

پروژه پایانی جبر خطی

محیا شهشهانی - ۸۱۰۱۹۹۵۹۸

بخش اول:

۱. تنسورها ابعاد بیشتری از ماتریس دارند. در حالی که ماتریس‌ها دو بعدی هستند، تنسورها می‌توانند ابعاد بیشتری داشته باشند، از جمله بعدهای سوم، چهارم، و غیره. این به این معناست که تنسورها می‌توانند اطلاعات را در فضای چند بعدی نگهداری کنند، که برای پردازش داده‌های پیچیده مانند تصاویر و ویدیوها بسیار مفید هستند.

۲. هر دو الگوریتم‌های استخراج ویژگی هستند که در تجزیه و تحلیل داده‌ها و کاهش ابعاد مورد استفاده قرار می‌گیرند. اما در Robust PCA، تلاش می‌شود تا الگوریتم در برابر داده‌های پرت مقاوم باشد. در مقابل، PCA به صورت حساس به داده‌های پرت است و ممکن است تاثیر مخربی از آن‌ها بپذیرد. در Robust PCA، یک تابع هزینه یا (non-convex penalty function) به مسئله اضافه می‌شود تا تاثیر داده‌های پرت کاهش یابد. یکی از روش‌های معمول برای این منظور افزودن یک عبارت بهینه‌سازی به مسئله است:

$$\min_{L,S} \|L\|_* + \lambda \|S\|_1$$

که در آن L ماتریس اصلی است و $\|L\|_*$ نمایانگر Nuclear Norm ماتریس L است و S ماتریس Sparse است.

۳. SVD (Singular Value Decomposition):

SVD یک تکنیک مهم در تجزیه و تحلیل ماتریس‌ها است که ماتریس را به حاصلضرب سه ماتریس زیر تجزیه می‌کند:

$$A = U \Sigma V^T$$

Σ ماتریس قطری و شامل ویژگی‌های مهم است.

Randomized SVD

Randomized SVD یک رویکرد تقریبی به SVD است که از تصادف در مراحل مختلف برای کاهش پیچیدگی محاسباتی استفاده می‌کند. با این روش، ابعاد ماتریس به صورت تصادفی کاهش می‌یابند تا بتوان به سرعت SVD را محاسبه کرد. این روش مناسب برای ماتریس‌های بزرگ است و می‌تواند به تسریع فرآیند تحلیل ماتریس‌های بزرگ کمک کند.

Truncated SVD

Truncated SVD نیز یک نسخه از SVD است که با حفظ تعداد معینی از مقادیر منفرد (singular values) به جای استفاده از تمام آن‌ها، از پیچیدگی محاسباتی کمتری برخوردار است. این کاهش ابعاد می‌تواند منجر به تقریب بهتر و کاهش حجم داده‌ها شود.

Higher-order SVD

Higher-order SVD یا Tucker Decomposition یک تعمیم از SVD است. در این حالت، به جای ماتریس یک تنسور با ابعاد بیشتر داریم. برخلاف SVD که بر روی دو بعد اعمال می‌شود، در Higher-order SVD تجزیه و تحلیل از بُعدهای بیشتری انجام می‌شود. این نوع تجزیه و تحلیل ممکن است در مواردی که داده‌ها ساختارهای پیچیده‌تری دارند، مفید باشد.

تفاوت اصلی این روش‌ها با SVD در استفاده از رویکردهای مختلف برای کاهش ابعاد و تقریب زودتر نتایج است. هر کدام از این روش‌ها مزایا و محدودیت‌های خود را دارند و بسته به موارد مختلف می‌توانند مفید باشند.

۴. Nuclear Norm ، یک معیار است که در ماتریس‌ها مورد استفاده قرار می‌گیرد و برابر است با مجموع مقدار منفردهاب ماتریس. برای یک ماتریس X با ابعاد $m \times n$ Nuclear Norm به صورت زیر تعریف می‌شود:

$$\|X\|_* = \sum_{i=1}^{\min(m,n)} \sigma_i$$

که در آن σ_i مقدار منفردهای ماتریس X هستند. و مرتب شده‌اند. و این نرم به عنوان یک معیار کمی از میزان "متفاوت بودن" ماتریس از ماتریس دیگر می‌شود.

فریم شماره ۵۹۸ را به انتخاب مشاهده می‌کنیم:



In [10]: `frame.shape`

Out[10]: (360, 640, 3)

دو مقدار اول نشان دهنده رزولوشن، و سومی چنل‌های تصویر را نمایندگی میکند.

به دلیل رنگی بودن، RGB.

Resolution of the Frame: 640x360

سپس به ویدیوی سیاه سفید شده، نویز اضافه کردیم. تعداد فریم‌های ویدیوی رنگی اولیه ۹۰۱ و برای این ویدیو، ۳۰۳ بود.



خروجی به صورت رو به رو میباشد:

باتوجه به کد، مقادیر لاندا و میو و تعداد تکرارها روی خروجی تاثیر گذارند. الگوریتم به طریقی انجام میشود تا تکرارها تا زمان رسیدن به مقدار خطای قابل قبولی تکرار شوند. خروجی قابل قبول، ماتریس low rank، شامل بک گراند، و Sparse شامل مقادیر پراکندگی‌های موجود می‌باشد. خیر با این روش نمیتوانیم با یک فریم تصویر پس زمینه را به دست آوریم زیرا برای تجزیه تنسور به Low Rank و Sparse، نیاز به چندین فریم است تا اطلاعات خطی و پس‌زمینه با دقت تجزیه شوند. با داشتن تنها یک فریم، اطلاعات خطی و پس‌زمینه به خوبی تفکیک نمی‌شوند و ممکن است به ویژه در مواجهه با نویز، نتایج دقیقی حاصل نشود.

در پایین سمت راست، فریمی از Sparse و سمت چپ از Low rank می‌بینیم.



با تغییر مقادیر گفته شده، به دلیل حساسیت الگوریتم، ممکن است خروجی‌هایی به شکل زیر داشته باشیم:



Reconstruction error خطا یا اختلاف بین داده اصلی و داده بازسازی شده است. این خطا معمولاً به عنوان یک معیار اندازه‌گیری کیفیت بازسازی مدل استفاده می‌شود. هرچه این خطا کمتر باشد، نشان‌دهنده این است که مدل بهترین بازسازی را انجام داده است.

err: 6.602390708094621e-06

ویدیوهای استخراج شده در فایل موجود است.

بخش دوم:

۵. همانطور که در مقدمه گفته شد، Watermarking یک فرآیند است که با افزودن اطلاعات نهان به یک تصویر یا ویدئو، اطلاعات اصلی را حفظ می‌کند و به آن امکان تشخیص اصالت و مالکیت را می‌دهد. ساختار کلی Watermarking شامل:

Embedding:

در این مرحله، اطلاعات watermark به تصویر اصلی اضافه می‌شود. این اطلاعات ممکن است به صورت متن، تصویر، یا سیگنال‌های دیگر باشند. Embedding معمولاً به گونه‌هایی صورت می‌گیرد که تغییرات در تصویر اصلی به نحوی باشد که برای چشم انسان غیر قابل مشاهده باشد.

Watermark استخراج:

در این مرحله، کاربر یا سیستمی که حق دسترسی به اطلاعات watermark دارد، اقدام به تجزیه و تحلیل تصویر کرده و اطلاعات نهان را استخراج می‌کند. این عملیات معمولاً نیاز به کلید یا الگوریتم مشخصی دارد که اطلاعات را از تصویر جدا کند.

۶. الگوریتم واترمارکینگ با استفاده از تجزیه مقادیر منفرد، برای نهان سازی و آشکار سازی به صورت زیر می باشد:

Watermark Casting

$$\begin{aligned} A &\Rightarrow USV^H \\ S + aW &\Rightarrow U_W S_W V_W^H \\ A_W &\Leftarrow U S_W V^H \end{aligned}$$

A: Original Image

A_W : Watermarked image

Watermark Detection

$$\begin{aligned} A_W^* &\Rightarrow U^* S_W^* V^{*H} \\ D^* &\Leftarrow U_W S_W^* V_W^H \\ W^* &\Leftarrow \frac{1}{a}(D^* - S) \end{aligned}$$

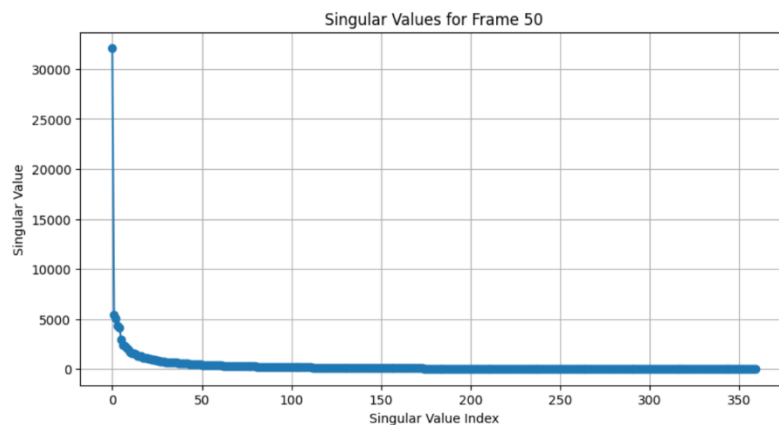
A_W : Watermarked Image

A_W^* : Distorted Image

۷. نحوه Blur کردن با استفاده از Kernel:

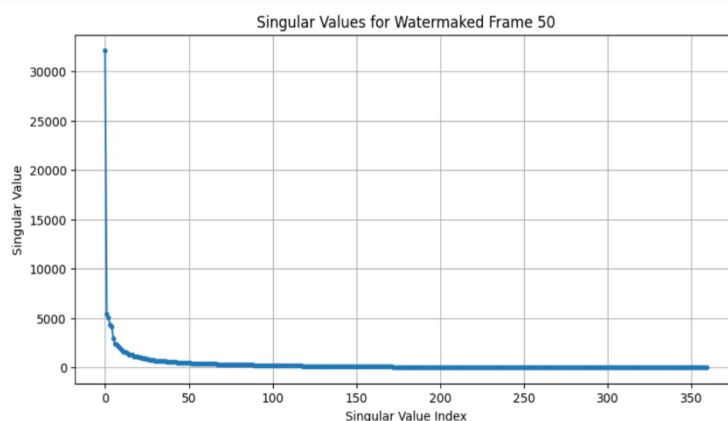
تعیین Kernel: در این مرحله، یک Kernel یا فیلتر انتخاب می شود. این Kernel معمولاً یک ماتریس کوچک با اعداد وزن دارد. برای مثال، فیلتر Gaussian Blur از یک ماتریس گوسی برای اعمال اثر پراکندگی استفاده می کند. اعمال Kernel: برای هر پیکسل در تصویر، Kernel بر روی محل متناظر با آن پیکسل قرار گرفته و عملیات مشخصی (مثل جمع وزن دار) بر روی مقادیر پیکسل های همجوار انجام می دهد. این عمل به ازای هر پیکسل در تصویر تکرار می شود. نتیجه Blur: مقدار جدید بدست آمده برای هر پیکسل تصویر به عنوان نتیجه Blur قرار می گیرد. این مقدار معمولاً میانگین وزن دار مقادیر پیکسل های همجوار است. تاثیر سایز Kernel بر روی Blur: استفاده از Kernel با سایز بزرگتر باعث می شود تاثیر Blur قوی تر و گسترده تر شود. این باعث کاهش جزئیات در تصویر می شود اما ممکن است باعث از دست رفتن تفصیل بیش از حد شود. Kernel با سایز کوچکتر باعث Blur کمتر و حفظ جزئیات بیشتر در تصویر می شود، اما ممکن است منجر به افزایش نویز و جزئیات مخرب شود.

اگر ما پس از آنکه ویدیو را خواندیم، نمودار مقادیر ویژه را برای یک فریم رسم کنیم، خواهیم داشت: برای فریم شماره ۵۰:

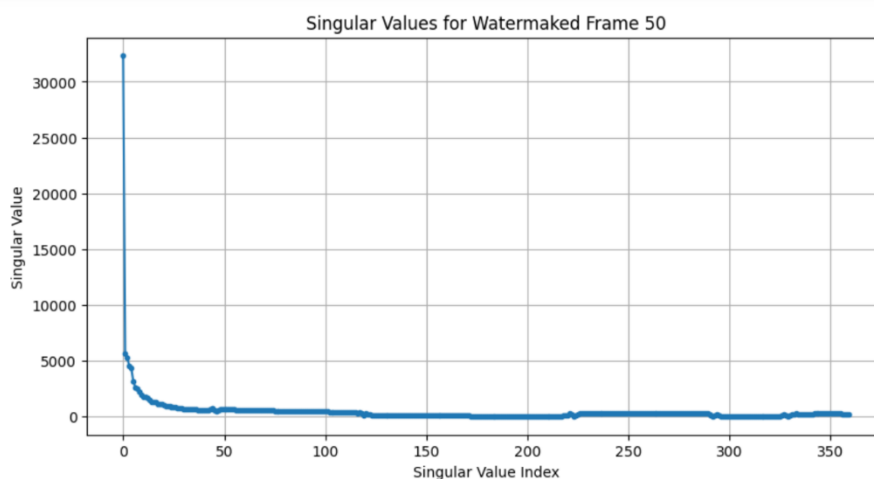


طبق الگوریتم میدانیم که نهان سازی از طریق اضافه شدن ماتریس واترمارک (عموما ضربی از آن) به ماتریس مقادیر منفرد صورت می گیرد.

پس از انجام عملیات های برابر کردن اندازه و ...، واترمارک را روی فریم ها اعمال می کنیم. به طور مثال، در اینجا نمودار فریم ۵۰ پس از اعمال واترمارک به صورت زیر خواهد بود:



با توجه به اینکه یک مقدار ویژه خیلی بزرگ داریم، تغییرات خیلی محسوس نیست. مخصوصاً که ضریب اعمالی را ۰.۱ قرار دادیم. اگر ضریب اعمالی را تغییر بدیم، تغییرات بهتر قابل مشاهده خواهند بود:



حال اگر دو تصویر را بعد و قبل از اعمال واتر مارک با هم مقایسه کنیم:



میبینیم که همانطور که انتظار داشتیم، تفاوتی در ظاهر تصاویر وجود ندارد. اطلاعات به طریقی اضافه شده که قابل مشاهده با چشم انسان نباشد.