

لَهُ مَنْ فِي السَّمَاوَاتِ وَالْأَرْضِ
وَمَا يَرَىٰ إِلَّا مَا أَنْشَأَ
وَإِنَّهُ لَعَلِيٌّ وَكَفِيلٌ



تشخیص مکان گل زعفران

پروژه درس تصویرپردازی رقمی

امیرمهدی نیکوکاران (۹۹۷۲۲۱۴۸)

امیرحسین امینی‌مهر (۹۹۷۲۲۱۶۲)

آخرین ویرایش: ۲۹ بهمن ۱۳۹۹ در ساعت ۱۹ و ۴۶ دقیقه

فهرست مطالب

۲	فهرست اشکال
۳	فصل ۱ مقدمه
۴	فصل ۲ تشخیص اشیا با EfficientDet
۸	فصل ۳ مساله اصلی
۱۶	فصل ۴ کارهای بیشتر
۲۰	فصل ۵ کارهای آینده
۲۱	مراجع

فهرست تصاویر

۵	مقایسه شبکه‌های هرم و بزرگی	۱.۲
۶	مقایسه شبکه‌ها	۲.۲
۷	ساختار شبکه	۳.۲
۹	نمونه چارچوب	۱.۳
۱۰	مربع‌های ۳۰ در ۳۰ با ۱۸ بازه برای زاویه با آستانه کوچک	۲.۳
۱۰	مربع‌های ۳۰ در ۳۰ با ۱۸ بازه برای زاویه با آستانه متوسط	۳.۳
۱۱	مربع‌های ۳۰ در ۳۰ با ۱۸ بازه برای زاویه با آستانه بزرگ	۴.۳
۱۱	مربع‌های ۱۰ در ۱۰ با ۳۶ بازه برای زاویه با آستانه کوچک	۵.۳
۱۲	مربع‌های ۱۰ در ۱۰ با ۳۶ بازه برای زاویه با آستانه متوسط	۶.۳
۱۲	مربع‌های ۱۰ در ۱۰ با ۳۶ بازه برای زاویه با آستانه بزرگ	۷.۳
۱۳	مربع‌های ۸ در ۸ با ۳۶ بازه برای زاویه با آستانه کوچک	۸.۳
۱۳	مربع‌های ۸ در ۸ با ۳۶ بازه برای زاویه با آستانه متوسط	۹.۳
۱۴	مربع‌های ۸ در ۸ با ۳۶ بازه برای زاویه با آستانه بزرگ	۱۰.۳
۱۴	مربع‌های ۴ در ۴ با ۳۶ بازه برای زاویه	۱۱.۳
۱۶	مساله جورچین	۱.۴
۱۷	مربع‌های ۳۰ در ۳۰ با ۱۸ بازه برای زاویه با آستانه کوچک	۲.۴
۱۷	مربع‌های ۳۰ در ۳۰ با ۱۸ بازه برای زاویه با آستانه متوسط	۳.۴
۱۸	مربع‌های ۳۰ در ۳۰ با ۱۸ بازه برای زاویه با آستانه بزرگ	۴.۴
۱۹	جدول نتیجه	۵.۴

۱ مقدمه

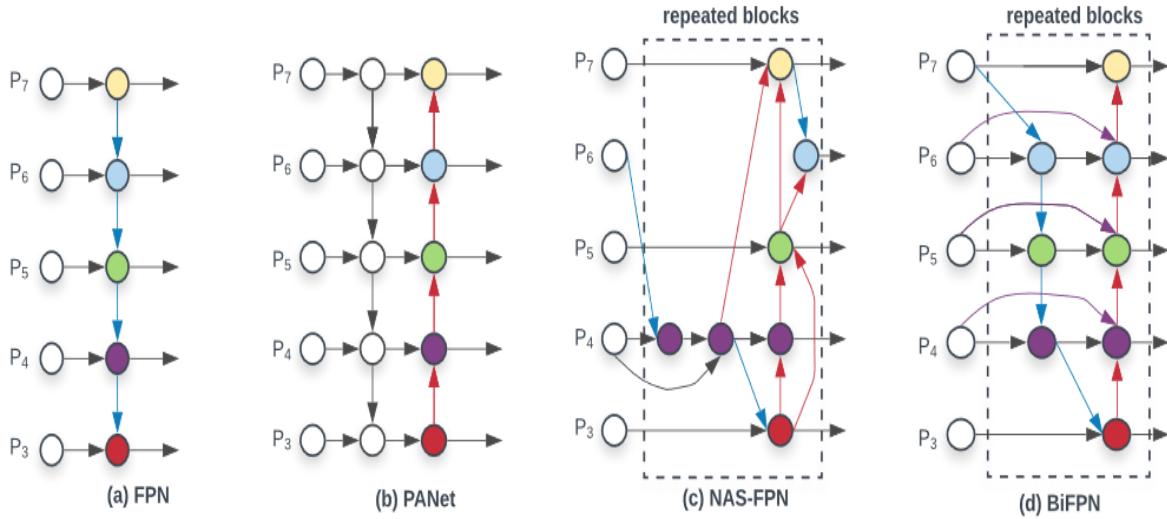
برای تهیه زعفران که محصولی بالارزش بسیار بالا به ویژه در حوزه صادرات در زمرة صوغات ویژه ایران محسوب می‌شود، فرایند چالش برانگیزی برای جداسازی میوه زعفران از گل آن وجود داشته است. اما با توجه به استفاده‌های بسیار مفید هوش مصنوعی در حوزه‌های مختلف به ویژه کاربردهای قدرتمند یادگیری عمیق، در این پژوهه سعی داشته‌ایم با استفاده از یک الگوریتم مبتنی بر یادگیری عمیق مکان و زاویه گل‌های زعفران در تصویر را تشخیص دهیم تا بتواند بهترین شکل برش را به دستگاه نشان دهد.

۲ تشخیص اشیا با EfficientDet

در این پروژه مساله تشخیص مکان و زاویه گل زعفران تعریف شده است و مقداری داده (تصاویر گل‌های زعفران) آموزشی به همراه برچسب که همان مختصات نقطه مرکز و زاویه گل هستند در اختیار قرار داده شده است و هدف آموزش شبکه برای دستیابی به دقت بالا در پیش‌بینی کردن این برچسب‌هاست تا در قسمت آزمون بتواند تا حد ممکن به بهترین دقت پیش‌بینی محل برش برسد. به همین منظور از شبکه کارآمد و با دقت بالای EfficientDet^[۱] که برای تشخیص اشیا به کار می‌رود استفاده شده که در ادامه خلاصه‌ای از مقاله آن آورده شده است.

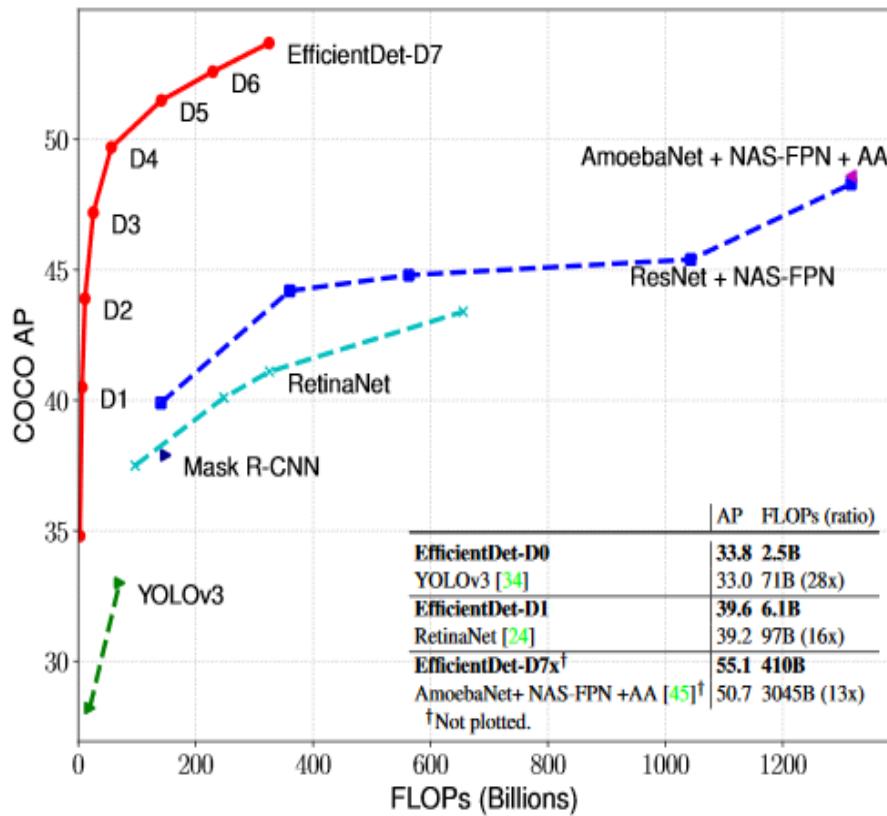
از آنجا که کارآیی شبکه‌های آموزشی روز به روز اهمیت بیشتری پیدا می‌کند، شبکه EfficientDet با هدف افزایش کارآیی در حوزه تشخیص اشیا طراحی شده است که از چندین ایده کارآمد به منظور بهینه‌سازی استفاده می‌کند تا کارآیی شبکه را بهبود بخشد. از این رو این شبکه EfficientDet نامیده شده است چون با این هدف طراحی شده که حجم محاسبات را به میزان قابل توجهی کاهش دهد و سرعت اجرا را با توجه به محدودیت منابعی که اکثریت کاربران با آن روبه‌رو هستند چه از نظر CPU یا GPU، افزایش دهد طوری که حتی روی CPU تاثیر به سزای آن مشهود است و نسبت به شبکه‌های دیگر سرعت خیلی بالاتری را فراهم می‌کند.

اولین ایده این شبکه استفاده از شبکه‌های هرم ویژگی دوچهته (BiFPN) است که قابلیت ادغام ساده و سریع ویژگی‌های با مقیاس‌های مختلف را می‌دهد و نسبت به روش‌های قبلی (تصویر ۱.۲) دقت و کارآیی بالاتری دارد. در FPN ادغام از بالا (مقیاس کمتر) به پایین (مقیاس بیشتر) صورت می‌گیرد. PANet ادغام از پایین به بالا را روی FPN اضافه کرده است. NAS-FPN ابتدا به دنبال ویژگی‌های مقیاس‌بندی شده مناسب برای ادغام می‌گردد که زمان بسیار زیادی می‌برد و بر آن اساس ادغام انجام می‌دهد و دارای بلوك تکرارشونده است ولی باز هم دقت PANet از هر دو شبکه دیگر بهتر است. شبکه BiFPN با ادغام دو ایده NAS-FPN و یعنی بلوك‌های تکرارشونده و PANet یعنی حرکت رفت و برگشتی و البته حذف گره‌های تک ورودی (چون ورودی ادغام شده ندارند) و گرفتن ورودی توسط گره‌های خروجی از طرف گره‌های لایه اصلی هم‌سطح با آن توانسته به دقت و کارآیی بالاتری نسبت به PANet برسد.



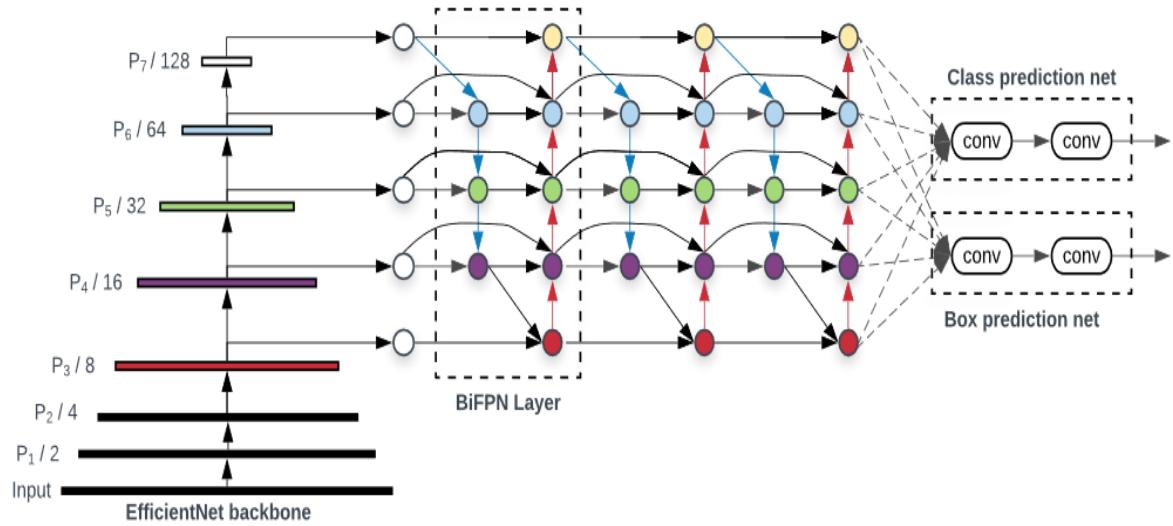
شکل ۱۰.۲: مقایسه شبکه‌های هرم ویژگی

به عنوان ایده دوم استفاده از یک روش مقیاس‌دهی ترکیبی پیشنهاد شده است که به صورت یکنواخت رزولوشن، عمق و عرض را برای همه قسمت‌های ستون فقرات (backbone) شبکه، شبکه‌های هرم ویژگی و شبکه‌های پیش‌بینی چارچوب یا دسته مقیاس‌دهی می‌کند. هم‌جنین EfficientNet از شبکه دسته‌بند کارآمد EfficientDet به عنوان ستون فقرات (backbone) خود استفاده می‌کند. برای مقایسه می‌توان گفت به طور ویژه شبکه EfficientDet-D7 به میانگین دقت سرآمد ۱.۵۵ درصد روی مجموعه داده‌های COCO با ۷۷ میلیون پارامتر و ۴۱۰ بیلیون FLOP (تعداد جمع و ضرب) که ۶ تا ۹ برابر کوچک‌تر است و ۱۳ تا ۴۲ برابر FLOP کمتر نسبت به شبکه‌های تشخیص اشیا قبلی دارد. قسمتی از این مقایسه در تصویر ۲.۲ مشهود است.



شکل ۲.۲: مقایسه شبکه‌ها

در تصویر ۳.۲ ساختار شبکه یک مرحله‌ای EfficientDet قابل ملاحظه است که قسمت‌های مختلف آن که در آن نوآوری نسبت به شبکه‌های دیگر به وجود آمده است به اختصار توضیح داده شد. از شبکه EfficientNet پیش‌آموزش داده شده روی ImageNet به عنوان ورودی استفاده شده و در چند مقیاس از آن ویژگی استخراج شده است. این ویژگی‌ها برای ادغام به BiFPN داده شده است. در نهایت دو شبکه پیش‌بینی‌کننده چارچوب و دسته با این ویژگی‌های ادغام شده آموزش می‌بینند.



شكل ٣.٢: ساختار شبکه

۳ مساله اصلی

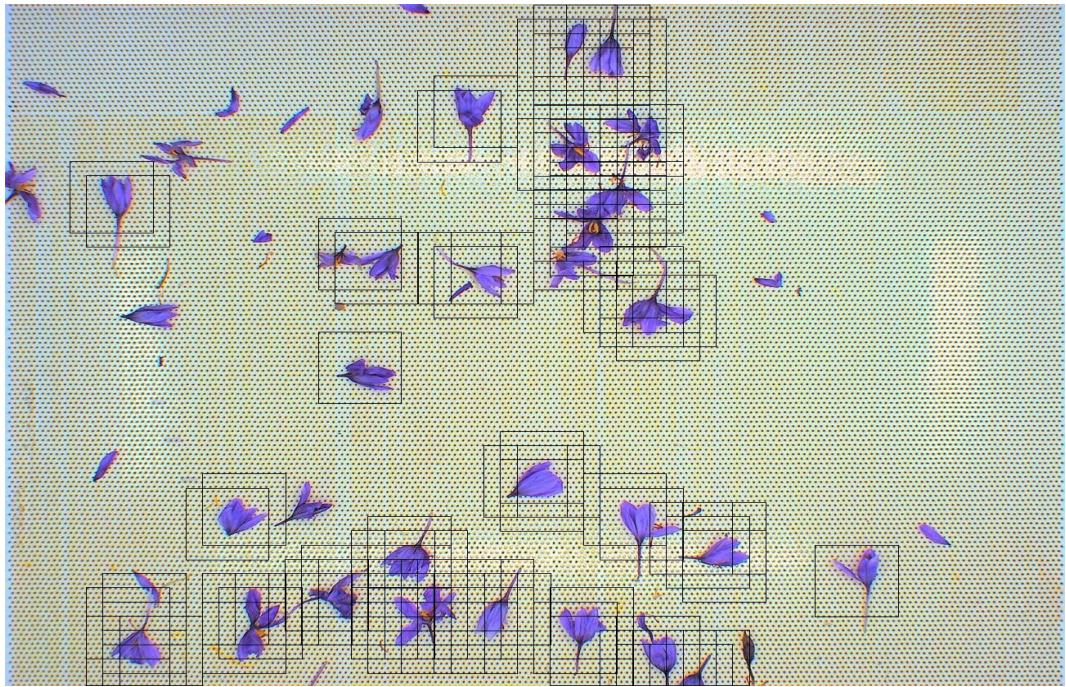
برای حل مساله اصلی از کد پیاده شده مقاله^[۲] با Keras استفاده شده است. ایده اجرا شده در این پژوهه پیدا کردن گل های زعفران

با همان روش مرسوم و با پیدا کردن یک مربع دور قسمت گل زعفران در تصویر است

از آنجایی که داده ها شامل یک نقطه و یک زاویه است باید نقطه ای را که گل زعفران در آن قرار دارد مرکز یک مربع در نظر گرفت و پیدا کردن آن مربع را به شبکه آموزش داد. البته باید مربع مورد نظر تا حد امکان کوچک باشد زیرا اگر مربع بیش از حد بزرگ باشد به جای یادگیری محل قسمت انتهایی گل محل کل گل را یاد می گیرد و همچنین اگر مربع بزرگ باشد حتی اگر پیشビینی مکان مربع توسط مدل ما دارای خطای خیلی کمی باشد ولی چون مربع پیش‌بینی شده بسیار بزرگ‌تر از یک نقطه می‌باشد (نقطه مرکز مربع که گل زعفران در آن قرار دارد) خطای کم آن باعث به وجود آمدن خطای زیادی در پیش‌بینی نقطه مرکز مربع می‌شود. پس به همین دلیل مربع پیش‌بینی شده باید تا حد امکان کوچک و کوچک‌تر باشد تا با ابعاد یک نقطه که یک در یک است نزدیک شود.

در ابتدا ابعاد مورد استفاده برای مربع که مکان گل زعفران را پیش‌بینی کند ۱۰۰ در ۱۰۰ در نظر گرفته شده که در واقع مرکز این مربع محل گل زعفران می‌باشد. چون ابعاد مربع‌ها به اندازه کافی بزرگ بودند چالش خاصی برای آموزش شبکه برای یافتن مربع‌های ۱۰۰ در ۱۰۰ که گل زعفران در مرکز آن وجود دارد نبود و مدل دقیق مناسبی نیز داشت ولی به همان دلیلی که پیش‌تر ذکر شد چون مربع‌ها خیلی بزرگ بودند نتیجه نهایی چندان مناسب نبود.

نمونه‌ای از چارچوب‌های پیش‌بینی شده توسط شبکه آموزش داده شده در تصویر زیر مشهود است:



شکل ۱.۳: نمونه چارچوب

در مرحله بعد سعی بر آموزش مدل برای پیش‌بینی مربع‌های 30×30 در گل زعفران شد که باعث به وجود آمدن یک چالش شد زیرا در آموزش مدل خطای رگرسیون برابر با صفر نشان داده می‌شد پس از جستجوهای فراوان متوجه شدیم^[۳] دلیل این اتفاق ابعاد لنگرهای anchor (anchor) می‌باشد که برای مربع‌های کوچک و شناسایی اجسام کوچک مناسب نمی‌باشند پس ابعاد لنگرهای استرایدها را طوری تنظیم کردیم که توانایی پیش‌بینی مربع‌های کوچک را داشته باشد.

برای پیش‌بینی زاویه‌ها ابتدا زاویه‌ها را به جای بازه 0° تا 360° به بازه 0° تا 180° بردهیم زیرا جهت خط (برش) اهمیت نداشت و فقط زاویه خط اهمیت دارد. در مرحله بعد زاویه‌ها را به 18° بازه ده تایی تقسیم کردیم و این 18° بازه را به عنوان بر جسب‌های قسمت شبکه دسته‌بند به شبکه دادیم و زاویه را به جای رگرسیون با دسته‌بندی به شبکه یاد دادیم که در مرحله بعد برای دستیابی به دقیق‌ترین زاویه از 0° تا 180° را به 36° بازه پنج تایی تبدیل کردیم.

در تصاویر زیر نمونه‌هایی از نتایج به دست آمده برای داده‌های آزمون (تست) مشهود است:



شکل ۲.۳: مربع های 30° در 30° بازه برای زاویه با آستانه کوچک



شکل ۳.۳: مربع های 30° در 30° بازه برای زاویه با آستانه متوسط



شکل ۴.۳: مربع های ۱۸ در ۳۰ بازه برای زاویه با آستانه بزرگ



شکل ۵.۳: مربع های ۱۰ در ۳۶ بازه برای زاویه با آستانه کوچک



شکل ۶.۳: مربع های ۱۰ در ۱۰ بازه برای زاویه با آستانه متوسط



شکل ۷.۳: مربع های ۱۰ در ۱۰ بازه برای زاویه با آستانه بزرگ



شکل ۸.۳: مربع های ۸ در ۸ با زاویه ۳۶ باره برای آستانه کوچک



شکل ۹.۳: مربع های ۸ در ۸ با زاویه ۳۶ باره برای آستانه متوسط



شکل ۱۰.۳: مربع های ۸ در ۸ بازه برای زاویه با آستانه بزرگ



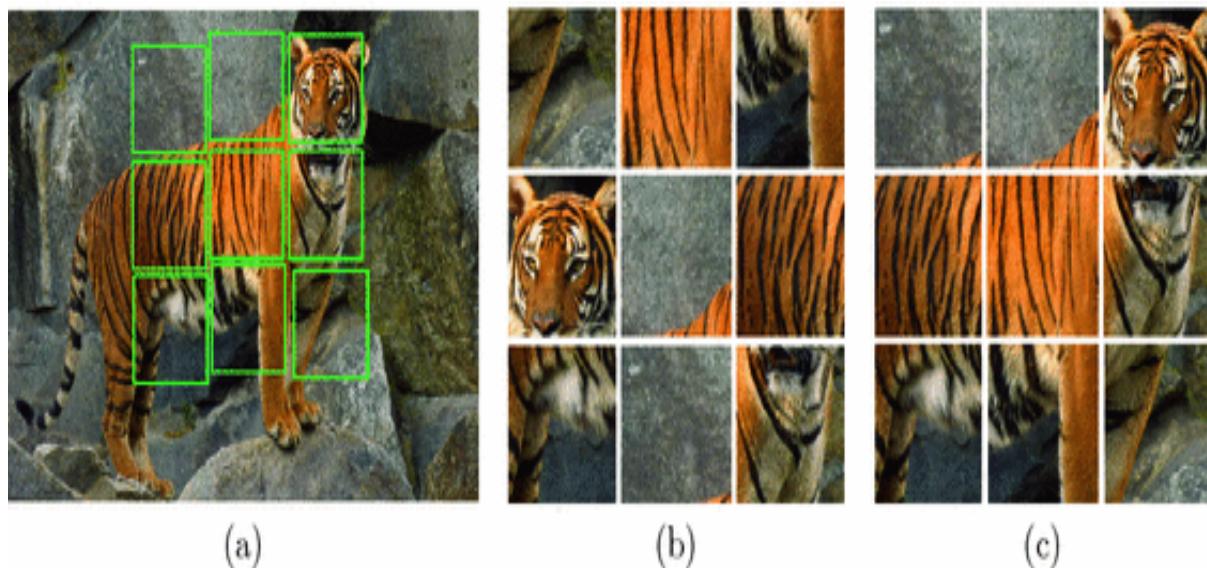
شکل ۱۱.۳: مربع های ۴ در ۴ بازه برای زاویه

شبکه‌هایی که سعی در پیش‌بینی مربع‌هایی با ابعاد کوچک‌تر دارند در نهایت اتلاف بیشتری خواهند داشت و شبکه‌هایی که سعی در پیش‌بینی مربع‌هایی با ابعاد بزرگ‌تر دارند در نهایت اتلاف کمتری خواهند داشت و این نیاز به سبک سنگین کردن دارد تا در تعادل قرار گیرد که بهترین نتیجه را بین شبکه‌ها، شبکه با پیش‌بینی مربع‌های با ابعاد ۳۰ در ۳۰ داشتنند. برای زاویه هم بهترین نتیجه را تقسیم زاویه به ۱۸ بازه داشت زیرا شبکه در دسته‌بندی آن دقیق خیلی بالاتری نسبت به تقسیم

بر ۳۶ بازه داشت و این تفاوت زیاد در دقق به نسبت کوچک شدن بازه ها تاثیر بیشتری داشت بنابراین بهترین نتیجه برای زاویه هم تقسیم زاویه های ۰ تا ۱۸۰ به ۱۸ قسمت با بازه های ده تایی می باشد.

۴ کارهای بیشتر

در قسمت آموزش داده‌های برچسب نخورده می‌توان از روش‌های نیمه‌ناظر یا خودناظر استفاده کرد و از این داده‌ها نیز برای آموزش بهتر شبکه استفاده نمود. ایده‌ای که در کد ما پیاده شده است استفاده از روش خودناظر با تعریف مساله از پیش تعریف شده که هدف اصلی نیست به عنوان بلهای برای آموزش شبکه روی داده‌های برچسب نخورده و استفاده از وزن این شبکه برای آموزش شبکه اصلی روی داده‌های برچسب خورده است. در کد ما از مساله جورچین (تصویر ۱.۴) استفاده شده است که هر تصویر به ۹ قسمت تقسیم شده است و به هر قسمت یک برچسب از ۰ تا ۸ داده شده است و این قسمت‌ها به عنوان داده به شبکه EfficientNet داده شده است تا آموزش ببیند و وزن این شبکه ذخیره شده و به شبکه اصلی به عنوان وزن از پیش آموزش داده شده تحويل داده شده است [۱۴].



شکل ۱.۴: مساله جورچین

در تصاویر زیر نمونه‌هایی از نتایج به دست آمده برای داده‌های آزمون (تست) مشهود است:



شکل ۲.۴: مربع های 30° در 30° بازه برای زاویه با آستانه کوچک



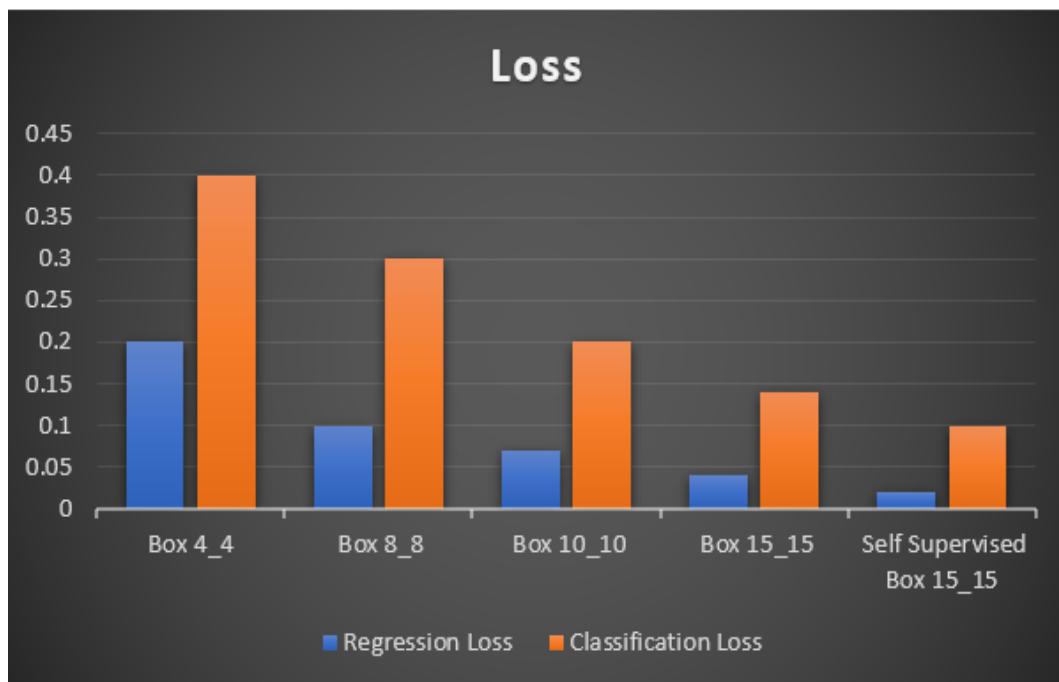
شکل ۳.۴: مربع های 30° در 30° بازه برای زاویه با آستانه متوسط



شکل ۴.۴: مربع های ۳۰ در ۳۰ بازه برای زاویه با آستانه بزرگ

البته حل کردن جورچین برای مساله گل زعفران بسیار سخت است چون قسمت‌های تصاویر گل‌ها خیلی شبیه همند طوری که حل کردن جورچین آن توسط انسان هم بسیار سخت است مساله جورچین مساله چندان مناسبی برای این شبکه حداقل به این شکل تعریف شده به نظر نماید و شبکه آموزش دیده دقت آنچنانی به دست نیاورد ولی با این وجود این وزن آموخته شده تاثیر به سزاگی داشت و به هدف اصلی شبکه کمک کرده و دقت را بالاتر برده است به طوری که بهترین نتیجه ما پس از استفاده از روش خودناظر بوده است و استفاده از آن بسیار مناسب بود.

در تصویر ۵.۴ نتیجه دقت شبکه‌های مختلف مشهود است:



شكل ٥.٤: جدول نتیجه

۵ کارهای آینده

به منظور بهبود آموزش داده‌های برچسب نخورده می‌توان از روش‌های معروف^۵ [CPC، AMDIM و ...] به عنوان مساله افزایی مثل CPC، AMDIM و ... به عنوان مساله از پیش‌تعریف شده استفاده نمود که فرصت امتحان کردن آن‌ها پیش نیامد. مثلاً در روش CPC پس از مقداری تبدیلات تصویر تکه‌های جورچین طوری ساخته می‌شوند که تکه‌های مجاور همپوشانی دارند و مساله به صورت تشخیص جفت‌های مثبت و منفی حل می‌شود. در روش AMDIM دو نوع تبدیل در دو مرحله روی تصویر زده می‌شود و البته یکی وارون دیگریست و در نهایت تشخیص جفت‌های مثبت انجام می‌شود.

- [1] M. Tan, R. Pang, and Q. V. Le, “Efficientdet: Scalable and efficient object detection,” 2020.
- [2] “Efficientdet in keras,” <https://github.com/xuannianz/EfficientDet>, 2020.
- [3] “how to design anchor ratios and scales?,” <https://github.com/xuannianz/EfficientDet/issues/154>, 2020.
- [4] “Image classification via fine-tuning with efficientnet,” https://keras.io/examples/vision/image_classification_efficientnet_fine_tuning/, 2020.
- [5] “A framework for contrastive self-supervised learning and designing a new approach,” <https://towardsdatascience.com/a-framework-for-contrastive-self-supervised-learning-and-designing-a-new-approach-3caab5d29619>, 2020.