بر عهده دارد که سپس توسط رمزگشا برای پردازش و بازسازی اطلاعات به فرم دیگر استفاده می‌شود. روشی که فرآیند رمزگذاری کار می‌کند، بسته به نوع کاربرد متفاوت است. در وظایف پردازش تصویر، عموماً از لایه‌های کانولوشنی [۱۴] به همراه لایه‌های ادغام [۱۵] ، فعال‌ساز [۱۶] و نرمال‌ساز [۱۷] استفاده می‌شود تا تصویر اصلی به مرور کوچک‌تر شده، اطلاعات اضافی و فضاهای خالی آن حذف شده و در نهایت به تعدادی بردار ویژگی شکسته شود.

لایه‌های کانولوشنی مسئول استخراج ویژگی‌های تصویر ورودی هستند. این لایه‌ها از ویژگی‌های سطح پایین مانند لبه‌ها و رنگ‌ها تا ویژگی‌های سطح بالاتری مانند شکل‌ها و ساختارهای اشیاء را یاد می‌گیرند. هر لایه کانولوشنی مجموعه‌ای از فیلترها را به تصویر ورودی اعمال می‌کند و آن را به بردارهای ویژگی تبدیل می‌کند که جنبه‌های مختلفی از محتوای تصویر را شامل می‌شود.

لایه‌های ادغام نقشه‌های ویژگی را با حفظ اطلاعات مهم‌تر آن کاهش یا به اصطلاح خلاصه می‌کنند. توابع فعال‌سازی غیرخطی [۱۸] به شبکه این امکان را می‌دهند که روابط پیچیده در داده‌ها آموخته شود. لایه‌های نرمال‌ساز مانند نرمال‌سازی دسته‌ای [۱۹] نیز حساسیت شبکه را به وزن‌های اولیه و نرخ یادگیری

[14] Convolutional layers
[15] Pooling layer
[16] Activation function
[17] Normalization
[18] Non-linear activation functions
[19] Batch normalization

[۲۰] کاهش می‌دهند و کمک می‌کنند مدل بتواند نرخ‌های یادگیری بالاتری را نیز تحمل کرده و مشکلاتی نظیر انفجار [۲۱] و یا ناپدید شدن گرادیان‌ها [۲۲] رخ ندهد.

۲-۲-۲ شبکه رمزگشا

رمزگشا، بخش دوم و مهم از معماری رمزگذار-رمزگشا است که مسئول بازسازی بردارهای ویژگی حاصل از رمزگذار و بازسازی آن به شکل اصلی یا شبیه به آن است. این بخش از معماری معمولاً با استفاده از لایه‌های کانولوشنی معکوس [۲۳] و یا لایه‌های ادغام معکوس [۲۴] طراحی می‌شود. برای انجام این کار، باید ارتباطی بین آنچه که رمزگذار شده و آنچه که باید بازسازی شود وجود داشته باشد که عموما در لایه و یا لایه‌هایی بین رمزگذار و رمزگشا به عنوان فضای پنهان [۲۵] ذخیره می‌شود تا رمزگشا بتواند خروجی معناداری با استفاده از این واحد تولید کند.

از مشکل رایج در معماری رمزگذار-رمزگشا به اندازه بزرگ نبودن فضای پنهان یا بیش از اندازه بزرگ بودن آن است که باعث تولید خروجی ضعیف و یا با جزئیات نامطلوب می‌شود. به عبارت دیگر، اگر فضای پنهان به اندازه کافی بزرگ نباشد، ارتباط بین آنچه رمزگذاری شده و آنچه باید بازسازی شود به طور کامل داخل این ذخیره نشده و در نتیجه نمی‌توان بازسازی معناداری داشته باشیم. در عین حال، اگر فضای پنهان بیش از اندازه بزرگ باشد، الگوهای نامطلوبی توسط مدل کشف شده که متناسب با مساله لزوماً مطلوب ما نیستند.

۲-۳ معماری‌های تقسیم‌بندی معنایی در پزشکی

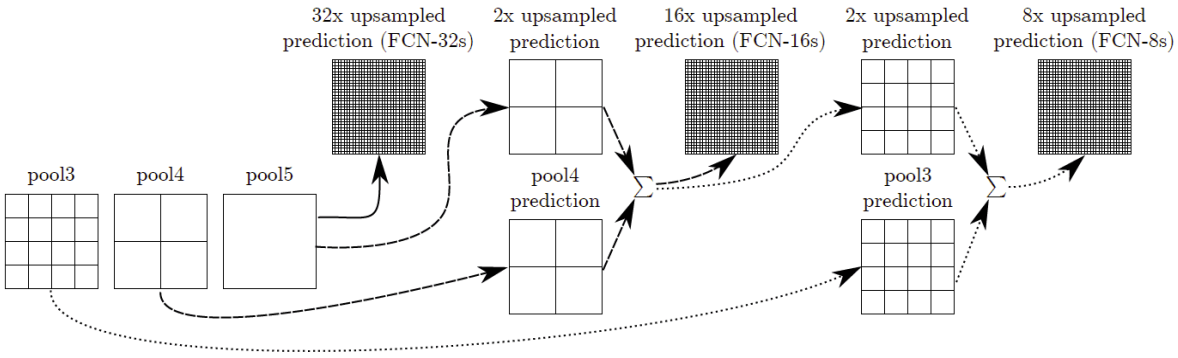
در این بخش به مطالعه چند معماری مورد استفاده در مسایل تقسیم بندی معنایی می‌پردازیم.

[20] Learning rate
[21] Exploding gradient descent
[22] Vanishing gradient descent
[23] Transposed Convolutional layers
[24] Unpooling layers
[25] Latent space

۲-۳-۱ معماری FCN

معماری شبکه کاملا کانولوشنی [۳] [۲۶] یک نوآوری بسیار مهم در زمینه تقسیم‌بندی معنایی است که از مدل‌های سنتی شبکه‌های عصبی کانولوشنی با لایه‌های انتهایی کاملا متصل تفاوت دارد. در این معماری، هر پیکسل تصویر به دسته‌های مختلف تقسیم می‌شود، که این امر با استفاده از لایه‌های کانولوشنی و بدون نیاز به لایه‌های کاملاً [۲۷] متصل انجام می‌شود. در مدل‌های سنتی‌تر مانند VGG [۴] ، از لایه‌های کاملاً متصل برای تولید خروجی دسته‌بندی شده استفاده می‌شوند، در حالی که در معماری کاملا کانولوشنی، این لایه‌های کاملاً متصل با لایه‌های کانولوشنی جایگزین می‌شوند. این تغییر باعث می‌شود که شبکه بتواند تصاویر ورودی با ابعاد دلخواه را بپذیرد و خروجی با همان ابعاد تولید کند.

یکی از مزایای این معماری این است که امکان اجرای تقسیم‌بندی معنایی با دقت بالا بدون نیاز به لایه‌های کاملاً متصل فراهم می‌کند. همچنین، اتصالات پرش [۵] [۲۸] که در این معماری معرفی شده‌اند، امکان ترکیب اطلاعات معنایی از لایه‌های عمیق با اطلاعات ظاهری از لایه‌های کم عمق را فراهم می‌کنند. این امر منجر به تولید تقسیم‌بندی‌های با جزئیات بیشتر می‌شود که بهبود قابل توجهی در کیفیت و دقت تصویرهای تقسیم‌بندی شده دارد.



شکل ۲-۳: معماری اتصالات پرش در مدل کاملا کانولوشنی

اتصالات پرش یکی از ویژگی‌های کلیدی در شبکه‌های عصبی کانولوشنی [۲۹] هستند که در بسیاری از مسایل تقسیم‌بندی معنایی مورد استفاده قرار می‌گیرند. این اتصالات به شبکه این امکان را می‌دهند که اطلاعات از برخی از لایه‌ها عبور کرده و مستقیما به لایه‌های بعدی منتقل شوند، در نتیجه جریان مستقیم‌تری از داده دست نخورده به لایه‌های بعدی داشته باشیم.

[26] Fully convolutional network (FCN)
[27] Fully connected layer (FC)
[28] Skip Connection
[29] Convolutional Neural Network (CNN)

استفاده همزمان از اتصالات پرش بلند و کوتاه در معماری شبکه‌های عصبی کانولوشنی می‌تواند به بهبود قابل توجهی در دقت تقسیم‌بندی منجر شود. اندازه گام این اتصالات (با اندازه‌های ۳۲، ۱۶ و ۸ سلول) مستقیماً بر روی دقت بالانمایی تأثیر می‌گذارد. مدل‌های با گام‌های کوچکتر (در اینجا FCN8)، قادرند جزئیات فضایی بیشتری را حفظ کنند و نقشه‌های تقسیم‌بندی دقیق‌تری تولید کنند. اما به همراه این مزیت، گام‌های کوچکتر نیز هزینه محاسباتی [۳۰] و زمان استنتاج [۳۱] را افزایش می‌دهند.

در ارزیابی مدل‌ها با استفاده از شاخص دقت معمولاً مدل‌های با اندازه گام کوچک‌تر عملکرد بهتری را نسبت به مدل‌های مشابه از خود نشان دهند. به عنوان مثال، مدل FCN8 با مقدار ۷.۶۲ در شاخص میانگین اشتراک بر اجتماع [۳۲] عملکرد بهتری را نسبت به مدل‌های مشابه از خود نشان داده می‌دهد.

۲-۳-۲ معماری U-NET

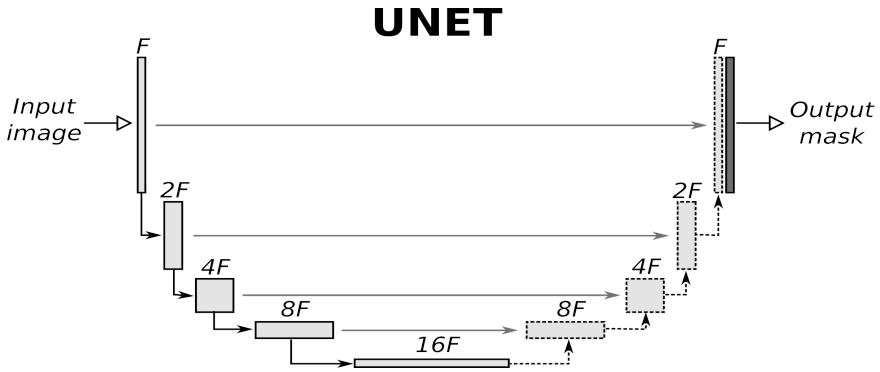
معماری U -شکل [۶] ایده اصلی طرح خود را از شبکه‌های عصبی کانولوشنی می‌گیرد و از آن برای پیش‌بینی پیکسل به پیکسل [۳۳] در تقسیم‌بندی معنایی استفاده می‌کند. این معماری محدودیت‌های معماری‌های سنتی را برای وظایف تقسیم‌بندی معنایی از بین می‌برد. بر خلاف شبکه‌های عصبی کانولوشنی استاندارد که از لایه‌های کاملاً متصل برای تولید خروجی تصنیفی استفاده می‌کنند، معماری U -شکل، مشابه معماری FCN ، یک شبکه کاملاً کانولوشنی است که می‌تواند تصاویر ورودی با ابعاد دلخواه را بپذیرد و نقشه‌های تقسیم‌بندی با همان ابعاد تصاویر ورودی را تولید کند.

این مدل نیز از معماری رمزگذار-رمزگشا پیروی کرده که شکلی مانند حرف U انگلیسی را تشکیل می‌دهند. در بخش رمزگذار، از کانولوشن‌های تکراری و لایه‌های ادغامی برای یادگیری ویژگی‌های سلسله‌مراتبی استفاده می‌شود. در بخش رمزگشا، ویژگی‌ها بالانمایی می‌شوند و با ویژگی‌های برخی از لایه‌های رمزگذار از طریق اتصالات پرش ترکیب می‌شوند. عموما مدل‌های U -شکل از معماری رمزگذار-رمزگشا متقارن [۳۴] [۷] استفاده می‌کنند که به معنای مشابه بودن این دو بخش دو تعداد لایه‌ها و مشخصات هر لایه بوده، با این تفاوت که عکس یکدیگر عمل می‌کنند.

نوآوری اصلی در معماری U -شکل، استفاده کارآمد از اتصالات پرش و توانایی تولید خروجی‌های با

[30] Computational cost
[31] Inference time
[32] Mean IoU
[33] Pixel-to-pixel
[34] Symmetric encoder-decoder architecture

وضوح بالا، حتی با مجموعه داده آموزشی نسبتاً کوچک است. اتصالات پرش به شبکه امکان می‌دهند جزئیات ویژگی‌های رمزگذار و رمزگشا را ترکیب کنند و نقشه‌های تقسیم‌بندی دقیق‌تری تولید کنند. این معماری همچنین بازسازی بهتری روی لبه‌های اشیا انجام می‌دهد.



شکل ۲-۴: معماری مدل U-شکل

۲-۴ خلاصه

به طور کلی، معماری‌های رمزگذار-رمزگشا به طور قابل توجهی رویکرد حل مسایل تقسیم‌بندی معنایی را تغییر داده‌اند. این معماری‌ها از توانایی‌های شبکه‌های عصبی کانولوشنی برای پیش‌بینی دقیق پیکسل به پیکسل بهره می‌برند و به وجود آوردن نقشه‌های تقسیم‌بندی دقیق و جزئی‌تر کمک می‌کنند. با استفاده از تکنیک‌هایی مانند اتصالات پرش و بالانمایی، ساختار رمزگذار-رمزگشا قادر است جزئیات فضایی و اطلاعات معنایی را با دقت بالاتر در نظر بگیرد و نقشه‌های تقسیم‌بندی دقیق‌تری را تولید کند. با این حال، معماری‌های FCN و U -شکل هر کدام ویژگی‌ها و مزایای خود را دارند. معماری شکل-□ به دلیل ساختار رمزگذار-رمزگشا تقارنی و استفاده گسترده از اتصالات پرش، نسبت به FCN برتری دارد. طراحی منحصر به فرد این مدل به آن امکان می‌دهد که حتی با مجموعه داده‌های آموزشی محدود، خروجی‌های تقسیم‌بندی با وضوح بالا را تولید کند، به ویژه در وظایفی که دقت به جزئیاتی نظیر لبه‌ها حیاتی است، مانند تقسیم‌بندی تصاویر پزشکی. از سوی دیگر، FCN رویکردی انعطاف‌پذیرتر را ارائه می‌دهد و ممکن است در مواردی که کارآیی محاسباتی یا مجموعه داده‌های آموزشی بزرگ اولویت دارند، ترجیح داده شود. به طور خلاصه، انتخاب بین این دو معماری بسته به نیازها و شرایط خاص هر پروژه است. با این حال هر دو از سرعت پایینی در زمان استنتاج برخوردار که آن‌ها را برای پردازش لحظه‌ای مناسب نمی‌سازد.

فصل سوم

روش‌های پیشنهادی

۳-۱ معماری رمزگذار-رمزگشا

در فصل پیشین، مرور کارهای مرتبط، به توضیح مفاهیم پرداخته و به چندین مدل کارآمد در مسایل تقسیم‌بندی معنایی اشاره کردیم. بزرگ‌ترین اشکال استفاده از مدل های ذکر شده برای پردازش آنی، سرعت پایین آنها بوده که در استفاده برای خودروهای خودران چالش‌برانگیز می‌شود. در این فصل به طور مفصل به بررسی چندین مدل پیشنهادی برای تقسیم‌بندی معنایی که بخصوص برای پردازش آنی در خودروهای خودران طراحی شده اند می‌پردازیم.

۳-۱-۱ مدل SQNet

از راهکار های ساده برای بهبود عملکرد اکثر مدل های یادگیری عمیق که برای حل مسایل، افزایش اندازه شبکه است. این راهکار در حوزه تقسیم‌بندی معنایی نیز منجر به بوجود آمدن معماری های نوین مانند شبکه های مولد [۱] و مدل‌های انتشاری [۲] شده است که به دلیل دقت بالا عموما در حوزه پزشکی مورد استفاده قرار می‌گیرند، اما به دلیل سرعت پایین آن‌ها در حوزه خودروهای خودران عملکرد خوبی از خود نشان نمی‌دهند. استفاده از این شبکه‌های دقیق اما بزرگ برای خودروهای خودران به‌طور کلی غیرقابل اجرا یا حداقل با دشواری و هزینه بسیار زیادی همراه است. پس تمرکز به سوی بهینه‌تر کردن مدل ها و استفاده از روش های نوین برای رسیدن به دقت مشابه و سرعت بیشتر تغییر کرده است، زیرا در خودروهای خودران قدرت پردازش و زمان کافی حائز اهمیت است. پس عمده مدل‌های معرفی شده به نوعی به معاوضه بین دقت و سرعت پرداخته و در تلاش هستند که با کمترین از دست رفت دقت بتوان سرعت پردازش را بالا برد.

تحقیقات گسترده ای بر روی کاهش توان پردازش مورد نیاز برای تقسیم‌بندی معنایی در خودروهای خودران صورت گرفته است. برای مثال معماری SqueezeNet [۸] نشان داد که با استفاده از یک معماری موثرتر که در آن از کانولوشن‌های تک واحدی برای فشرده کردن اطلاعات بکار گرفته شده، می‌توان همان دقت در پردازش تصاویر را با استفاده از ۵۰ برابر تعداد وزن‌های کمتر ایجاد نمود. همچنین با تغییر مولفه‌های جزئی‌تری مانند توابع فعال‌ساز و یا حذف لایه‌هایی نظیر نرمال‌ساز می‌توانند در افزایش سرعت موثر باشند. مدل SQNet [۹] که نیز از معماری رمزگذار-رمزگشا [۳] استفاده می کند، توانسته است با

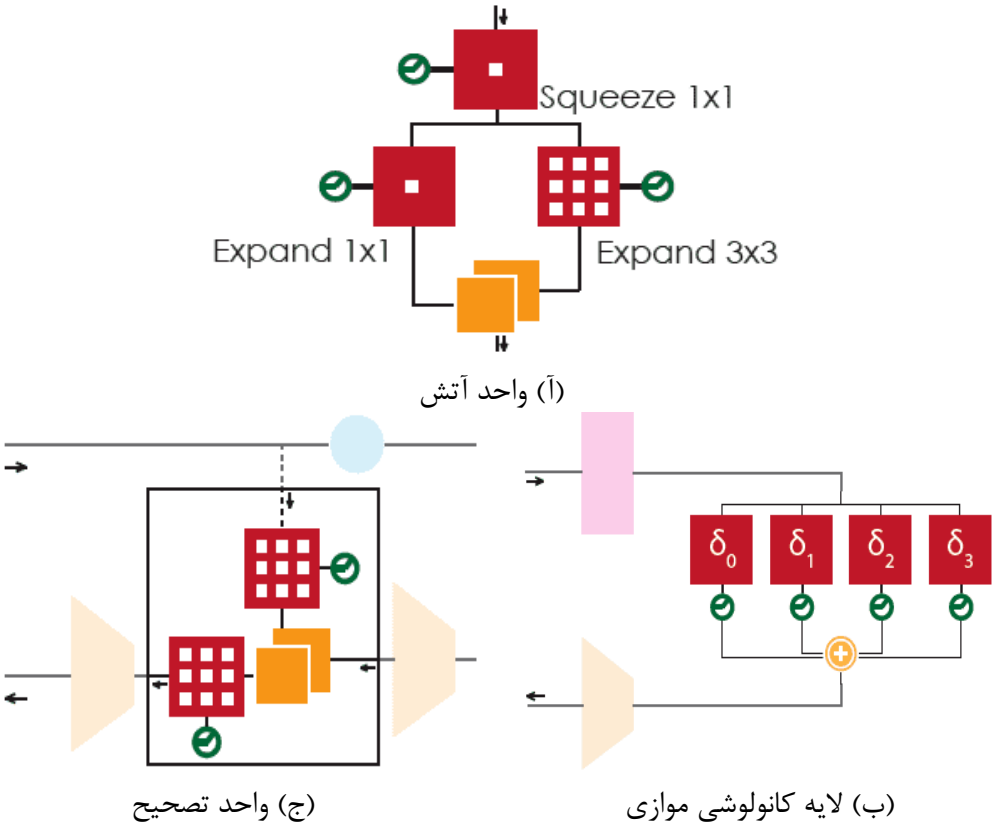
[1] Generative Adversarial Networks (GAN)

[2] Diffusion Model

[3] Encoder-Decoder Architecture

استفاده از این‌گونه تغییرات به پردازش تقریبا آنی دست یابند.

معماری رمزگذار، مشابه آنچه در SqueezeNet آمده طراحی شده که ویژگی قابل توجه آن تعداد وزن‌های کمتر آن است که منجر به سرعت گرفتن پردازش آن می‌شود. بخش محاسباتی اصلی در این معماری آتش [۴] نامیده می‌شود که شامل سه عمل کانولوشن به همراه دو تابع‌فعال‌ساز می‌باشد. توابع‌فعال‌ساز یکسوساز [۵] با توابع‌فعال‌ساز واحد نمایی خطی [۶] جایگزین شده‌اند که بار محاسباتی کمتری داشته، و در عین حال اطلاعات منفی همچنان انتقال پیدا می‌کنند. در رمزگذار از هشت واحد آتش و سه لایه ادغام برای کاهش ابعاد تصویر استفاده شده است.



(آ) واحد آتش

(ب) لایه کانولوشی موازی

(ج) واحد تصحیح

شکل ۳-۱: اجزاء معماری SQNet

رمزگشا بر اساس یک لایه‌های convolutions dilated parallel مبتنی است که نقشه ویژگی‌ها را در خروجی رمزگذار در اندازه‌های میدان تاثیر مختلف ترکیب می‌کند. این واحد این کار را با استفاده از چهار کانولوشن با اندازه کرنل ۳ انجام می‌دهد که معادل نمونه‌برداری لایه ورودی با نرخ‌های مختلف است. در نهایت خروجی چهار کانولوشن را با جمع زدن با یکدیگر با هم ادغام می‌کنیم

[4] Fire module

[5] Rectified linear unit (ReLU)

[6] Exponential linear unit (ELU)

که باعث می‌شود اندازه میدان دید در ورودی رمزگشا افزایش یابد. این کار باعث می‌شود نسبت به شبکه های تماما متصل، تعداد وزن‌های قابل توجه کمتری را داشته باشیم در حالی که عملکرد مدل حفظ می‌شود.

لایه‌های ادغامی داخل رمزگذار برای اطمینان از پایا بودن [۷] در انتقالات استفاده می‌شوند. با این حال، این لایه‌ها به مرور زمان باعث کاهش وضوح تصویر می‌شوند، چراکه هر بار برخی از اطلاعات تصویر خلاصه یا حذف می‌شود. کانولوشن‌های معکوس [۸] در رمزگشا برای افزایش ابعاد تصویر خارج شده از رمزگذار به اندازه اصلی استفاده می‌شوند. در حالت عادی، این معماری قدرت بازسازی بهینه‌ای نداشته و برخی اطلاعات تصویر اصلی از دست خواهد رفت که شدت آن به میزان استفاده از تعداد لایه های ادغامی و شدت کوچک‌نمایی تصویر دارد. برای کمرنگ‌تر کردن این مشکل، فقط از داده‌هایی که مستقیماً از لایه کانولوشن معکوس پیشین می‌آیند استفاده نمی‌کنیم، بلکه آن‌ها با دانش سطح پایین از لایه‌های زیرین رمزگذار ترکیب می‌شود. این کار به تشخیص ساختارهای با وضوح بالاتر کمک می‌کند که در تمیز کردن بهتر مرزهای اشیاء موثر است. پس از محاسبه کانولوشن‌های لایه فعلی و زیرین، هر دو ویژگی بدست آمده با یکدیگر ترکیب شده و سپس بزرگ‌نمایی می‌شود که به این واحد ها اصطلاحا واحد تصحیح [۹] گفته می‌شود. پیش‌تر اشاره شد که توابع‌فعال‌ساز متفاوتی برای رمزگذار استفاده می‌شود. این تابع فعال‌ساز برای بخش رمزگشا نیز به همین نحو استفاده می‌شود.

۳-۱-۲ مدل ENet

معماری‌های متعددی برای حل مسایل تقسیم‌بندی معنایی مطرح شده‌اند که FCN و SegNet [۱۰] دو مدل مطرح در این حوزه هستند. از آنجایی که هر دو معماری بر اساس معماری پایه VGG طراحی شده‌اند، تعداد پارامترها و زمان استنتاج بالایی دارند و برای استفاده در حوزه‌هایی که نیاز به پردازش سریع و یا سخت‌افزار ضعیفی دارند مناسب نیستند. مدل ENet [۱۱] [۱۰] با هدف پردازش سریع‌تر و دقت بالا طراحی شده است که نیز از معماری رمزگذار-رمزگشا استفاده می‌کند.

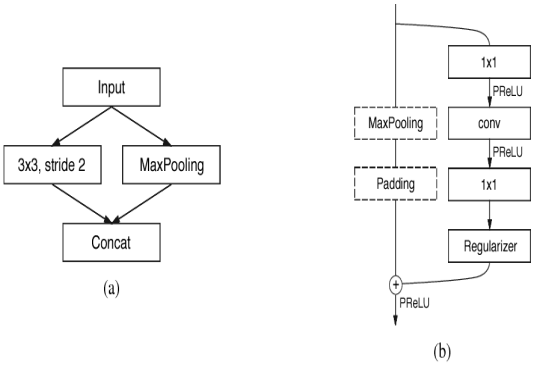
معماری ENet از چندین بلوک تشکیل شده. بلوک ابتدایی [۱۱] شامل یک لایه ادغام حداکثری با

[7] Translation Invariant
[8] Transposed Convolutions
[9] Refinement Module
[10] Efficient Neural Network
[11] Initial block

پنجره‌های ۲×۲ بدون همپوشانی و یک لایه کانولوشنی با ۱۳ فیلتر است که تصویر را به ۱۶ نقشه ویژگی تبدیل می‌کند. هدف استفاده از این بلوک، کاهش ابعاد تصویر و تبدیل آن به بردارهای ویژگی است تا اطلاعات غیرمرتبط تصویر حذف شده و بار محاسباتی کاهش پیدا کند. بلوک گلوگاه از اجزای کلیدی این معماری است که به طور مکرر در بخش های مختلف این معماری شاهد آن هستیم.

Name	Type	Output size
initial		$16 \times 256 \times 256$
bottleneck1.0	downsampling	$64 \times 128 \times 128$
$4\times$ bottleneck1.x		$64 \times 128 \times 128$
bottleneck2.0	downsampling	$128 \times 64 \times 64$
bottleneck2.1		$128 \times 64 \times 64$
bottleneck2.2	dilated 2	$128 \times 64 \times 64$
bottleneck2.3	asymmetric 5	$128 \times 64 \times 64$
bottleneck2.4	dilated 4	$128 \times 64 \times 64$
bottleneck2.5		$128 \times 64 \times 64$
bottleneck2.6	dilated 8	$128 \times 64 \times 64$
bottleneck2.7	asymmetric 5	$128 \times 64 \times 64$
bottleneck2.8	dilated 16	$128 \times 64 \times 64$
Repeat section 2, without bottleneck2.0		
bottleneck4.0	upsampling	$64 \times 128 \times 128$
bottleneck4.1		$64 \times 128 \times 128$
bottleneck4.2		$64 \times 128 \times 128$
bottleneck5.0	upsampling	$16 \times 256 \times 256$
bottleneck5.1		$16 \times 256 \times 256$
fullconv		$C \times 512 \times 512$

(آ) معماری کلی شبکه ENet



(ب) معماری بلوک های شبکه ENet

شکل ۳-۲: نمونه تبدیل نقشه تقسیم‌بندی شده به تصویر رنگارنگ متناظر

نمونه‌برداری کاهشی [۱۲] به طور کلی منجر به از دست رفتن برخی از اطلاعات داخل تصویر می‌شود و انجام آن بخصوص به صورت مکرر و با ضریب بزرگ به ضرر مدل است. اما از طرفی کاهش ابعاد تصویر به مصرف حافظه کمتر و کاهش بار محاسباتی و تعداد پارامتر های مدل کمک شایانی می‌کند. استراتژی استفاده شده در این معماری بدین گونه است که نمونه‌برداری کاهشی به کمترین تعداد ممکن و در ابتدای مدل انجام شود. مزیت انجام این‌کار در اول مسیر آن است که از پردازش تصاویر با ابعاد بزرگ که هزینه پردازش بالایی دارند جلوگیری می‌شود. برای سرعت بخشیدن به این فرآیند، عملیات ادغام به همراه یک کانولوشن به صورت موازی انجام شده و بردارهای ویژگی حاصل با یکدیگر ترکیب می‌شوند.

بهینه سازی دیگر بر روی تابع‌فعالساز است. به گونه‌ای که در تمام مدل تابع فعالساز توابع‌فعال‌ساز یکسوساز پارامترسازی شده [۱۳] جایگزین توابع‌فعال‌ساز یکسوساز شده است. این تابع‌فعالساز شیب منفی قابل آموزش دارد که به مدل انعطاف‌پذیری بیشتر و عملکرد بهتری داشته باشیم.

[12] Downsampling

[13] Parameterized ReLU (PReLU)

در آخر، استفاده از کانولوشن‌های گسترده [۱۴] نوعی دیگر از عملیات کانولوشن هستند که در آن‌ها فاصله بین پیکسل‌های ورودی افزایش می‌یابد. در واقع، این نوع از کانولوشن‌ها به ورودی‌ها اعمال می‌شوند با استفاده از یک فیلتر کانولوشن با فضای پیکسل‌های بزرگ‌تر از یک باعث می‌شود اطلاعات بیشتری از ورودی‌ها در نظر گرفته شود. این نوع از کانولوشن‌ها معمولاً این امکان را برای شبکه فراهم می‌کنند تا بدون افزایش تعداد پارامترها میدان تاثیر بزرگ‌تری داشته باشد.

۳-۲ معماری دو-شاخه

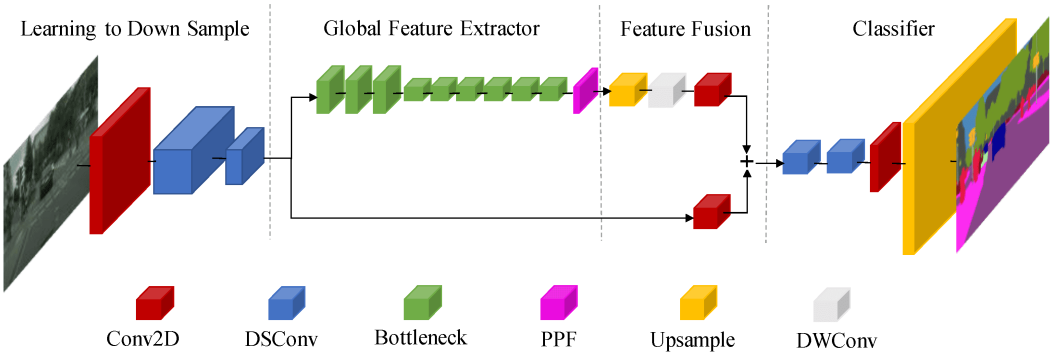
۳-۲-۱ مدل Fast-SCNN

به مرور، تمایل به استفاده از معماری دو-شاخه [۱۵] در مدل‌های مطرح شده برای تقسیم‌بندی معنایی سریع افزایش یافته است؛ به طوری که دو شبکه با عمق های متفاوت بر روی تصویر با وضوح‌های متفاوت عمل کرده و در نهایت هر دو شاخه با یکدیگر ترکیب می‌شوند. این معماری اجازه می‌دهد تا در یک شاخه از شبکه‌ای عمیق [۱۶] بر روی تصویر با وضوح پایین استفاده شود تا اطلاعات اشیاء استخراج و آموخته شود و در شاخه دیگر شبکه‌ای کم عمق [۱۷] بر روی همان تصویر اما با وضوح بالاتر به کار گرفته شود تا دقت نهایی تصویر خروجی بهبود یابد. از آنجایی که عمق شبکه و ابعاد تصویر اولیه به طور مستقیم بر روی سرعت پردازش تاثیرگذار هستند، معماری دو-شاخه بهینه‌سازی‌هایی برای پردازش سریع‌تر نسبت به معماری رمزگذار-رمزگشا دارد. مدل SCNN-Fast از ۴ بخش تقسیم شده که به صورت سری به یکدیگر متصل شده اند. در ادامه به معرفی اجزای مورد استفاده در این مدل می‌پردازیم.

لایه کانولوشنی تفکیک‌پذیر عمق‌محور

در یک کانولوشن استاندارد بر روی تصاویر رنگی که عموما ۳ کانال رنگ دارند اینگونه انجام می‌شود که فیلتر به اندازه عمق رنگ ورودی اعمال شده و به ما امکان می‌دهد که کانال‌های رنگی را ترکیب کرده و آن‌ها را کم یا زیاد کنیم. به عبارتی اگر بخواهیم یک تصویر با ابعاد (۱۲،۱۲،۳) را به (۸،۸،۲۵۶) تبدیل کنیم به ۲۵۶ کرنل با ابعاد (۵،۵،۳) نیاز خواهیم داشت که در مجموع کمی بیش از یک میلیون عملیات

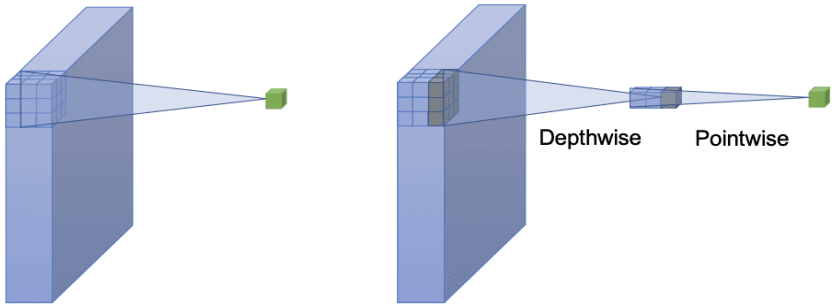
[14] Dilated convolutions
[15] Two-branch architecture
[16] Deep network
[17] Shallow network



شکل ۳-۳: معماری مدل Fast-SCNN

ضرب خواهیم داشت.

در عملیات کانولوشنی عمق‌محور [18] ، هر فیلتر به صورت جداگانه بر روی هر کانال اعمال شده و در نتیجه تعداد کانال‌های تصویر ثابت می‌ماند. به عبارتی، در تبدیل تصویر با ابعاد مشابه به ابعاد ثانوی (۸،۸،۳) نیاز به ۳ کرنل (به تعداد کانال‌های تصویر) با ابعاد (۵،۵،۱) داریم که در مجموع تقریبا ۵۰۰۰ عملیات ضرب می‌شود. سپس برای افزایش تعداد کانال‌های تصویر به عملیات کانولوشن نقطه‌محور [19] [۱۲] نیاز خواهیم داشت. برای مثال افزایش تعداد کانال‌های تصویر از ۳ به ۲۵۶ نیازمند ۲۵۶ کرنل با ابعاد (۱،۱،۳) دارد که در مجموع ۵۰۰۰۰ عملیات ضرب می‌شود. در نهایت ترکیب این دو لایه که لایه کانولوشنی تفکیک‌پذیر عمق‌محور [20] [۱۳، ۱۴] نام دارد، معادل عملیات کانولوشن استاندارد می‌شود. لایه جدید در تئوری ۲۵ برابر و در عمل ۲ الی ۸ برابر سریع تر از کانولوشن استاندارد است.



شکل ۳-۴: مقایسه کانولوشن استاندارد و تفکیک‌پذیر عمق‌محور

[18] Depthwise Convolution
[19] Pointwise Convolution
[20] Depthwise Separable Convolutions

بخش learning to downsample

این اولین بخش از معماری SCNN-Fast است که از سه لایه اصلی تشکیل شده است که اولین آنها لایه کانولوشنی استاندارد است و دو جزء دیگر لایه‌های کانولوشنی تفکیک‌پذیر عمق‌محور که پیش‌تر معرفی شدند نام دارند. لایه‌های کانولوشنی تفکیک‌پذیر عمق‌محور در حالت عادی به لحاظ محاسباتی کارآمدتر هستند، اما برای اولین لایه از لایه کانولوشن استاندارد استفاده می‌کنیم زیرا برتری محاسباتی این لایه در اولین لایه به دلیل تنها ۳ کاناله بودن تصویر بیشتر است. پس از تمامی لایه‌های اصلی از نرمال‌سازی دسته‌ای و تابع‌فعال‌ساز ReLU استفاده شده است. معماری این بخش در شکل ۳-۳ قابل مشاهده است.

بخش استخراج ویژگی‌های سراسری

بخش استخراج ویژگی‌های سراسری [۲۱] به دنبال استخراج اطلاعات از فضای تصویر برای تقسیم‌بندی است. تصویر ورودی این جزء، خروجی مستقیم بخش learning␣to␣downsample است که معادل یک هشتم ابعاد تصویر اصلی را دارد. این کوچک‌نمایی در عین کاهش میزان محاسبات، اکثر جزئیات مهم تصویر را حفظ می‌کند. در این بخش از تعدادی لایه بلوک اضافی گلوگاه [۲۲] استفاده می‌شود که در آنها لایه کانولوشنی تفکیک‌پذیر عمقی جایگزین لایه‌های کانولوشنی عادی شده تا تعداد وزن‌های مورد استفاده در هر بلوک و در نتیجه تعداد عملیات برای محاسبه خروجی کاهش یابد. در آخر از لایه ادغام هرمی [۲۳] [۱۵] استفاده شده که تا اطلاعات موجود در تصویر در مقیاس‌های مختلف تجمیع شوند که با استفاده از آنها پس از بلوک های اضافی گلوگاه تاثیر مثبتی بر روی خروجی می‌گذارد. نمای کلی این بخش در شکر ۳-۳ قابل مشاهده است.

گذاخت ویژگی‌ها و دسته‌بندی

در بخش گداخت ویژگی‌ها [۲۴]، از جمع ویژگی‌های به دلیل بهره‌وری بالای آن استفاده شده است. هرچند می‌توان از عملیات ترکیبی دیگر برای افزایش دقت استفاده کرد، در این معماری از جمع استفاده شده است. در بخش آخر به ترتیب از دو لایه DSConv و یک لایه Conv2D استفاده شد تا تصویر به اندازه اصلی برگردانده شود و در آخر از لایه softmax برای برگرداندن دسته‌بندی استفاده شد. به دلیل هزینه‌بر

[21] Global feature extractor
[22] Bottleneck residual block
[23] Pyramid pooling module (PPM)
[24] Feature fusion module

بودن محاسبات این تابع، می‌توان آن را با تابع argmax جایگزین کرد تا سرعت پردازش افزایش یابد.

۳-۳ خلاصه

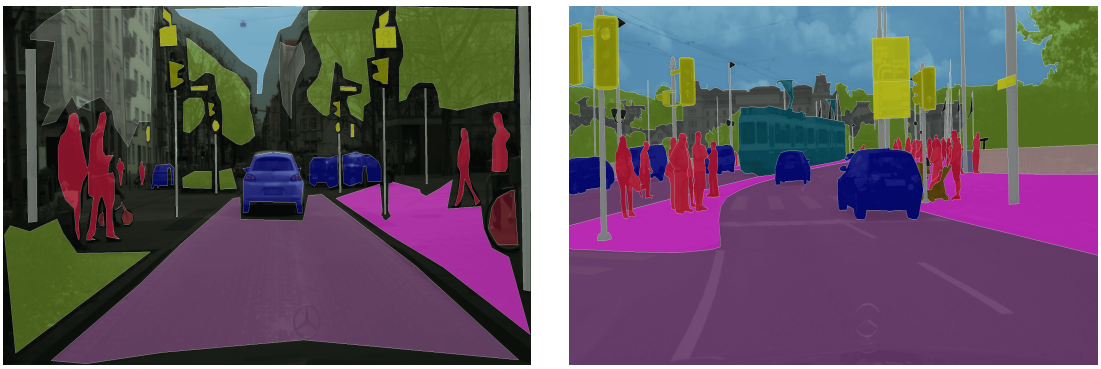
فصل چهارم

آزمایش‌ها و نتایج

۴-۱ دادگان

۴-۱-۱ مجموعه داده Cityscapes

مجموعه داده‌های Cityscapes یکی از پرکاربردترین مجموعه‌های داده برای مسائل تقسیم‌بندی معنایی است، که بر روی درک مفهومی صحنه‌های خیابانی شهری تمرکز دارد. این مجموعه شامل ۵۰۰۰ تصویر با برچسب‌گذاری دقیق و ۲۰۰۰۰ تصویر با برچسب‌گذاری خشن است، که برای ۳۰ کلاس معنایی مختلف آموزش دیده‌اند. تصاویر زیر مقایسه‌ای بین دو نوع برچسب‌گذاری ارائه می‌دهند.



(آ) برچسب‌گذاری دقیق (ب) برچسب‌گذاری خشن

شکل ۴-۱: انواع برچسب‌گذاری دادگان Cityscapes

ما در این پژوهش از ۵۰۰۰ تصویر با برچسب‌گذاری شده به شیوه دقیق استفاده خواهیم کرد، چراکه مراجع مورد استفاده قرار گرفته نیز از این نوع برچسب‌گذاری برای مقایسه استفاده کرده اند.

۴-۱-۲ مجموعه داده CamVid

در سال ۲۰۰۷، پایگاه داده ویدیویی با برچسب‌گذاری شهری کمبریج (CamVid)، از اولین مجموعه‌های داده تقسیم‌بندی معنایی برای خودروهای خودران، منتشر شد که در آن، ۷۰۰ تصویر از یک دنباله ویدیویی با مدت زمان ۱۰ دقیقه برچسب‌گذاری شد. برای گرفتن ویدیو، دوربین در جلوی ماشین قرار گرفته که دیدگاه مشابهی با راننده دارد. در این مجموعه داده ۳۲ دسته بندی معنایی وجود دارد.

۴-۲ معیارهای ارزیابی

زمان استنتاج

برای اندازه‌گیری زمان استنتاج از معیار fps [۱] استفاده می‌کنیم تا سرعت زمان استنتاج مدل‌ها را با یکدیگر بررسی کنیم. به طبع هر چه fps بالاتری داشته باشیم برای ما مطلوب‌تر خواهد بود.

بهره‌وری منابع

در این معیار به سه مشخصه زیر می‌پردازیم تا دید بهتری به مقیاس هر مدل پیدا کنیم:

- تعداد پارامتر ها: هر چه تعداد نورون‌های بیشتری داشته باشیم مدل سنگین تر می‌شود. تلاش بر آن است که مدل پیشنهادی سبک‌وزن (lightweight) باشد.
- میزان مصرف مموری: متناسب با پیچیدگی مدل در تعداد و نوع عملیات‌ها میزان مصرف مموری در حین اجرا متغیر است.
- میزان مصرف حافظه: این مقدار با تعداد پارامترها نسبت مستقیم دارد، اما دید شهودی به مقیاس هر مدل می‌دهد.

میانگین اشتراک بر اجتماع

میانگین اشتراک بر اجتماع [۲] یک معیار پرکاربرد در مسائل بینایی ماشین [۳] است که برای ارزیابی عملکرد مدل‌ها مورد استفاده قرار می‌گیرد و به وسیله محاسبه میزان تطابق بین ماسک تشخیص [۴] شیء پیش‌بینی شده توسط مدل و ماسک واقعی در تصویر عمل می‌کند. برای محاسبه این معیار، ابتدا IoU یا اشتراک بر اجتماع برای هر شیء در تصویر محاسبه می‌شود، سپس از آن‌ها میانگین گرفته می‌شود تا عملکرد کلی مدل در تشخیص شیء مورد ارزیابی قرار گیرد.

در اینجا اشتراک بر اجتماع هر دسته و میانگین کلی به صورت جدا سنجیده و مقایسه می‌شود.

[1] Frame per second

[2] Mean intersection-over-union (Mean IoU)

[3] Computer vision

[4] Prediction mask

۴-۳ شرایط آزمایش

برای ایجاد یک مقایسه عادلانه، دادگان مورد استفاده قرار گرفته را به دو بخش مجموعه‌داده آموزشی [۵] و مجموعه‌داده صحبت‌سنجی [۶] تقسیم کردیم. تقسیم‌بندی به همانگونه که در دادگان اولیه انجام‌شده بود نگه‌داشته شد تا در مقایسه با مقاله‌های معتبر دچار مشکل نشویم. تمامی مدل‌های پیاده سازی شده در چهارچوب پیاده‌سازی پایتورچ [۷] پیاده‌سازی شده اند و تمامی آن‌ها بر روی کارت گرافیکی NVIDIA GeForce GTX 3060 6GB انجام شده است.

۴-۴ نتایج آزمایش و مقایسه

در ابتدا به بررسی سرعت پردازش (fps) پرداخته می‌شود. در آزمایش فرض شده ۱۰ دسته‌بندی اشیاء داشته و تمامی تصاویر دارای ۳ کانال رنگی هستند. به دلیل محدودیت سخت‌افزاری روی پردازنده گرافیکی این مقایسه بر روی تصاویر با ابعاد ۲۰۴۸ در ۱۰۲۴ انجام نشده است. نتایج در جدول زیر قابل مشاهده است.

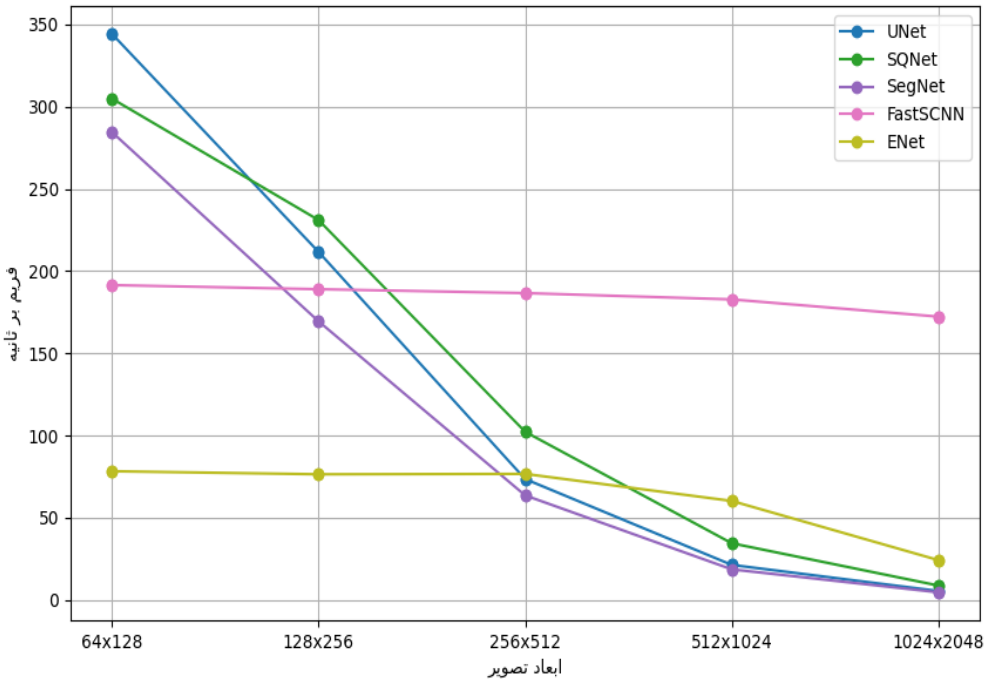
کارت گرافیک	ابعاد / مدل	FastSCNN	SQNet	ENet	UNet	SegNet
RTX 3060	64x128	۹۶٫۱۸	۱۳۹٫۷۳	۳۸٫۸۱	۱۶۸٫۷۴	۱۴۰٫۵۶
	128x256	۱۰۰٫۰۷	۱۲۳٫۰۷	۳۸٫۱۰	۱۳۰٫۱۱	۱۰۷٫۹۶
	256x512	۹۸٫۹۴	۶۳٫۸۷	۳۶٫۰۲	۴۵٫۱۲	۳۹٫۹۲
	512x1024	۹۷٫۹۳	۲۱٫۹۴	۳۰٫۷۵	۱۳٫۷۱	۱۱٫۹۱
Tesla P100	64x128	۱۹۱٫۵۲	۳۰۴٫۹۶	۷۸٫۳۲	۳۴۴٫۳۲	۲۸۴٫۵۹
	128x256	۱۸۹٫۰۴	۲۳۱٫۰۹	۷۶٫۴۷	۲۱۱٫۷۳	۱۶۹٫۵۱
	256x512	۱۸۶٫۶۷	۱۰۲٫۳۰	۷۶٫۷۰	۷۳٫۵۸	۶۳٫۶۴
	512x1024	۱۸۲٫۸۳	۳۴٫۵۲	۶۰٫۲۵	۲۱٫۳۶	۱۸٫۶۵
	1024x2048	۱۷۲٫۳۴	۸٫۷۹	۲۴٫۲۵	۵٫۴۶	۴٫۶۷

جدول ۴-۱: مقایسه شاخص fps روی کارت گرافیکی‌های متفاوت

با مقایسه نتایج جدول ۴-۱ می‌توان مشاهده می‌شود مدل FastSCNN بر روی تصاویر با ابعاد بزرگ

[5] Training set
[6] Validation set
[7] PyTorch framework

بهتر از مدل‌های دیگر عمل می‌کند و مدل SQNet برای تصاویر کوچک‌تر بهت عمل می‌کند. علت نتایج متفاوت به ضعیف‌تر نسبت به مقادیر ارائه شده در مقاله‌های اصلی، کارت گرافیکی متفاوت (و ضعیف‌تر) استفاده شده و تفاوت‌هایی در پیاده‌سازی‌های صورت گرفته است. این تفاوت‌ها در مقایسه تعداد پارامترهای مدل‌های پیاده‌سازی شده که در جدول ۴-۲ قابل مشاهده است مشهود است. شکل زیر نمایش بهتری از روند تغییر فریم بر ثانیه نشان می‌دهد:



شکل ۴-۲: روند تغییر FPS بر حسب افزایش ابعاد تصویر

در گام بعد، میزان منابع مورد نیاز برای اجرای مدل‌های معرفی شده برای تصاویر رنگی با ابعاد ۲۵۶ در ۵۱۲ پیکسل بر روی ۱۰ دسته بررسی می‌شود. همانطور که مشخص شده مدل FastSCNN از بقیه مدل‌ها سبک‌تر بوده و کاندید مناسب‌تری برای استفاده در سامانه‌های نهفته [8] است.

در آخر شاخص میانگین اشتراک بر اجتماع را بر روی دسته‌بندی ها و گروه ها محاسبه شده است. مجموعه‌داده cityscapes دارای ۳۰ دسته و ۷ گروه هست که به طور مستقل شاخص اشتراک بر اجتماع را بر روی آن‌ها محاسبه می‌کنیم.

...

[8] Embedded systems

مدل / شاخص	تعداد پارامتر ها (میلیون)	اندازه مدل (گیگابایت)
FastSCNN	۱٫۱۴	۰٫۲۱
SQNet	۱۶٫۲۵	۰٫۵۳
ENet	۰٫۳۶	۱٫۱۷
UNet	۱۳٫۴۰	۱٫۸۱
SegNet	۲۹٫۴۵	۱٫۰۲

جدول ۴-۲: مقایسه میزان مصرف منابع

مدل / شاخص	IoU دسته	IoU گروه
FastSCNN	۶۸٫۰	۸۴٫۷
SQNet	۵۹٫۸	۸۴٫۳۰
ENet	۵۸٫۳	۸۰٫۴
UNet	۷۷٫۵	۸۳٫۸
SegNet	۵۶٫۱	۷۹٫۸

جدول ۴-۳: مقایسه شاخیص اشتراک بر اجتماع

۴-۵ خلاصه

در آزمایش‌های انجام شده برای ارزیابی عملکرد مدل‌های SQNet ، FastSCNN ، ENet و SegNet بر روی مجموعه داده‌های Cityscapes برای مسأله تقسیم‌بندی زمینه‌ای در زمینه خودروهای خودران، یافته‌های مهمی به دست آمد. معیارهای ارزیابی استفاده شده شامل تعداد فریم در ثانیه، میانگین اشتراک بر اجتماع هر کلاس و هر دسته، تعداد پارامترها و اندازه کلی مدل بود.

نتایج نشان می‌دهد که مدل FastSCNN می‌تواند با تغییر ابعاد تصویر، تعداد فریم‌های پردازش شده در ثانیه را ثابت نگه دارد. این در حالی است که در مدل‌های دیگر با افزایش اندازه تصویر، کاهش چشم‌گیری در این شاخص دیده می‌شود. همچنین، با وجود اینکه FastSCNN کمترین تعداد پارامترها را نداشت، به طور کلی حافظه کمتری نسبت به سایر مدل‌های با معماری رمزگذار-رمزگشا داشت و دقت نسبتاً بالایی در شاخص اشتراک بر اجتماع در هر دو کلاس و دسته نشان داد.

در کل، مشاهده شد که معماری دو شاخه‌ای نسبت به معماری رمزگذار-رمزگشا از نظر سرعت و دقت عملکرد بهتری نشان می‌دهد.

فصل پنجم

نتیجه گیری، جمع بندی و پیشنهادات

۵-۱ جمع‌بندی و نتیجه‌گیری

در این پژوهش به بررسی معماری‌های مختلف برای حل مسئله تقسیم‌بندی معنایی تصاویر در حوزه خودروهای خودران با استفاده از روش های یادگیری عمیق پرداختیم. همانطور که در مقدمه به طور مفصل‌تر به آن پرداخته شد، مدل‌های مورد استفاده برای این حوزه بخصوص، علاوه بر دقت نیاز به سرعت عمل بالا نیز دارند که معیار مهمی در ارزیابی نهایی آنها است.

در فصل دوم، به مطالعه مفاهیم پرتکرار این حوزه پرداخته و معماری‌های مورد استفاده، نظیر معماری رمزگذار-رمزگشا، را معرفی کردیم که کاربرد گسترده ای در مدل‌های مطرح برای این حوزه دارد و به جزئیات آن پرداختیم. سپس چندین معماری مطرح در حوزه تقسیم‌بندی معنایی را معرفی کردیم که از دقت بالایی برخوردار بوده، اما عملکرد خوبی در پردازش آنی ندارند که مشخصه مهمی در ارزیابی نهایی است.

در فصل سوم، به طور عمیق وارد معماری منحصر به فرد مدل‌های پیشنهادی و اجزای کلیدی آنها شدیم و نقاط ضعف و قوت هر یک را بررسی و مقایسه کردیم. با وجود مدل‌های متعدد در حوزه تقسیم‌بندی معنایی، می‌توان گفت اکثر مدل‌ها از معماری رمزگذار-رمزگشا و یا مشابه آن استفاده می‌کنند تا بتوانند عملکرد بهینه‌تری در سرعت پردازش بدست آورند. متوجه شدیم با تغییر بر روی اجزای این معماری، مانند تعداد لایه ها، نوع توابع‌فعال‌ساز، حذف نرمال‌سازی، و موارد مشابه می‌توان بر تعداد وزن‌های مورد نیاز و میزان محاسبات لازم را کاهش داد و در نهایت بر روی سرعت پردازش تاثیر مثبت گذاشت. همچنین می‌توان با تغییر در معماری مانند افزودن پرش، ترکیب داده‌های لایه‌ها و استفاده از معماری دو-شاخه بر، دقت و کیفیت تصویر بازسازی شده را بهبود داد.

در فصل چهارم، آزمایش‌های و نتایج، به آموزش و ارزیابی معماری‌های مطرح شده پرداختیم. ارزیابی‌های صورت گرفته نه تنها بر روی دقت و شاخص بازسازی تصاویر بود، بلکه بر روی سرعت پردازش و مصرف منابع مدل‌ها نیز تمرکز داشتیم. طی مقایسه عملکرد مدل‌ها متوجه شدیم استفاده از معماری دو-شاخه در معماری اولیه رمزگذار-رمزگشا تاثیر مثبتی بر روی سرعت پردازش می‌گذارد و در عین حال دقت مدل دچار نوسان چندانی نمی‌شود که مطلوب ما است.

۵-۲ پیشنهادات و کارهای آتی

روش‌های مورد بحث و بررسی قرار گرفته داخل این پروژه همگی بر روی تصاویر تمرکز داشته‌اند؛ به گونه‌ای که برای پردازش ویدیو، هر فریم به تنهایی و مجزا از فریم‌های پیشین پردازش می‌شود. چالش روش فعلی آن است که امکان تغییر دسته‌بندی ها بین دو تصویر متوالی در یک ویدیو وجود دارد و سازوکاری برای کاهش و یا جلوگیری از آن نداریم. مدل‌های نوین‌تر تقسیم‌بندی معنایی ویدیویی [1] نظیر TMANET [۱۶] [2] به حل این مشکل می‌پردازند. در این گونه مدل های، یک یا چند تصویر گذشته بر روی تقسیم‌بندی معنایی تصویر بعدی، به صورت وزن‌دار، تاثیرگذار هستند و بنابراین امکان تغییر ناگهانی یک دسته به دلیل خطای مدل و یا شرایط جدید محیطی کاهش میابد. هرچند استفاده از این گونه مدل‌های ویدیووی در حوزه خودروهای خودران مانند مدل‌های تقسیم‌بندی تصویر مرسوم نیست، تمرکز بیشتر بر روی این مدل ها و بهینه‌سازی آنها برای پردازش سریع‌تر پیشنهاد می‌شود.

[1] Video semantic segmentation

[2] Temporal Memory Attention Network

منابع و مراجع

[1] O'shea, Keiron and Nash, Ryan. An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458, 2015.

[2] Sutskever, Ilya, Vinyals, Oriol, and Le, Quoc V. Sequence to sequence learning with neural networks. Advances in neural information processing systems, 27, 2014.

[3] Long, Jonathan, Shelhamer, Evan, and Darrell, Trevor. Fully convolutional networks for semantic segmentation. in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431–3440, 2015.

[4] Simonyan, Karen and Zisserman, Andrew. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[5] Drozdzal, Michal, Vorontsov, Eugene, Chartrand, Gabriel, Kadoury, Samuel, and Pal, Chris. The importance of skip connections in biomedical image segmentation. in International workshop on deep learning in medical image analysis, international workshop on large-scale annotation of biomedical data and expert label synthesis, pp. 179–187. Springer, 2016.

[6] Ronneberger, Olaf, Fischer, Philipp, and Brox, Thomas. U-net: Convolutional networks for biomedical image segmentation. in Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18, pp. 234–241. Springer, 2015.

[7] Mao, Xiaojiao, Shen, Chunhua, and Yang, Yu-Bin. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. Advances in neural information processing systems, 29, 2016.

[8] Iandola, Forrest N, Han, Song, Moskewicz, Matthew W, Ashraf, Khalid, Dally, William J, and Keutzer, Kurt. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size. arXiv preprint arXiv:1602.07360, 2016.

[9] Treml, Michael, Arjona-Medina, José, Unterthiner, Thomas, Durgesh, Rupesh, Friedmann, Felix, Schuberth, Peter, Mayr, Andreas, Heusel, Martin, Hofmarcher, Markus, Widrich, Michael, et al. Speeding up semantic segmentation for autonomous driving. 2016.

[10] Badrinarayanan, Vijay, Kendall, Alex, and Cipolla, Roberto. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence, 39(12):2481–2495, 2017.

[11] Paszke, Adam, Chaurasia, Abhishek, Kim, Sangpil, and Culurciello, Eugenio. Enet: A deep neural network architecture for real-time semantic segmentation. arXiv preprint arXiv:1606.02147, 2016.

[12] Hua, Binh-Son, Tran, Minh-Khoi, and Yeung, Sai-Kit. Pointwise convolutional neural networks. in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 984–993, 2018.

[13] Chollet, François. Xception: Deep learning with depthwise separable convolutions. in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1251–1258, 2017.

[14] Nascimento, Marcelo Gennari do, Fawcett, Roger, and Prisacariu, Victor Adrian. Dsconv: Efficient convolution operator. in Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 5148–5157, 2019.

[15] Zhao, Hengshuang, Shi, Jianping, Qi, Xiaojuan, Wang, Xiaogang, and Jia, Jiaya. Pyramid scene parsing network. in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2881–2890, 2017.

[16] Wang, Hao, Wang, Weining, and Liu, Jing. Temporal memory attention for video semantic segmentation. in 2021 IEEE International Conference on Image Processing (ICIP), pp. 2254–2258. IEEE, 2021.

Abstract

Autonomous vehicles require a precise understanding of their surroundings to make informed decisions and navigate safely in various environments. Semantic segmentation, from its inception, stands as one of the fundamental stages in the process of analyzing images and extracting useful information for decision-making in such systems. It plays a vital role in detecting environmental objects, enabling the accurate identification of various entities including roads, pedestrians, other vehicles, and obstacles. Deep learning methods have significantly improved semantic segmentation, surpassing traditional approaches in performance. This project delves into recent advancements in semantic image segmentation for autonomous vehicles using deep learning methods. We investigate various architectures of deep learning in the context of rapid semantic segmentation, comparing their strengths and weaknesses for the specific task of autonomous driving. Additionally, the datasets used for training and evaluating semantic segmentation models in this domain are scrutinized, employing them to assess different deep learning models. In conclusion, a summary of the examined models is provided, along with suggestions for future research aimed at enhancing the sustainability, efficiency, and general applicability of real-time semantic segmentation systems based on deep learning for autonomous vehicles.

Key Words:

Artificial intelligence, Self-driving cars, Deep learning, Semantic segmentation, Fast image semantic segmentation

Amirkabir University of Technology
(Tehran Polytechnic)

Department of Computer Engineering

B. Sc. Project

Image Semantic Segmentation for Autonomous Driving with Deep Learning

By

Keivan Ipchi Hagh

Supervisor

Dr. Ehsan Nazerfard

March 2024