به نام خدا

عنوان درس

« پروژه درس مبانی بینایی کامپیوتر »

استاد درس: دکتر حسن ختن لو

نام و نام‌خانوادگی - شماره دانشجویی

محمدامین احمدی 9912358001

یاسین مسیح پور 9912358037

زمستان ۱۴۰۲

فهرست

5 - چکیده

پروژه تشخیص و حذف نویز در تصاویر با استفاده از شبکه‌های عصبی و شبکه‌های عصبی کانولوشنی شامل سه مرحله اصلی است. در مرحله اول، با استفاده از یک مدل یادگیری عمیق، تصاویر دارای نویز از دو دسته "OK" و "Defected" تشخیص داده می‌شوند. در مرحله دوم، تشخیص نوع نویز در تصاویر (نویز gaussian، salt & paper و periodic) انجام می‌شود و تصاویر با استفاده از مدل یادگیری عمیق دیگری که آموزش دیده است، از نویز پاک‌سازی می‌شوند. در مرحله سوم، با استفاده از مدل حاصل از مرحله دوم، تصاویر مرحله اول که دارای نویز هستند ، پاک‌سازی می‌شوند و نتایج آن با نتایج مرحله اول مقایسه می‌شوند.

۲- مقدمه

بینایی کامپیوتر علمی است که کامپیوتر را قادر میسازد تا اطلاعات را از تصویر استخراج کند و این اطلاعات را تفسیر و تحلیل کند. یکی از مهم ترین موارد برای تشخیص درست اطلاعات درون تصویر بالا بودن کیفیت آن است. مهم ترین دلیل پایین آمدن کیفیت تصویر وجود نویز در آن است و در این پروژه قرار است به حل این مشکل بپردازیم و با مقایسه دقت مدل توسعه یافته خود برای تصاویر دارای نویز و تصاویر بدون نویز به این جواب برسیم که آیا با حذف نویز می توان به دقت بهتری برای مدل خود رسید؟

۳- روش و الگوریتم پیشنهادی

برای انجام این پروژه، ابتدا دو مرحله اصلی مشخص شده است. در مرحله اول، هدف تشخیص تصاویر عادی و عیب‌دار از یک دیسک است. در مرحله دوم، باید نویز موجود در تصاویر را تشخیص داده و از آن حذف شود. پس از این دو مرحله، نتایج به دست آمده از مرحله سوم با نتایج مرحله اول مقایسه خواهد شد.

در هر مرحله ابتدا با استفاده از برنامه split.py دیتاست مربوطه را به داده های test , train , validation تقسیم می کنیم.

در فاز اول، از شبکه‌های عصبی پیچشی (CNN) برای تشخیص تصاویر عادی و عیب‌دار استفاده شده است. این شبکه‌ها با استخراج ویژگی‌های تصاویر و آموزش بر روی داده‌های آموزشی، قادر به تمایز بین تصاویر عادی و عیب‌دار می‌شوند.

در این فاز پس از مشخص کردن اندازه ورودی ها مدل vgg16 را لود کرده و همه لایه های آن را فریز می کنیم.

VGG16یک معماری شبکه عصبی کانولوشنی است ، این معماری شامل 16 لایه است، که شامل 13 لایه کانولوشنی و 3 لایه کاملا متصل می‌شود.

یکی از مزایای بزرگ VGG16 این است که پیش‌آموزش داده شده است روی مجموعه داده ImageNetکه شامل میلیون‌ها تصویر با برچسب دسته‌بندی مختلف است. این امر باعث می‌شود که VGG16 بتواند ویژگی‌های عمومی از تصاویر را یاد بگیرد و بتواند به خوبی در وظایف دسته‌بندی تصاویر به کار گرفته شود.
همچنین از تابع imageDataGenerator که برای تولید داده های تصویری برای آموزش شبکه های عصبی کانولوشنی است استفاده کرده ایم.
ImageDataGenerator می‌تواند داده‌های ورودی را تغییر شکل دهد، از جمله چرخش، تغییر مقیاس، شیفت دادن افقی و عمودی، تغییر روشنایی، اعمال تکانه (Augmentation) و سایر تغییراتی که به تصویر داده شده است. این تغییرات باعث می‌شود که مدل بتواند الگوهای مختلفی را یاد بگیرد و به طور کلی مقاومت بیشتری در برابر داده‌های جدید داشته باشد.

پس از لود کردن داده های test , train , validation در صورت نامتوازن بودن کلاس ها ، باید به آن ها وزن مناسبی داده شود.
در نهایت از مدل ساخته شده خروجی گرفته و از آن در برنامه test_model.py استفاده می کنیم.

در فاز دوم، از معماری‌های مختلفی مانند شبکه‌های عصبی پیچشی و روش‌های پردازش تصویر برای تشخیص و حذف نویز از تصاویر استفاده می‌شود. این مدل‌ها با استفاده از داده‌های مرتبط با نوع نویز، آموزش داده می‌شوند و سپس بر روی تصاویر نویزدار اعمال می‌شوند.

در این فاز از مدل آموزش دیده resnet50 برای تشخیص نوع نویز تصاویر کمک می گیریم.
ResNet50یک معماری از شبکه‌های عصبی عمیق کانولوشنی است که در زمینه بینایی کامپیوتری مورد استفاده قرار می‌گیرد. این معماری به طور خاص برای دسته‌بندی تصاویر، تشخیص اشیاء، تقسیم تصویر و وظایف مشابه دیگر به کار می‌رود.
پس از مراحل پیش پردازش و موازنه کردن کلاس ها از مدل ساخته شده در این فاز نیز خروجی گرفته و در test_model2.py استفاده می کنیم.
پس از تشخیص نویز توسط مدلی که ساخته ایم ، درصورتی که نویز از نوع salt & paper باشد ، از فیلتر median blur استفاده می کنیم.
در صورتی که نویز از نوع گوسین باشد ، از GaussianBlur استفاده می کنیم.
و در صورتی که از نوع periodic باشد از تابعی به نام remove_periodic_noise_color استفاده می کنیم.
تابع remove_periodic_noise_color:
این تابع برای حذف نویزهای دوره‌ای در تصاویر رنگی استفاده می‌شود. در این تابع:

1. تصویر ورودی به کانال‌های رنگی تقسیم می‌شود.
2. برای هر کانال رنگی، تبدیل فوریه انجام می‌شود.
3. یک فیلتر گوسی در فضای فرکانس ایجاد می‌شود و با تبدیل فوریه تصویر ضرب می‌شود تا نویزهای دوره‌ای را حذف کند.
4. تبدیل فوریه بازگردانده می‌شود به فضای زمان.
5. مقادیر پیکسل‌ها نرمالایز شده و به مقادیر بین 0 تا 255 تبدیل می‌شوند.
6. کانال‌های رنگی فیلتر شده ترکیب می‌شوند و تصویر نهایی حاصل برگردانده می‌شود.

در نهایت عکس های بدون نویز را در آدرس داده شده ذخیره می کنیم.

برای مقایسه از معیار هایی مثل lpips , psnr , ssin استفاده شده است.
PSNRیا Signal-to-Noise Ratio (نسبت سیگنال به نویز) پارامتری است که در پردازش تصویر برای ارزیابی کیفیت تصاویر استفاده می‌شود. این معیار به ما می‌گوید که چقدر تصویر با کیفیتی که از تصویر اصلی دریافت کرده‌ایم متفاوت است. معنای اصطلاحی "سیگنال" در اینجا تصویر اصلی است و "نویز" تفاوت بین تصویر اصلی و تصویر با کیفیت پایین است.

LPIPS یک معیار ارزیابی کیفیت تصاویر است که اختصاراً به معنای "Perceptual Similarity" است. این معیار به دنبال ارزیابی تشابه درکی بین دو تصویر است، به جای ارزیابی معیارهای سنتی مانند MSEیا PSNR که بر اساس تفاوت‌های پیکسلی عمل می‌کنند.

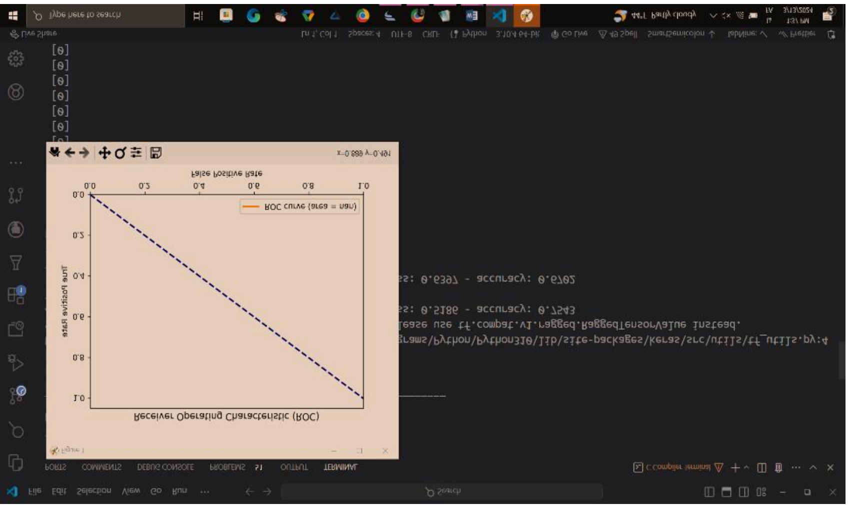
SIM یا Structural Similarity Index Measure یک معیار برای اندازه‌گیری شباهت ساختاری بین دو تصویر است. این معیار به طور خاص برای ارزیابی میزان تطابق بین یک تصویر اصلی و یک تصویر پیش‌بینی شده یا تصویری که توسط یک الگوریتم پردازش تولید شده است، استفاده می‌شود.

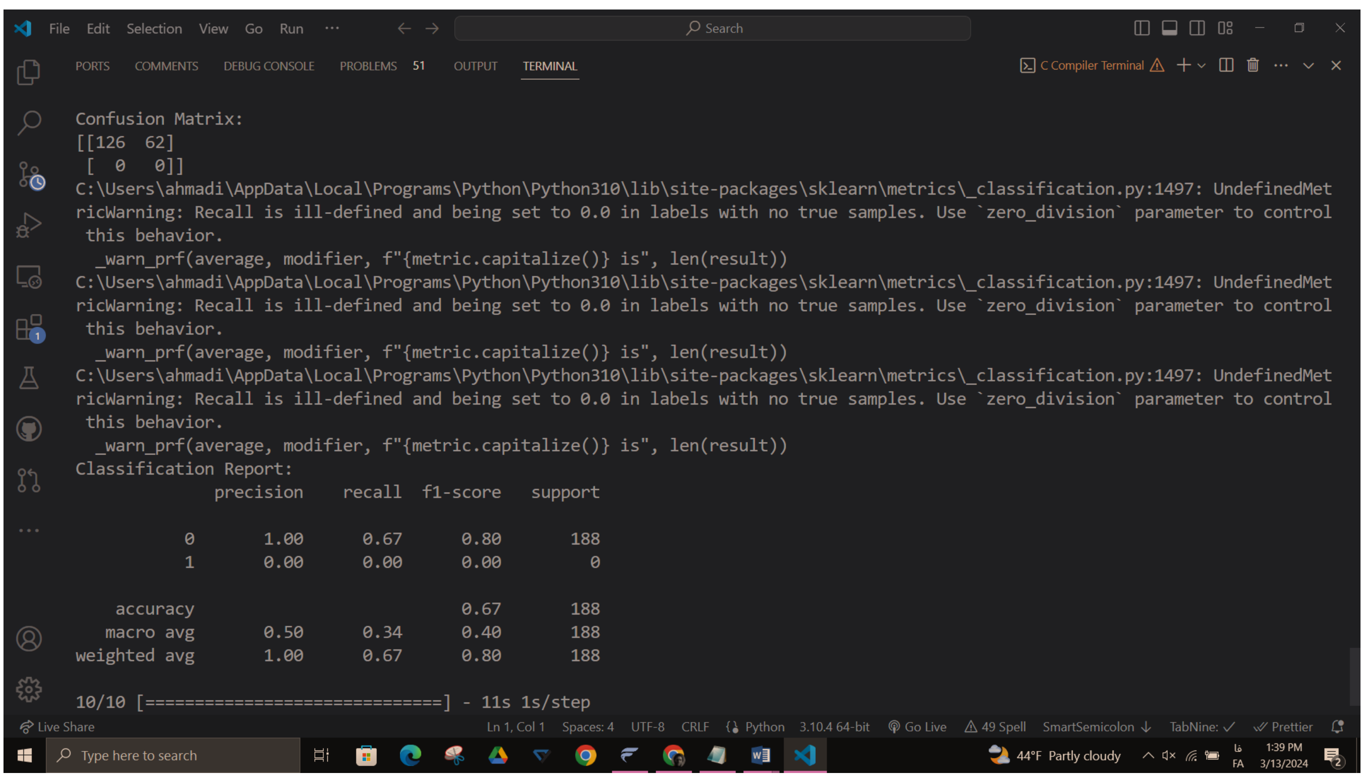
و در نهایت با کمک مدل ساخته شده توسط test_model2.py عکس های دیتاست اول را denoise کرده و با کمک مدل ساخته شده توسط test_model.py تشخیص می دهیم کدام یک از عکس های جدید عضو کلاس ok و کدام defected است.

پس از پایان مراحل تشخیص و حذف نویز، نتایج به دست آمده با استفاده از معیارهای ارزیابی مناسب ارزیابی می‌شوند. سپس نتایج این مرحله با نتایج مرحله اول مقایسه شده و ارزیابی نهایی انجام می‌شود.

4 -آزمایش و نتایج

عکس ها و متریک های مربوط به فاز یک :





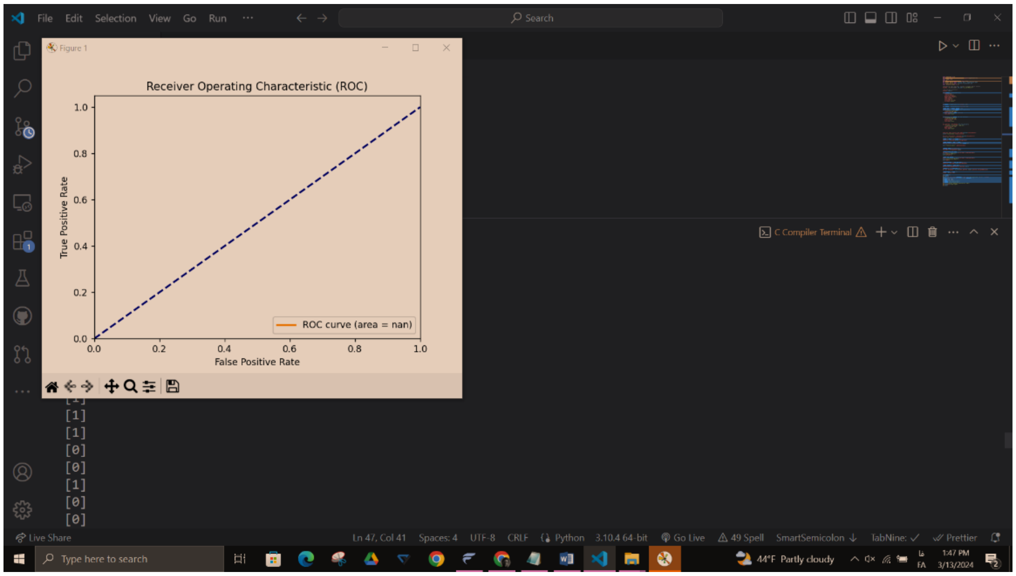
Confusion Matrix:

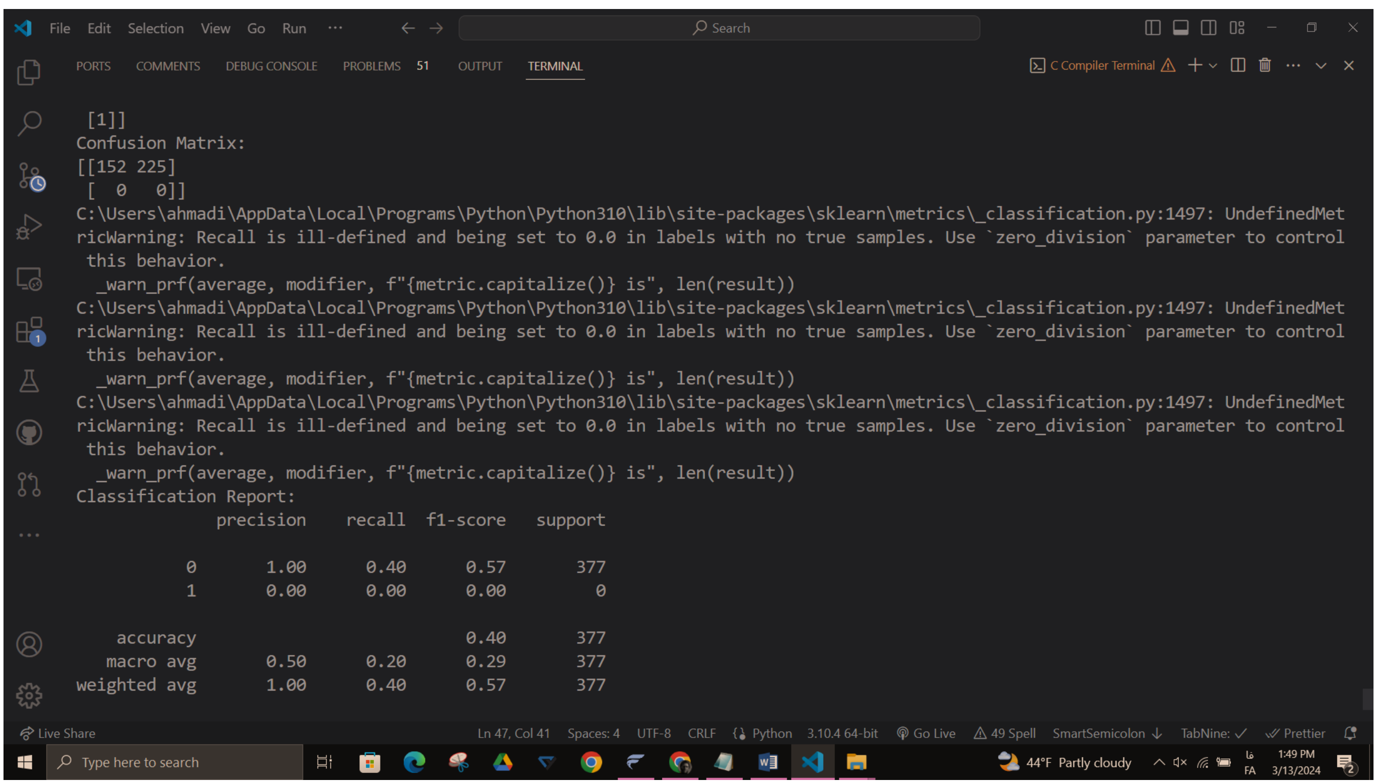
```
[[126  62]
 [  0   0]]
```

Classification Report:

	precision	recall	f1-score	support
0	1.00	0.67	0.80	188
1	0.00	0.00	0.00	0
accuracy			0.67	188
macro avg	0.50	0.34	0.40	188
weighted avg	1.00	0.67	0.80	188

عکس ها و متریک های مربوط به فاز3 :





Confusion Matrix:
[[152 225]
[0 0]]

Classification Report:

	precision	recall	f1-score	support
0	1.00	0.40	0.57	377
1	0.00	0.00	0.00	0
accuracy			0.40	377
macro avg	0.50	0.20	0.29	377
weighted avg	1.00	0.40	0.57	377

با توجه به متریک های پس از حذف نویز تصاویر دقت تصویر بال تر می رود.

جست‌وجو پارامترهای بهینه

پارامتر	مقدار بهینه
Epoch	50
Batch	20
Learning rate	0.001

۵- نتیجه‌گیری

در این پروژه، ما به بررسی و پیاده‌سازی سه فاز اصلی در حوزه بینایی ماشین پرداختیم. در فاز اول، موفق به طراحی و آموزش یک مدل برای تشخیص دیسک‌های نرمال و معیوب با استفاده از شبکه‌های عصبی کانولوشنال شدیم. این مدل قادر بود تا با دقت نه چندان بالا دیسک‌های نرمال و معیوب را تشخیص دهد.

در فاز دوم، ما به تشخیص و پاکسازی نویزهای مختلف در تصاویر دیسک‌ها پرداختیم. با استفاده از مدل‌های پیش‌آموزش‌شده مانند ResNet50 و الگوریتم‌های پردازش تصویر مختلف، توانستیم نویزهای گوسی، نویز سالت و پریودیک را تشخیص داده و از تصاویر پاکسازی کنیم.

در فاز سوم، ما از مدلی که در فاز دوم آموزش دیده بودیم، برای پاکسازی تصاویر دیسک‌های معیوب در دیتاست جدید استفاده کردیم و سپس نتایج را با مدلی که در فاز اول آموزش دیده بودیم مقایسه کردیم.

نتایج نشان دادند که مدل‌های طراحی شده موفق به تشخیص و پاکسازی نویزها و تشخیص دیسک‌های معیوب و نرمال بودند. این پروژه نه‌تنها ارزش تحقیقاتی بالایی دارد، بلکه می‌تواند در صنایع مختلفی از جمله پزشکی و تولید دیسک‌های فیزیکی مورد استفاده قرار گیرد.

پایان