

بخش ۱

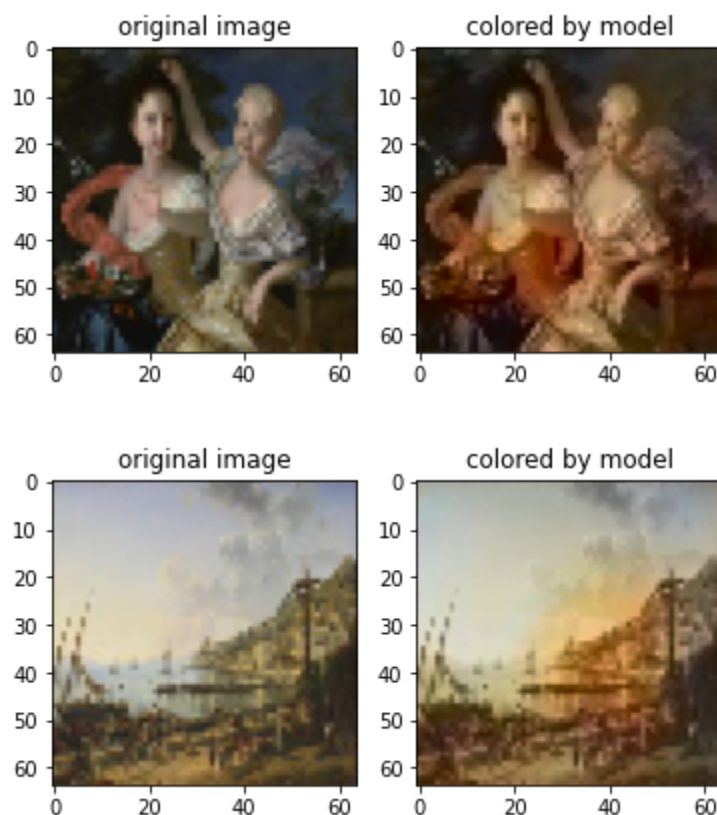
برای پیش پردازش داده از دیتاست pytorch استفاده می‌کنیم. با استفاده از این قابلیت و تعریف یک کلاس دیتاست جدید، سایز عکس‌ها را یکسان می‌کنیم و عکس target هر نمونه داده را، سیاه و سفید همان عکس قرار می‌دهیم. یکی از نکات مهم در پروژه image colorization، فرمت ذخیره‌سازی عکس‌هاست. برای مدل پایه از فرمت RGB استفاده شد، اما در این تسک معمولاً از فرمت Lab استفاده می‌شود. در این فرمت L نشان‌دهنده lightness است یعنی همان عکس سیاه و سفید است و a, b ترکیب رنگی هر پیکسل را نشان می‌دهد. مزیت این فرمت نسبت به RGB برای رنگ‌آمیزی این است که مدل کافی است دو کانال را پیش‌بینی کند (در حالی که در RGB باید سه کانال را پیش‌بینی کند).

یکی دیگر از عوامل بسیار تاثیرگذار در نتیجه رنگ‌آمیزی، تابع خطا مورد استفاده است. ساده‌ترین انتخاب، MSE است. اما مشکل این تابع این است که رنگ‌های تولید شده توسط مدل معمولاً کمرنگ و نزدیک به رنگ قهوه‌ای خواهد بود. برای رفع این مشکل در مقاله colorful colorization، فضای a, b را به تعدادی پارتیشن تفکیک می‌کند (در مقاله 313 پارتیشن انتخاب می‌شود) و مدل از بین این پارتیشن‌های رنگی یکی را انتخاب می‌کند. یکی دیگر از ایده‌های مطرح شده در این مقاله، color rebalancing است. در این تابع خطا به هر رنگ با توجه به میزان تکرار آن رنگ در مجموعه داده، وزنی تعلق می‌گیرد تا مدل بتواند رنگ‌های جالب‌تری را انتخاب کند. در این مقاله بهبود دیگری نیز برای تابع خطا معرفی می‌شود که مدل را مجاب می‌کند به پیکسل‌های مجاور با lightness یکسان رنگ‌های مشابهی نسبت دهد. (متأسفانه موفق به پیاده‌سازی این مقاله نشدم)

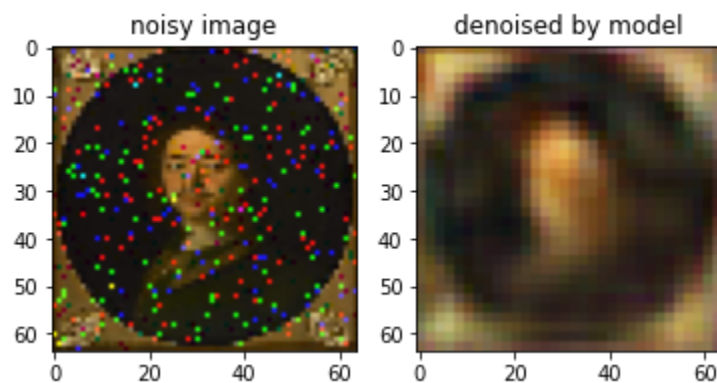
با مقایسه لرنینگ ریت‌های مختلف (همچنین روش reduce on plateau برای تغییر دینامیک لرنینگ ریت)، می‌توان دید در این مسئله لرنینگ ریت در حدود 0.001 مناسب‌تر است. لرنینگ ریت بزرگ‌تر باعث می‌شود مدل همگرا نشود و لرنینگ ریت کمتر یادگیری مدل را بسیار کند می‌کند.

همچنین با مقایسه بین Adam, AdamW, SGD به نظر می‌رسد Adam اوپتیمایزر بهتری در این مسئله می‌باشد.

با مقایسه مدل‌ها با معماری‌های مختلف و فرمت عکس‌های مختلف مشاهده شد با استفاده از مدل resnet پرتیرین شده برای انکودر و فرمت Lab بهترین نتیجه حاصل می‌شود.



می‌توان هنگام ساخت دیتاست، به عکس‌ها نویز اضافه کرد و مدل را طوری آموزش داد تا عکس نویزدار را به عنوان ورودی بگیرد و سعی کند عکس اصلی بدون نویز را بسازد. نتایج یک مدل تست شده روی این دیتاست را در تصویر زیر می‌بینیم.



بخش ۲

در این بخش با استفاده از مقاله مدیوم مطرح شده در تمرین، مدل VAE را ساخته و روی مجموعه داده میوه‌ها ترین می‌کنیم. حال برای تولید میوه جدید از ایده درونیابی استفاده می‌کنیم. دو عکس از مجموعه داده را وارد انکودر می‌کنیم، حال دو بردار برای هر عکس داریم، که با درونیابی بین این دو بردار مثلاً ۱۰ بردار دیگر بدست می‌آوریم، سپس آن ده بردار را به دیکودر می‌دهیم تا تصویر متناظرشان را تولید کند. می‌توان دید این ده تصویر نحوه تبدیل میوه اول به دوم را می‌سازد که می‌توان میوه‌های جدیدی در این میان مشاهده کرد. نمونه‌ای از این کار در تصویر زیر قابل مشاهده است.

