

Image Processing

Phase2

کورش حسن زاده – متین مژگانی – شروین خادمی کلانتری

اعمال فیلتر های مکانی و فرکانسی

1. Average Filter (Mean Filter)

فیلتر میانگین یک فیلتر هموارسازی مکانی است که برای کاهش نویز و smooth کردن تصویر استفاده می شود.

نحوه عملکرد:

در این روش، مقدار هر پیکسل با میانگین مقادیر پیکسل های همسایه در یک پنجره (Kernel) جایگزین می شود.
در واقع همه پیکسل های داخل کرنل وزن یکسان دارند.

کاربرد:

- کاهش نویز های یکنواخت
- صاف کردن تصویر
- کاهش جزئیات ریز

نکته:

باعث محو شدن لبه ها می شود.

2. Gaussian Filter

فیلتر گاوی یک فیلتر هموارسازی است که بر اساس توزیع نرمال (Gaussian Distribution) وزن دهی می کند.

نحوه عملکرد:

برخلاف فیلتر میانگین، پیکسل های نزدیک به مرکز کرنل وزن بیشتری دارند و پیکسل های دورتر وزن کمتری می گیرند.
وزن ها از تابع گاوی به دست می آیند.

کاربرد:

- کاهش نویز با حفظ بهتر لبه ها نسبت به Average
- پیش پردازش قبل از تشخیص لبه

مزیت نسبت به میانگین:
هموارسازی طبیعی‌تر و کمتر مخرب برای ساختار تصویر.

3. Median Filter

فیلتر میانه یک فیلتر غیرخطی برای حذف نویز، مخصوصاً نویز Salt & Pepper است.

نحوه عملکرد:
در هر پنجره، مقادیر پیکسل‌ها مرتب می‌شوند و مقدار میانه جایگزین پیکسل مرکزی می‌شود.

کاربرد:

- حذف نویز Salt & Pepper Noise
- حفظ بهتر لبه‌ها نسبت به فیلترهای خطی

ویژگی مهم:
چون غیرخطی است، ساختار لبه‌ها را بهتر حفظ می‌کند.

4. Sharpening Filter

فیلتر تیزکننده برای افزایش وضوح و برجسته کردن لبه‌ها استفاده می‌شود.

نحوه عملکرد:
با استفاده از یک کرنل تقویت‌کننده، اختلاف شدت بین پیکسل مرکزی و همسایه‌ها افزایش داده می‌شود.
در نتیجه تغییرات شدت (لبه‌ها) تقویت می‌شوند.

کاربرد:

- افزایش وضوح تصویر
- برجسته کردن جزئیات
- بهبود نمایش لبه‌ها

5. Sobel Filter

فیلتر سوبل یک روش تشخیص لبه در حوزه مکانی است.

نحوه عملکرد:

گرادیان شدت تصویر در دو جهت افقی (x) و عمودی (y) محاسبه می‌شود.
سپس اندازه گرادیان کلی از ترکیب این دو به دست می‌آید.

مفهوم اصلی:

لبه‌ها نقاطی هستند که تغییر شدت در آن‌ها زیاد است.

کاربرد:

- تشخیص لبه
- استخراج ویژگی
- پیش‌پردازش در بینایی ماشین

6. Low-Pass Filter (در حوزه فرکانس)

فیلتر پایین‌گذر (Low Pass) فرکانس‌های پایین را عبور داده و فرکانس‌های بالا را حذف می‌کند.

نحوه عملکرد:

پس از تبدیل فوریه (FFT)، یک ماسک گاوی روی طیف فرکانسی اعمال می‌شود که فرکانس‌های مرکزی (پایین) را حفظ می‌کند.

نتیجه:

- تصویر نرم و هموار می‌شود
- جزئیات ریز و نویز کاهش می‌یابد

7. High-Pass Filter (در حوزه فرکانس)

فیلتر بالاگذر فرکانس‌های بالا را عبور داده و فرکانس‌های پایین را حذف می‌کند.

نحوه عملکرد:

در حوزه فرکانس، مکمل فیلتر پایین‌گذر اعمال می‌شود تا فقط فرکانس‌های بالا باقی بمانند.

نتیجه:

- لبه‌ها و جزئیات تقویت می‌شوند
- تغییرات سریع شدت بر جسته می‌شوند

نتیجه فیلترهای مختلف:



افزودن نویز و حذف نویز از تصاویر

در این بخش، ابتدا انواع نویز به تصاویر افزوده می‌شود و سپس با استفاده از فیلترهای مناسب، فرآیند حذف نویز (Denoising) انجام می‌گیرد.

1. Salt & Pepper Noise

نویز نمک و فلفل نوعی نویز ضربه‌ای (Impulse Noise) است که به صورت نقاط کاملاً سفید (Salt) و کاملاً سیاه (Pepper) در تصویر ظاهر می‌شود.

نحوه ایجاد:
به صورت تصادفی تعدادی از پیکسل‌ها با مقدار 0 (سیاه) یا 255 (سفید) جایگزین می‌شوند.
پارامترهای اصلی شامل:

- میزان نویز (amount)

- نسبت نویز سفید به سیاه (salt_vs_pepper)

ویژگی‌ها:

- به صورت پراکنده در تصویر ظاهر می‌شود
- معمولاً ناشی از خطا در سنسور یا انتقال داده است

2. Gaussian Noise

نویز گوسین نوعی نویز آماری است که از توزیع نرمال (Gaussian Distribution) پیروی می‌کند.

نحوه ایجاد:

به هر پیکسل یک مقدار تصادفی که از توزیع نرمال با میانگین (mean) و انحراف معیار (std) مشخص نمونه‌داری شده است اضافه می‌شود.

ویژگی‌ها:

- در کل تصویر پخش می‌شود
- شدت نویز با افزایش انحراف معیار بیشتر می‌شود
- مدل رایج نویز در سیستم‌های تصویربرداری

روش‌های حذف نویز

3. Median Filter برای حذف Salt & Pepper

فیلتر میانه یک فیلتر غیرخطی مؤثر برای حذف نویز ضربه‌ای است.

نحوه عملکرد:

در هر پنجره همسایگی:

1. مقادیر پیکسل‌ها مرتب می‌شوند.
2. مقدار میانه جایگزین پیکسل مرکزی می‌شود.

چرا مناسب Salt & Pepper است؟

چون پیکسل‌های بسیار بزرگ (255) یا بسیار کوچک (0) که به صورت غیرعادی وارد شده‌اند، در مرتب‌سازی حذف شده و مقدار میانه انتخاب می‌شود.

مزیت:

- حفظ بهتر لبه‌ها
- حذف مؤثر نویز ضربه‌ای

4. Bilateral Filter (برای حذف Gaussian Noise)

فیلتر دو طرفه (Bilateral Filter) یک فیلتر غیر خطی است که هم‌زمان از اطلاعات مکانی و شدت روشنایی استفاده می‌کند.

نحوه عملکرد:
وزن دهنده بر اساس دو معیار انجام می‌شود:

1. فاصله مکانی بین پیکسل‌ها (Spatial Distance)
2. اختلاف شدت روشنایی (Intensity Difference)

در نتیجه، فقط پیکسل‌هایی که هم نزدیک هستند و هم شدت مشابه دارند در میانگین‌گیری مشارکت می‌کنند.

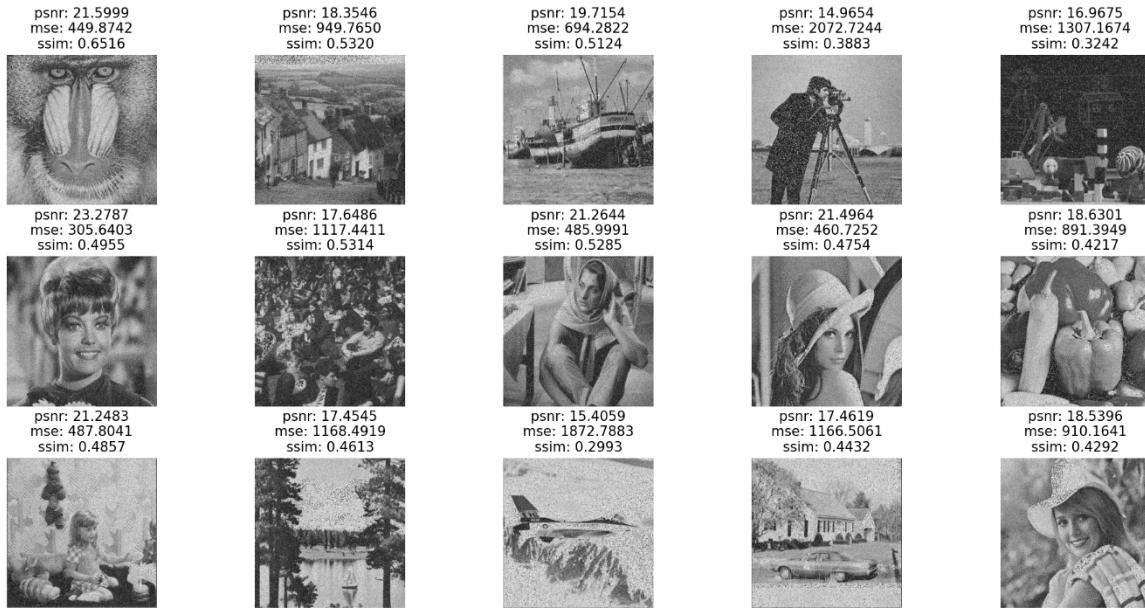
مزیت اصلی:

- کاهش نویز
- حفظ لبه‌ها
- جلوگیری از محوشدن ساختار تصویر

نتیجه اعمال فیلتر median بر روی نویز salt & peper



نتیجه اعمال فیلتر bilateral بر روی نویز gaussian:



معیاری کتی برای سنجش میزان شباهت تصویر بازسازی شده به تصویر اصلی است که بر اساس خطای میانگین مربعی (MSE) محاسبه می‌شود. این معیار معمولاً بر حسب دسیبل (dB) بیان می‌شود و هرچه مقدار آن بیشتر باشد، کیفیت بازسازی بهتر و میزان نویز کمتر است. با این حال، PSNR این تفاوتها اختلاف عددی پیکسل‌ها را اندازه‌گیری می‌کند و لزوماً با ادراک بصری انسان کاملاً منطبق نیست.

معیاری ادراکی‌تر است که شباهت دو تصویر را بر اساس سه مؤلفه روشنایی (Luminance)، کنترast (Contrast) و ساختار (Structure) ارزیابی می‌کند. مقدار SSIM معمولاً بین 0 و 1 قرار می‌گیرد و هرچه به 1 نزدیک‌تر باشد، شباهت ساختاری تصویر بازسازی شده به تصویر اصلی بیشتر است. برخلاف PSNR، این معیار بهتر می‌تواند کیفیت بصری و حفظ ساختار تصویر را منعکس کند.

Edge Detection Methods

در این بخش چهار الگوریتم مختلف تشخیص لبه پیاده‌سازی و مقایسه شده‌اند: Canny، Marr-Hildreth، Kirsch و Prewitt

1. Prewitt Edge Detection

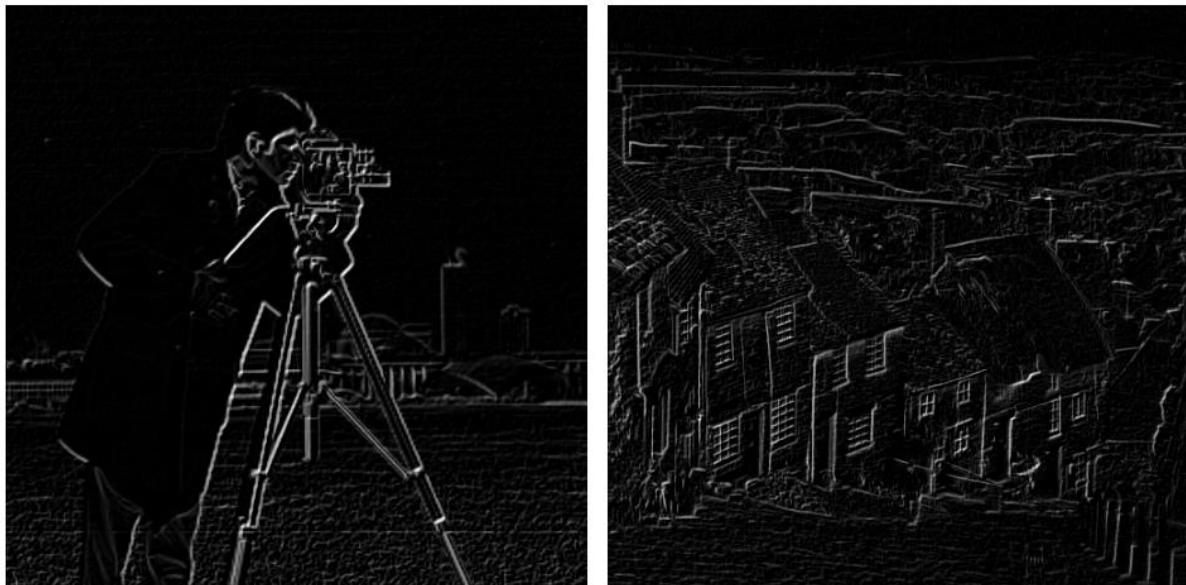
یک روش مبتنی بر گرادیان برای تشخیص لبه‌ها است که تغییرات شدت روشنایی را در دو جهت افقی و عمودی اندازه‌گیری می‌کند. Prewitt

مراحل کلی:

1. تبدیل تصویر به سطح خاکستری
2. کانولوشن با کرنل‌های Prewitt_x و Prewitt_y
3. محاسبه اندازه گرادیان
4. اعمال آستانه برای استخراج لبه‌ها

تحلیل عملکرد:

این روش لبه‌ها را در تمام جهات تشخیص می‌دهد و عملکرد مناسبی دارد. با این حال، چون مبتنی بر مشتق است، نویز را تقویت می‌کند. در تصاویر بدون نویز شدید، نتیجه قابل قبول و دقیق است.



2. Kirsch Edge Detection

الگوریتم Kirsch از ۸ ماسک کانولوشنی در جهات مختلف استفاده می‌کند و قوی‌ترین پاسخ گرادیانی را برای هر پیکسل انتخاب می‌کند.

مراحل کلی:

1. اعمال ۸ ماسک در جهات مختلف
2. محاسبه پاسخ گرادیان برای هر جهت
3. انتخاب بیشینه پاسخ به عنوان مقدار نهایی لبه

تحلیل عملکرد:

Kirsch نسبت به Prewitt لبه‌های قوی‌تر و واضح‌تری تولید می‌کند. کیفیت لبه‌ها بهتر است و نسبت به Prewitt کمتر نویز را تقویت می‌کند، اگرچه همچنان در حضور نویز شدید حساس است.



3. Marr-Hildreth (LoG) Edge Detection

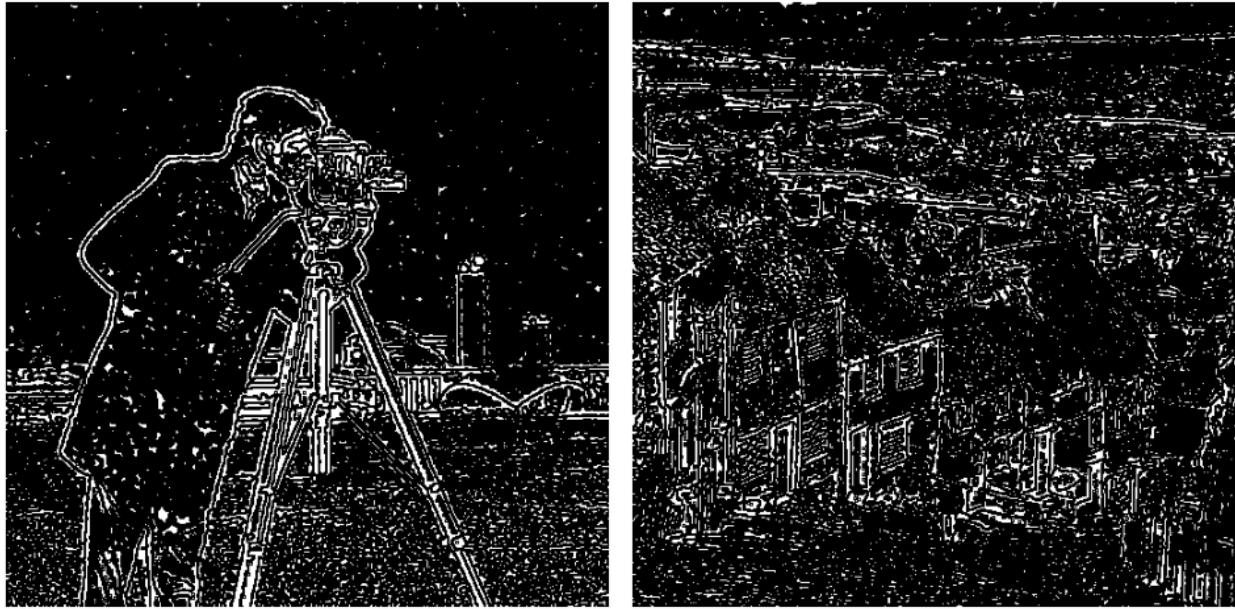
این روش ترکیبی از هموارسازی گاوی و عملگر لاپلاسین است و به نام Laplacian of Gaussian (LoG) نیز شناخته می‌شود.

مراحل کلی:

1. هموارسازی تصویر با فیلتر گاوی
2. اعمال لاپلاسین
3. تشخیص Zero-Crossing
4. آستانه‌گذاری برای تعیین لبه‌ها

تحلیل عملکرد:

این الگوریتم نسبت به دو روش قبلی لبه‌های ضخیم‌تری تولید می‌کند و نویز را بیشتر تقویت می‌کند. پیچیدگی محاسباتی بالاتری دارد و زمان اجرای آن بیشتر است. با این حال، نسبت به چرخش مقلوم است و برای تحلیل چندمقیاسی- (Multi-Scale Analysis) مناسب می‌باشد.



4. Canny Edge Detection

Canny یک روش چند مرحله‌ای و پیشرفته برای استخراج لبه‌ها است که هدف آن بیشینه‌سازی دقت و کاهش پاسخ‌های کاذب است.

مراحل کلی:

- .1 هموارسازی گلوسوی
- .2 محاسبه گرادیان با استفاده از Sobel
- .3 Non-Maximum Suppression
- .4 Double Thresholding و اتصال لبه‌ها

تحلیل عملکرد:

در مقایسه با سه روش دیگر، Canny بهترین عملکرد را دارد. لبه‌ها نازک‌تر، دقیق‌تر و منظم‌تر هستند و نویز کمتر تقویت می‌شود. با این حال، در حضور نویز بسیار شدید عملکرد آن نیز کاهش می‌یابد.

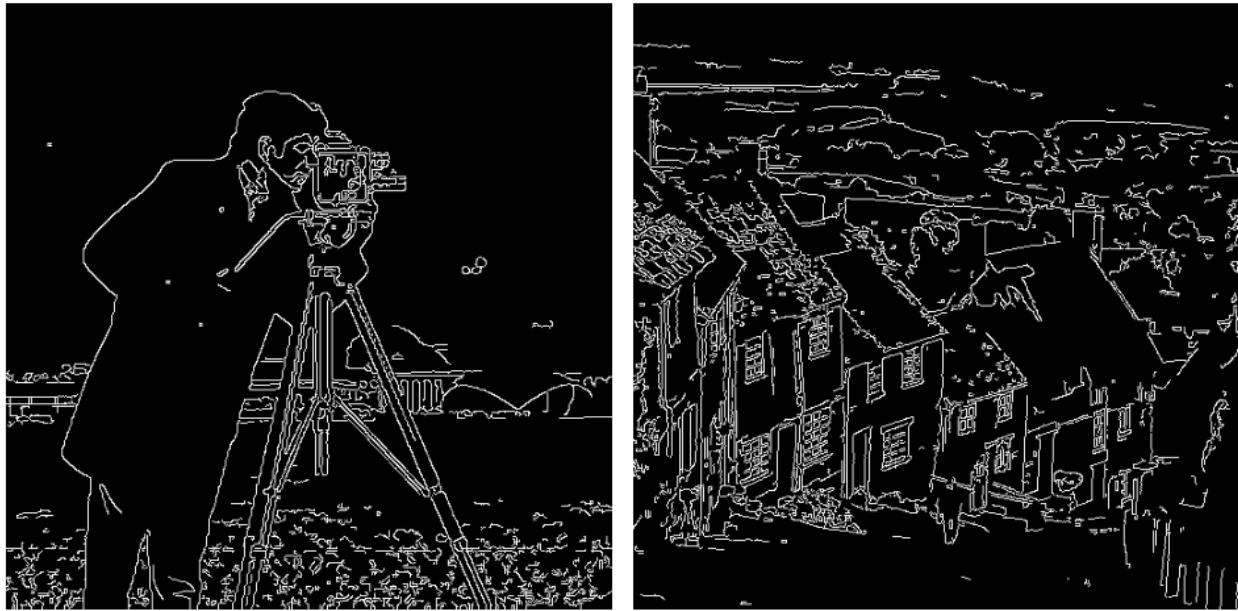


Image Segmentation using K-Means Clustering

فرآیند تقسیم تصویر به نواحی یا گروههایی از پیکسل‌ها است که ویژگی‌های مشابهی دارند. در این بخش، از روش خوشبندی K-Means برای بخش‌بندی رنگی تصویر استفاده شده است.

Clustering-Based Segmentation

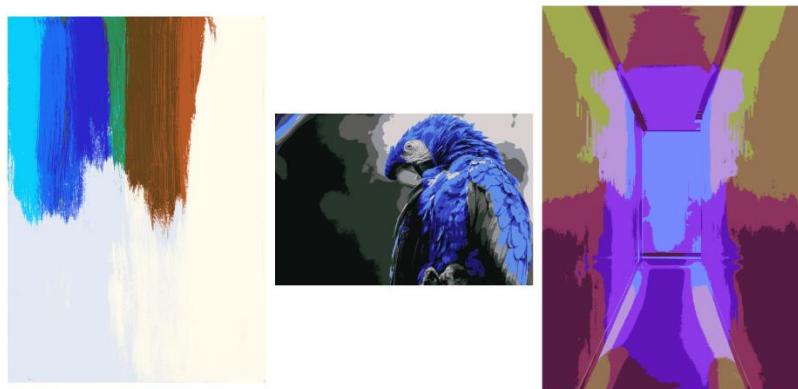
در این روش، پیکسل‌ها بر اساس شباهت ویژگی‌ها) در اینجا مقدار رنگ (RGB) به چند خوش‌بندی تقسیم می‌شوند. فرض اصلی این است که پیکسل‌های متعلق به یک خوش، از نظر رنگی به یکدیگر نزدیک هستند.

مراحل الگوریتم K-Means

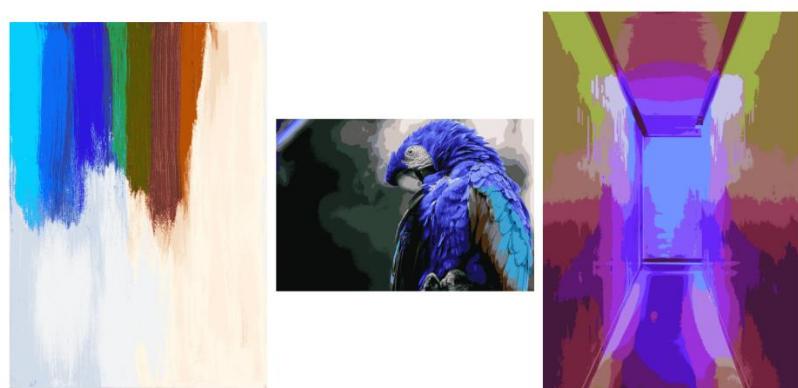
1. انتخاب تصادفی k مرکز اولیه(Cluster Centers)
2. تخصیص هر پیکسل به نزدیکترین مرکز بر اساس فاصله اقلیدسی
3. محاسبه مجدد مرکز هر خوش به عنوان میانگین نقاط آن
4. تکرار مراحل تخصیص و بهروزرسانی مرکز
5. توقف زمانی که مرکز تغییر نکند یا به حداقل تعداد تکرار برسیم

در نهایت، هر پیکسل با رنگ مرکز خوشه خود جایگزین می‌شود و تصویر به k رنگ غالب محدود می‌شود.

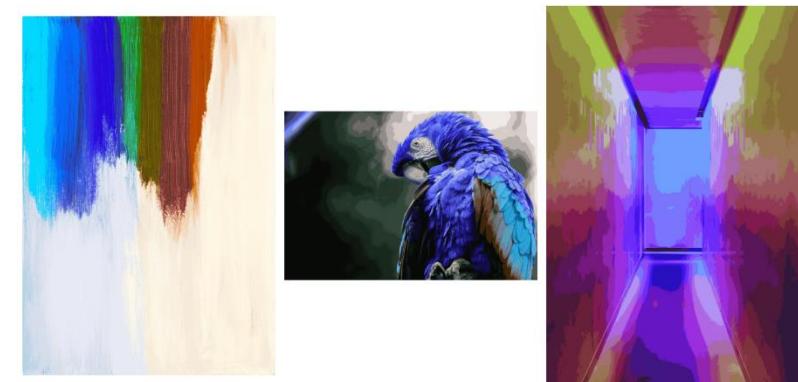
خروجی با 8 کلاستر:



خروجی با 16 کلاستر:



خروجی با 32 کلاستر:



Thresholding Methods (Binarization)

آستانه‌گذاری (Thresholding) یکی از روش‌های پایه در پردازش تصویر برای تبدیل تصویر خاکستری به تصویر دودویی (Binary) است. هدف آن جداسازی نواحی مهم (مثل شیء/متن) از پس زمینه با تبدیل پیکسل‌ها به دو مقدار 0 و 255 می‌باشد.

1. Simple (Global) Thresholding

در آستانه‌گذاری ساده، یک مقدار ثابت به عنوان آستانه برای کل تصویر انتخاب می‌شود.

نحوه عملکرد:
اگر مقدار شدت پیکسل از آستانه بزرگتر باشد، خروجی سفید (255) و در غیر این صورت سیاه (0) می‌شود.
در حالت معکوس (Inverse) این تصمیم بر عکس اعمال می‌شود.

کاربرد و محدودیت:
سریع و ساده است، اما در تصاویر با نور غیر یکنواخت (سایه/روشنایی متغیر) معمولاً نتیجه ضعیفتر می‌دهد.

2. Adaptive Mean Thresholding

آستانه‌گذاری تطبیقی به جای یک آستانه ثابت، برای هر ناحیه کوچک از تصویر یک آستانه محلی محاسبه می‌کند.

نحوه عملکرد:
برای هر پیکسل، میانگین شدت روشنایی در یک پنجره همسایگی (block size) محاسبه شده و سپس یک مقدار ثابت (C) از آن کم می‌شود. این مقدار به عنوان آستانه همان ناحیه استفاده می‌شود.

مزیت:
در شرایطی که نور تصویر در بخش‌های مختلف مقاومت است، عملکرد بهتری نسبت به روش سراسری دارد.

3. Adaptive Gaussian Thresholding

این روش مشابه Adaptive Mean است، با این تفاوت که میانگین محلی به صورت وزن‌دار (Gaussian Weighted) محاسبه می‌شود.

نحوه عملکرد:

پیکسل های نزدیک به مرکز پنجه وزن بیشتری دارند و پیکسل های دورتر وزن کمتری میگیرند. سپس مقدار C از میانگین وزن دار کم می شود و آستانه محلی تولید میگردد.

مزیت:

به دلیل وزن دهی گاوی، معمولاً نتیجه دقیق تر و پایدار تر (خصوصاً در حضور نویز و تغییرات نرم روشنایی) ارائه می دهد.

4. Otsu's Thresholding

روش اتسو یک روش خودکار برای تعیین آستانه بهینه در آستانه گذاری سراسری است و نیازی به انتخاب دستی آستانه ندارد.

نحوه عملکرد:

با تحلیل هیستوگرام تصویر، آستانه ای انتخاب می شود که واریانس درون کلاسی را کمینه و جدایی بین کلاس پس زمینه و پیش زمینه را بیشینه کند.

کاربرد:

زمانی مناسب است که تصویر دارای دو فله واضح در هیستوگرام باشد (یعنی پیش زمینه و پس زمینه تقسیک پذیر باشند).

نتایج:



Histogram of Oriented Gradients (HOG)

یک توصیفگر ویژگی (Feature Descriptor) HOG (Histogram of Oriented Gradients) پرکاربرد در بینایی مашین است که برای استخراج اطلاعات ساختاری تصویر، بهویژه در تشخیص اشیاء، استفاده می‌شود. ایده اصلی HOG شمارش جهت‌های گرادیان در نواحی محلی تصویر است.

مراحل الگوریتم HOG

Gradient Calculation .1

گرادیان تصویر در جهات افقی و عمودی محاسبه می‌شود) معمولاً با فیلترهایی مانند Sobel) سپس مقدار (Orientation) و جهت (Magnitude) گرادیان برای هر پیکسل استخراج می‌شود.

Cell Division .2

تصویر به سلوال‌های کوچک (مثلاً 8×8 پیکسل) تقسیم می‌شود.

Histogram Formation .3

برای هر سلوال، یک هیستوگرام از جهت‌های گرادیان ساخته می‌شود. هر bin نمایانگر یک بازه زاویه‌ای (مثلاً بین ۰ تا ۱۸۰ درجه) است و مقدار آن برابر مجموع شدت گرادیان‌ها در آن جهت می‌باشد.

Block Normalization .4

چند سلوال مجاور در قالب یک بلوک گرویندی شده و هیستوگرام‌ها نرمال‌سازی می‌شوند. این مرحله باعث کاهش تأثیر تغییرات روشنایی و کنتراست می‌شود.

Descriptor Formation .5

تمامی هیستوگرام‌های نرمال‌شده به هم متصل می‌شوند و بردار نهایی ویژگی (Feature Vector) تشکیل می‌شود.

پارامترهای اصلی HOG

orientations:

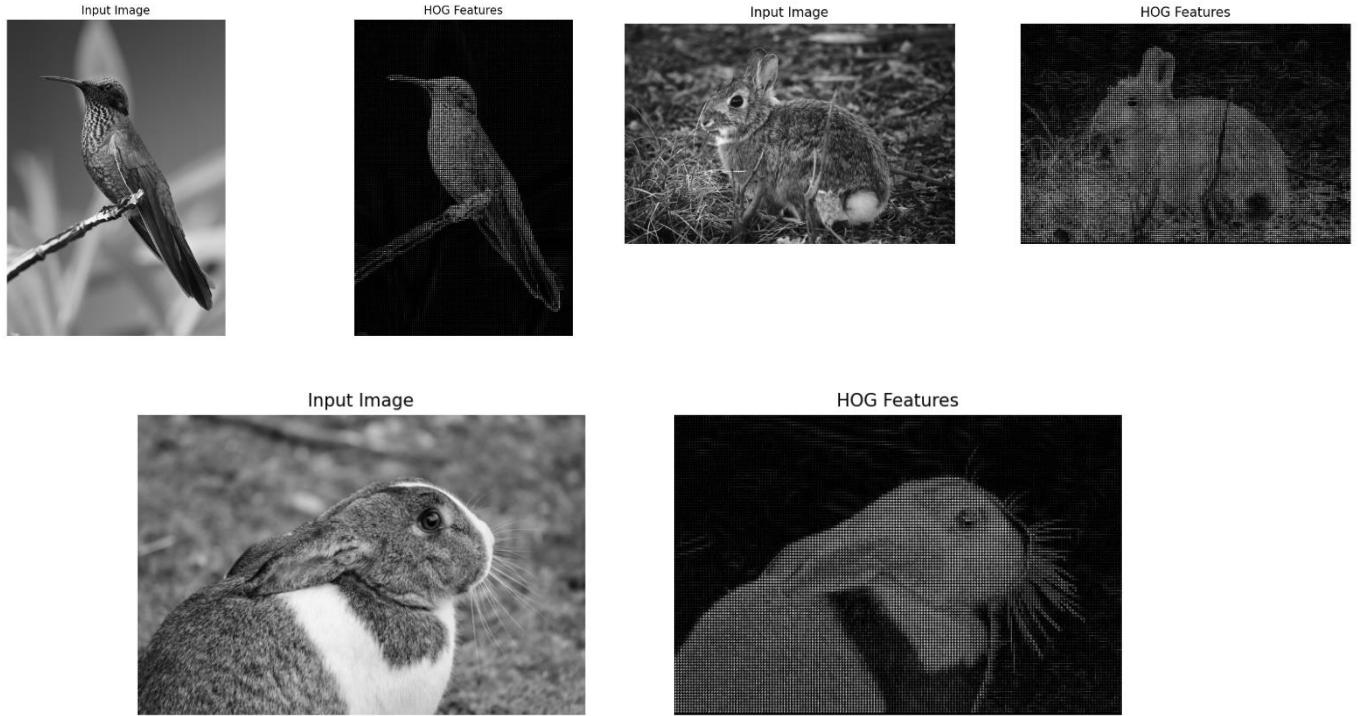
تعداد بازه‌های زاویه‌ای هیستوگرام را تعیین می‌کند. افزایش این مقدار باعث نمایش دقیق‌تر جهت‌ها می‌شود، اما بعد ویژگی‌ها و هزینه محاسباتی را افزایش می‌دهد.

pixels_per_cell:

اندازه هر سلوال را مشخص می‌کند.
سلول‌های کوچک‌تر → استخراج جزئیات دقیق‌تر
سلول‌های بزرگ‌تر → نمایش الگوهای کلی‌تر با کاهش جزئیات

cells_per_block:

تعداد سلوال‌هایی که در یک بلوک برای نرمال‌سازی استفاده می‌شوند.
بلوک‌های بزرگ‌تر اطلاعات فضایی بیشتری را در بر می‌گیرند، اما ممکن است جزئیات محلی کاهش یابد.



Denoising with Convolutional AutoEncoder (Fashion-MNIST)

در این بخش برای حذف نویز از تصاویر از یک **AutoEncoder** کانولوشنی استفاده شده است. دیتاست مورد استفاده تصاویر 28×28 تک کاناله (Fashion-MNIST) بوده و هدف مدل این است که با دریافت تصویر نویزی، نسخه‌ی تمیز آن را بازسازی کند. در مرحله آموزش، به تصاویر ورودی نویز گاوی با ضریب noise_factor=0.2 (Clean) اضافه شده و سپس مدل طوری آموزش می‌بیند که خروجی آن به تصویر اصلی (بدون نویز) نزدیک شود.تابع هزینه مورد استفاده MSE است که اختلاف پیکسل به پیکسل بین تصویر بازسازی شده و تصویر اصلی را کمینه می‌کند.



ساختار مدل (Model Architecture)

مدل از دو بخش اصلی تشکیل شده است:

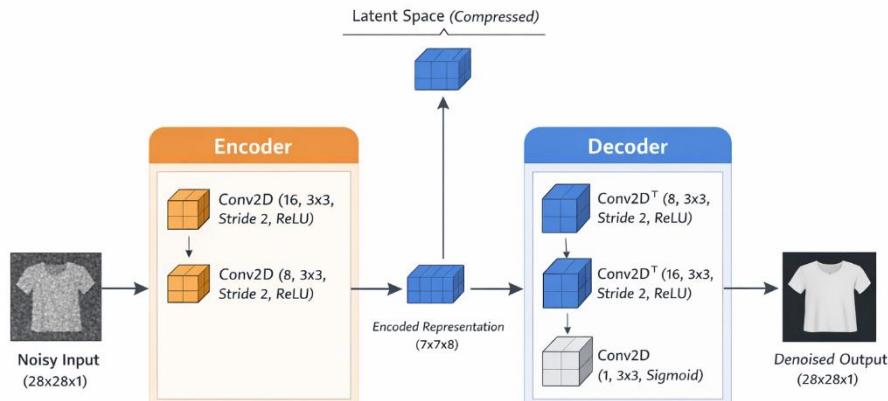
Encoder:

با دو لایه Conv2D و stride=2 ابعاد تصویر را کاهش می‌دهد و یک نمایش فشرده (Latent Representation) از تصویر تولید می‌کند. این بخش ویژگی‌های مهم تصویر را استخراج کرده و اطلاعات زائد (از جمله نویز) را تا حدی حذف می‌کند.

Decoder:

با استفاده از Conv2DTranspose ابعاد را دوباره افزایش می‌دهد و تصویر را بازسازی می‌کند. در لایه آخر از فعال‌ساز استفاده شده تا خروجی در بازه [0, 1] باقی بماند و با نرمال‌سازی ورودی‌ها سازگار باشد.

Denoising Convolutional AutoEncoder



روند آموزش

- داده‌ها به شکل (1, 28, 28) آماده‌سازی شده‌اند.
- به داده‌های ورودی نویز گاوی اضافه و سپس مقدار پیکسل‌ها به بازه [0, 1] محدود شده است. (Clipping).
- مدل با Adam و Mean Squared Error (MSE) آموزش داده شده است.
- برای کنترل آموزش از دو مکانیزم استفاده شده است:
 - ModelCheckpoint برای ذخیره مدل در هر epoch به همراه مقدار val_loss
 - EarlyStopping برای جلوگیری از overfitting (در صورت عدم بهبود val_loss طی چند epoch)

ارزیابی مدل(Evaluation)

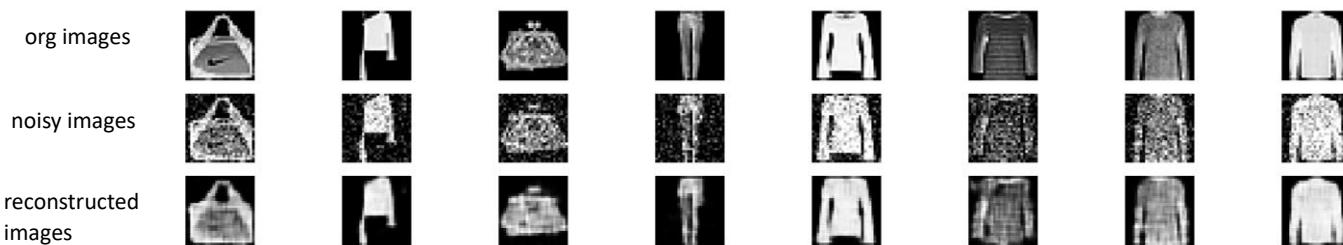
برای ارزیابی کیفیت بازسازی علاوه بر خطاهای عددی، معیارهای ادراکی نیز محاسبه شده‌اند:

- **MSE / MAE:** خطای پیکسلی بین تصویر اصلی و بازسازی شده
- **PSNR:** هرچه بزرگتر باشد کیفیت بازسازی بالاتر است (نویز کمتر)
- **SSIM:** میزان شباهت ساختاری بین تصاویر؛ هرچه به 1 نزدیکتر باشد حفظ ساختار بهتر است

نتایج روی دیتابی تست: Fashion-MNIST

- MSE: **0.007488**
- MAE: **0.047937**
- PSNR(mean): **21.6279**
- SSIM(mean): **0.7586**

این نتایج نشان می‌دهند مدل توانسته نویز را به شکل قابل قبولی کاهش دهد و ساختار کلی لباس‌ها را تا حد خوبی حفظ کند.



روی یک تصویر خارج از دیتابی آموزش Inference

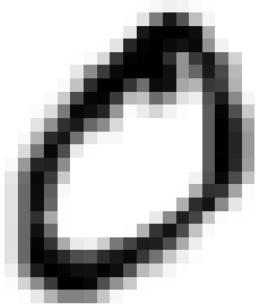
در اسکریپت Inference، یک تصویر از دیتابی **MNIST** (عدد دستنویس) به مدل داده شده و پس از افزودن نویز با همان $noise_factor=0.2$ بازسازی انجام شده است. نتیجه از نظر بصری مناسب بوده اما از نظر معیارهای عددی اختلاف بیشتری با حالت تست Fashion-MNIST دارد (به دلیل تغییر توزیع داده‌ها و Domain Shift)

نتایج Inference روی تصویر: MNIST

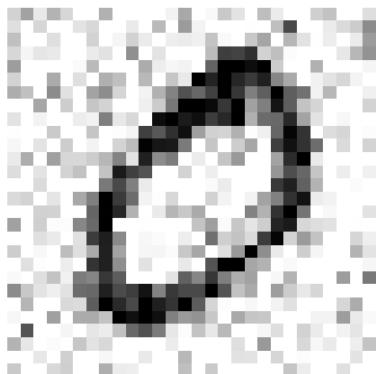
- MSE: **0.034839**
- MAE: **0.133155**
- PSNR: **14.5794**
- SSIM: **0.8240**

با وجود افزایش MSE و کاهش PSNR (اختلاف پیکسلی بیشتر)، مقدار SSIM بالاتر نشان می‌دهد که مدل همچنان توانسته ساختار کلی رقم را نسبتاً خوب حفظ کند؛ یعنی بازسازی از نظر ادراکی قابل قبول بوده است، حتی اگر از نظر پیکسلی دقیقاً مشابه تصویر اصلی نباشد.

Clean



Noisy



Denoised

