

۱ فاز اول: استخراج ویژگی‌ها

اهداف اصلی

هدف از این فاز، پردازش تصاویر طبیعت و استخراج ویژگی‌های کلیدی برای خوشه‌بندی بود. ویژگی‌های استخراج شده باید بتوانند تفاوت بین انواع مختلف تصاویر طبیعی را به خوبی نشان دهند.

چالش‌ها و راهکارها

یکسان‌سازی اندازه تصاویر:

تصاویر ورودی با اندازه‌های مختلف وارد می‌شدند. برای حل این مشکل، همه تصاویر به اندازه استاندارد 128×128 پیکسل تغییر اندازه داده شدند. این اندازه به دلیلی انتخاب شد که هم جزئیات کافی حفظ شود و هم محاسبات بهینه باشد.

انتخاب ویژگی‌های معنادار:

ویژگی‌های انتخاب شده باید تفاوت بین کلاس‌ها را به خوبی نشان می‌دادند. برای این منظور از ترکیب چند نوع ویژگی استفاده شد.

محاسبه کارآمد ویژگی‌ها:

برای پردازش ۳۶۰۰ تصویر، از توابع بهینه‌شده کتابخانه‌های OpenCV و scikit-image استفاده شد تا محاسبات سریعتر انجام شود.

ویژگی‌های استخراج شده

ویژگی‌های رنگی:

- میانگین کانال‌های B، G، R

- انحراف معیار کانال‌های رنگی

این ویژگی‌ها تفاوت رنگ بین مناطق مختلف مثل دریا و جنگل را نشان می‌دهند.

ویژگی‌های آماری:

- میانگین سطح خاکستری
- واریانس سطح خاکستری
- این موارد روشیایی و کنتراست کلی تصویر را اندازه می‌گیرند.

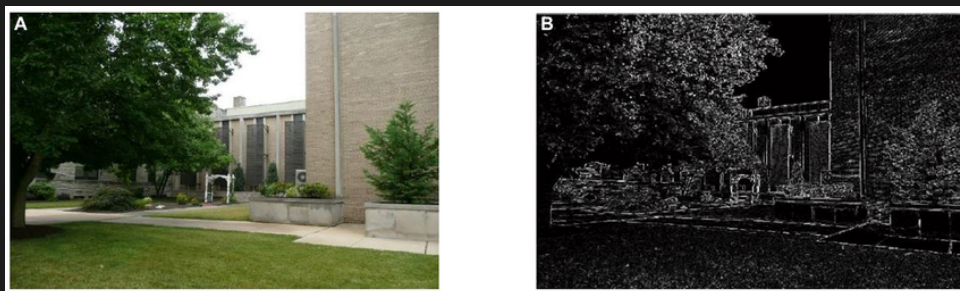
ویژگی‌های لبه‌ای:

- تراکم لبه‌ها با فیلتر سوبل
- برای تشخیص مرزهای واضح مثل خط ساحلی مفید است.

ویژگی‌های بافتی:

- کنتراست GLCM
- همگنی GLCM
- این ویژگی‌ها تفاوت بین مناطق یکنواخت و پر جزئیات را نشان می‌دهند.

مثال



شکل ۱: (A) تصویر نمونه، (B) تراکم لبه‌ها

نتایج و خروجی‌ها

- تمامی ویژگی‌های استخراج شده در فایل features.csv ذخیره شدند.
- ۱۱ ویژگی عددی برای هر تصویر
- برچسب کلاس مربوطه
- امکان استفاده در مراحل بعدی

جمع‌بندی

- ویژگی‌های انتخاب شده:
- تفاوت بین کلاس‌های مختلف طبیعت را نشان دهند
- نسبت به تغییرات جزئی در تصاویر حساس نباشند

- خوب برای مراحل بعدی خوشه‌بندی
این ویژگی‌ها امکان تفکیک صحیح تصاویر طبیعی را فراهم می‌کنند.

۲ فاز دوم: انتخاب ویژگی‌ها

اهداف

هدف از فاز دوم انتخاب بهینه‌ترین ویژگی‌ها از میان ۱۱ ویژگی استخراج شده بود. و به این منظور نیاز به ایجاد ماتریس همبستگی و تحلیل روابط بین ویژگی‌ها بود.

محاسبه ماتریس همبستگی

ماتریس همبستگی به صورت دستی با استفاده از فرمول زیر محاسبه شد:

$$r_{xy} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}} \quad (1)$$

برای اینکه میانگین ویژگی‌ها چندبار نیاز بود در ابتدا به صورت جدا حساب شد.

معیارهای انتخاب ویژگی

بهترین ویژگی‌ها در این پروژه، ویژگی‌هایی هستند که کمترین همبستگی را دارند تا بتوانیم تصاویر را با معیارهای متفاوت و دور از هم دسته‌بندی کنیم. برای پیدا کردن بهترین ویژگی‌ها ابتدا داده‌های ماتریس را مرتب کرده و کمترین آن‌ها را انتخاب کردیم ولی پی بردیم که این روش خوبی نیست چون ممکن چند داده که به هم شبیه هستند انتخاب شوند. بعد جمع این مقادیر را لحاظ کردیم که باز هم روش مناسبی نبود. و در آخر از واریانس کمک گرفتیم.
تابع `select_least_k_features` به منظور انتخاب بهینه‌ترین ویژگی‌ها برای عملیات خوشه‌بندی طراحی شده است. این تابع با ترکیب دو معیار اساسی شامل **میزان همبستگی بین ویژگی‌ها و میزان اطلاعات هر ویژگی**، ویژگی‌های نهایی را انتخاب می‌نماید.
فرمول اصلی مورد استفاده در این تابع به صورت زیر است:

$$i \text{ واریانس ویژگی} = \frac{\sum |i \text{ همبستگی‌های ویژگی}|}{i} \quad (2)$$

این تابع در چهار گام اصلی عمل می‌کند:

۱. محاسبه واریانس هر ویژگی به صورت جداگانه که نشان‌دهنده میزان پراکندگی و اطلاعات موجود در آن ویژگی است.
۲. محاسبه مجموع قدر مطلق مقادیر همبستگی هر ویژگی با سایر ویژگی‌ها از ماتریس همبستگی که بیانگر میزان وابستگی و افزونگی آن ویژگی است.

۳. محاسبه امتیاز ترکیبی برای هر ویژگی از طریق تقسیم مجموع همبستگی‌ها بر واریانس. این نسبت نشان می‌دهد که به ازای هر واحد اطلاعات، چه میزان افزونگی وجود دارد.

۴. انتخاب k ویژگی با کمترین امتیاز ترکیبی که بیانگر ویژگی‌های با بیشترین اطلاعات و کمترین وابستگی به سایر ویژگی‌ها است.

0.0	0.09	0.96	0.06	0.89	0.15	0.95	0.15	-0.14	-0.07	0.13
0.09	0.0	-0.08	0.98	-0.17	0.93	-0.1	0.94	0.64	0.63	-0.04
0.96	-0.08	0.0	-0.1	0.97	-0.0	1.0	0.0	-0.22	-0.14	0.11
0.06	0.98	-0.1	0.0	-0.2	0.96	-0.12	0.97	0.63	0.62	-0.03
0.89	-0.17	0.97	-0.2	0.0	-0.11	0.98	-0.09	-0.24	-0.16	0.11
0.15	0.93	-0.0	0.96	-0.11	0.0	-0.03	0.95	0.54	0.53	0.03
0.95	-0.1	1.0	-0.12	0.98	-0.03	0.0	-0.02	-0.23	-0.14	0.11
0.15	0.94	0.0	0.97	-0.09	0.95	-0.02	0.0	0.56	0.57	0.02
-0.14	0.64	-0.22	0.63	-0.24	0.54	-0.23	0.56	0.0	0.91	-0.57
-0.07	0.63	-0.14	0.62	-0.16	0.53	-0.14	0.57	0.91	0.0	-0.45
0.13	-0.04	0.11	-0.03	0.11	0.03	0.11	0.02	-0.57	-0.45	0.0

شکل ۲: ماتریس همبستگی بین ویژگی‌ها

۳ فاز سوم: خوشه‌بندی

اهداف اصلی

- پیاده‌سازی ۴ الگوریتم خوشه‌بندی مختلف
 - تنظیم پارامترهای هر الگوریتم
 - مقایسه عملکرد الگوریتم‌ها و انتخاب بهترین روش
 - تحلیل ویژگی‌های متمایزکننده هر خوشه
- در این فاز، پس از آماده‌سازی داده‌ها و نرمال‌سازی ویژگی‌های انتخاب شده، چهار الگوریتم خوشه‌بندی مختلف را پیاده‌سازی کردیم:

- **K-Means**: با تنظیم تعداد خوشه‌ها بر اساس دانش اولیه از دیتاست (۶ کلاس اصلی)
- **Agglomerative**: با استفاده از روش پیوند کامل
- **DBSCAN**: با پارامترهای بهینه‌شده برای شناسایی نواحی پرتراکم
- **MeanShift**: برای شناسایی خودکار تعداد خوشه‌ها

برای مقایسه الگوریتم‌ها از معیار سیلهوئت استفاده کردیم که هرچه بیشتر باشد بهتر است. در ۲ الگوریتم آخر اپسیلون و bandwidth با آزمون خطا انتخاب شد؛ اینکه بتوانند تعداد کلاستر معقولی ایجاد کنند و سیلهوئت بالاتری ایجاد شود.

DBSCAN با بالا بردن اپسیلون تعداد خوشه های معقولی ایجاد میکرد ولی شاخص کیفیت کاهش پیدا میکرد و در نهایت منفی میشد. برای meanshift با تغییر bandwidth و نزدیک ۱ قرار دادن تعداد دسته های خوبی و همچنین معیار سیلهوئت بالایی می داد اما با ادامه ی مراحل متوجه شدیم که به درستی دسته بندی نمی کند. و براساس معیار گفته شده الگوریتم kmeans را انتخاب کردیم

Clustering Algorithm Comparison:

Algorithm	Silhouette Score	Number of Clusters
KMeans	0.383623	6
Agglomerative	0.303646	6
DBSCAN	-0.25363	9
MeanShift	0.36268	7

Best Algorithm: KMeans
Silhouette Score: 0.384
Number of Clusters: 6

شکل ۳: نتایج آماری خوشه بندی

۴ فاز چهارم: نمایش بصری نتایج الگوریتم ها

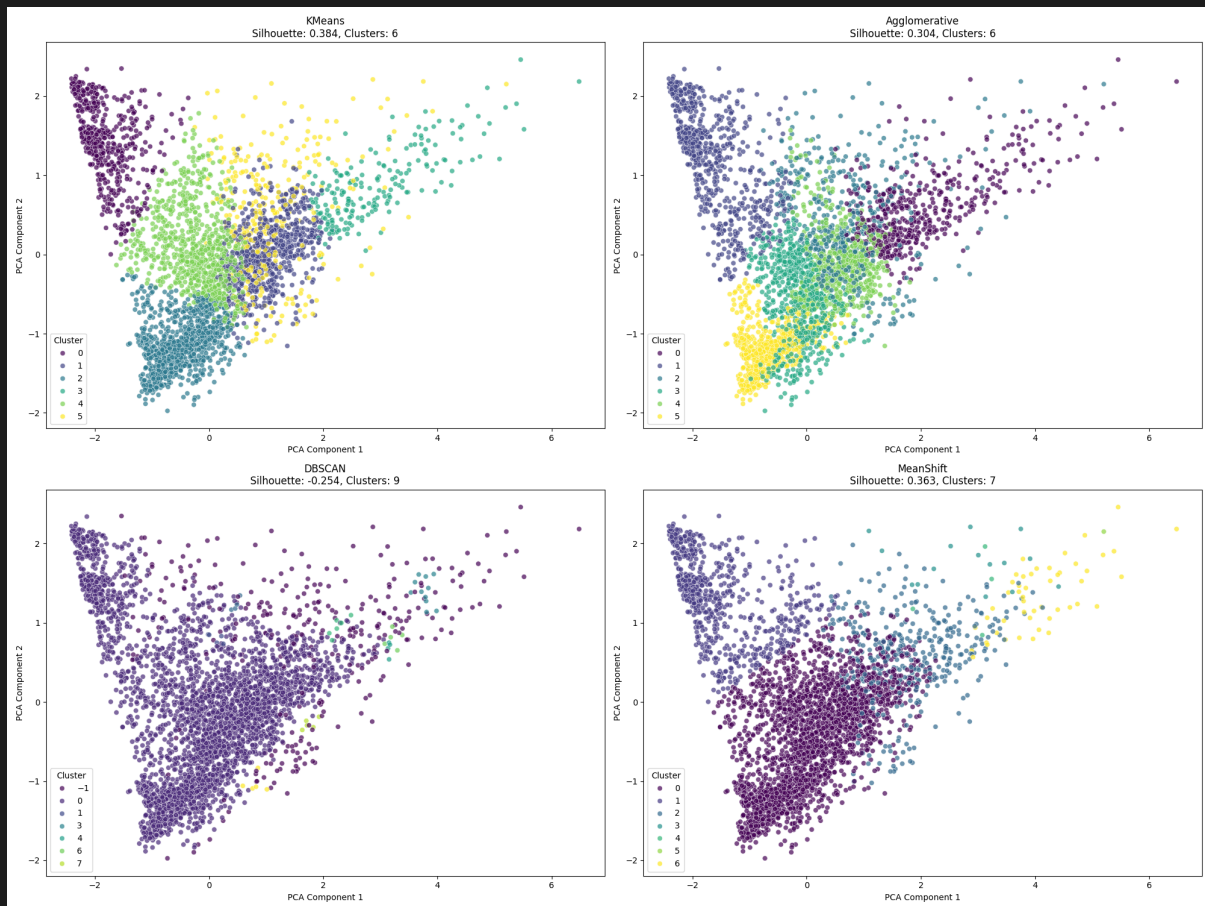
برای مصورسازی نتایج از کتابخانه های زیر استفاده شده است:

• **Matplotlib** (نسخه ۱.۵.۳):

- کتابخانه پایه برای ایجاد انواع نمودارها و ویژوال سازی ها
- استفاده شده برای ایجاد نمودارهای پراکندگی و تنظیم جزئیات ظاهری

• **Seaborn** (نسخه ۰.۱۱.۰):

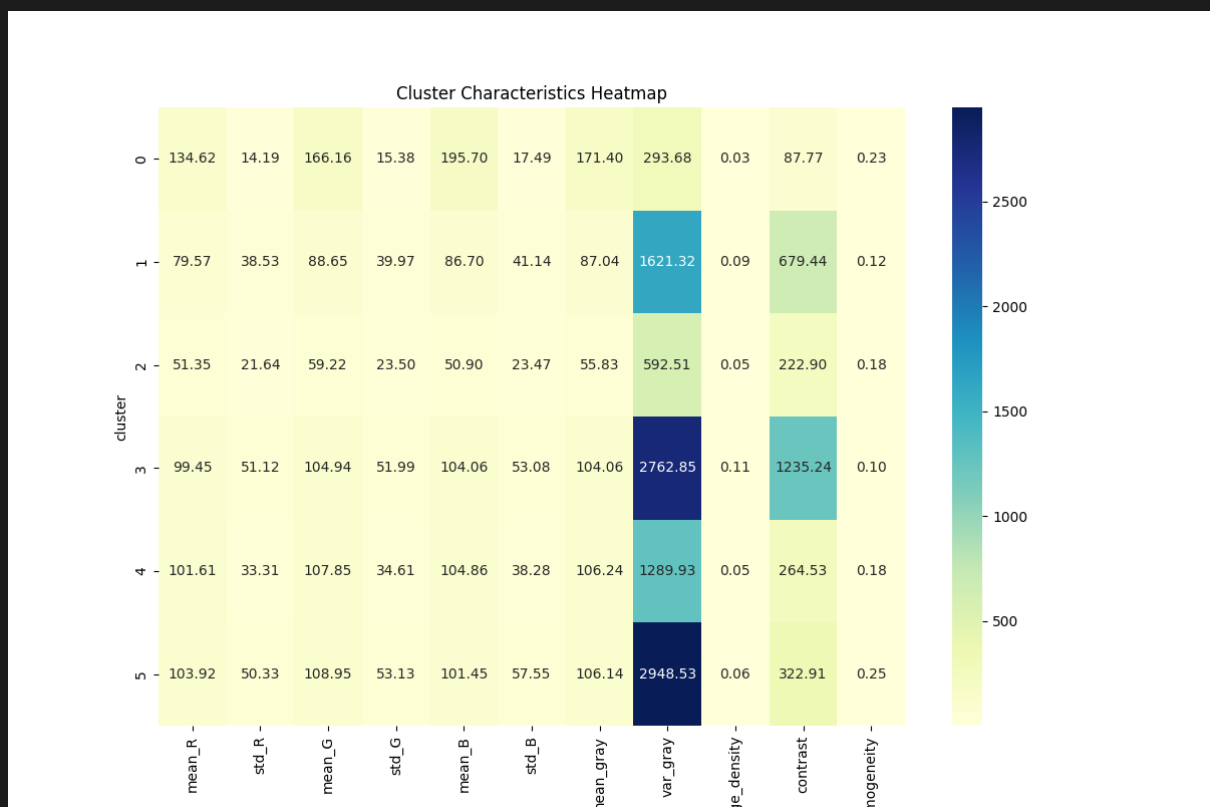
- کتابخانه سطح بالا مبتنی بر Matplotlib
- استفاده شده برای ایجاد هیتمپ و نمودارهای پراکندگی پیشرفته
- پالت رنگی viridis برای نمایش بهتر خوشه ها



شکل ۴: نتایج خوشه‌بندی تمام الگوریتم‌ها

هیت‌مپ

در طراحی این هیت‌مپ، برای هر یک از خوشه‌های شناسایی‌شده، میانگین مقدار هر ویژگی محاسبه شده است. این محاسبات بر روی داده‌های استانداردشده انجام گرفته تا امکان مقایسه معنادار بین ویژگی‌های مختلف فراهم شود. مقادیر به دست آمده سپس به دامنه‌ای بین ۰ تا ۱ نرمال‌سازی شده‌اند تا تغییرات رنگ‌ها به خوبی قابل تفسیر باشند. هیت‌مپ امکان شناسایی خوشه‌هایی با ویژگی‌های مشابه را فراهم کرده است و ویژگی‌های کلیدی که هر خوشه را از دیگران متمایز می‌کند، به وضوح قابل تشخیص بوده‌اند. همچنین یکنواختی یا عدم یکنواختی توزیع ویژگی‌ها در بین خوشه‌ها به راحتی قابل ارزیابی بوده است.



شکل ۵: هیت مپ

۵ فاز پنجم: ارزیابی خوشه‌بندی

اهداف اصلی

در این فاز، عملکرد الگوریتم‌های خوشه‌بندی با معیارهای مختلف ارزیابی شد. هدف اصلی سنجش کیفیت خوشه‌بندی انجام شده و تحلیل نتایج بود.

نتایج اجرای برنامه

خروجی کنسول پس از اجرای کد:

Clustering Evaluation Results:

Average Precision: 0.5878

Average Recall: 0.6033

Average F1-Score: 0.5964

Silhouette Score: 0.1472

```

Confusion Matrix:
Predicted 0 1 2 3 4 5
Actual
beach 21 18 82 0 291 188
dense_residential 0 318 124 132 24 2
desert 509 1 8 1 80 1
forest 0 56 542 0 2 0
intersection 0 283 47 67 168 35
sea_ice 5 47 222 1 229 96

Results saved to evaluation_results.csv

```

معیارهای ارزیابی

Silhouette:

این معیار با مقدار ۱۴۷۲.۰ نشان می‌دهد که ساختار خوشه‌بندی تا حدی مناسب است، ولی فضای بهینه‌ای بین خوشه‌ها وجود ندارد. این مقدار نشان می‌دهد که برخی نمونه‌ها نزدیک به مرز خوشه‌ها قرار گرفته‌اند.

Precision:

با مقدار متوسط ۵۸۷۸.۰ نشان می‌دهد که به طور متوسط حدود ۵۸٪ از نمونه‌های هر خوشه متعلق به کلاس غالب هستند. این نشان می‌دهد خوشه‌ها تا حدی خالص هستند.

Recall:

مقدار ۶۰۳۳.۰ نشان می‌دهد که حدود ۶۰٪ از نمونه‌های هر کلاس در خوشه مربوطه قرار گرفته‌اند.

F1-Score:

با مقدار ۵۹۶۴.۰ نشان می‌دهد که توازن نسبتاً خوبی بین دقت و فراخوانی وجود دارد.

تحلیل ماتریس درهم‌ریختگی

- کلاس desert بهترین عملکرد را با ۵۰۹ نمونه در خوشه صحیح دارد
- کلاس beach و ice sea بیشترین اختلاط را نشان می‌دهند
- برخی خوشه‌ها (مانند خوشه ۳) نمونه‌های کمی دارند

تفاوت معیارها

- تفاوت بین مقادیر معیارها به دلایل زیر است:
- Silhouette بر اساس فاصله‌های هندسی است
- Precision/Recall بر اساس تطابق با برچسب‌های واقعی است
- برخی مناطق طبیعی (مثل ساحل و یخ دریا) ویژگی‌های مشابهی دارند

جمع‌بندی

نتایج الگوریتم خوشه‌بندی:

- برای برخی کلاس‌ها (مانند بیابان) عملکرد خوبی دارد
- برای کلاس‌های با ویژگی‌های مشابه نیاز به بهبود دارد
- به طور کلی ساختار معقولی ایجاد کرده اما جای پیشرفت وجود دارد

۶ فاز ششم: پیش‌بینی و تحلیل خوشه‌ها

هدف اصلی این فاز، اعمال مدل آموزش‌دیده KMeans بر روی داده‌های تست و تحلیل نتایج بود. ابتدا داده‌ها را با اسکیلر آموزش‌دیده نرمال کردیم و پیش‌بینی را برای ۱۰ نمونه اجرا و در ادامه ۵ نمونه دیگر برای هر خوشه بررسی کردیم.

نتیجه خیلی دقیق نبود این به می‌تواند به دلیل انتخاب ویژگی‌ها باشد؛ چون از ویژگی‌های مربوط به رنگ و کنتراست لبه‌ها استفاده کردیم بعضی از تصاویر به اشتباه خوشه‌بندی شدند.

برای کلاستر ۰ که مربوط به dessert هست به دلیل رنگ بسیار متفاوت و تقریباً یک دست بودن تصاویر آن، بهتر از همه خوشه‌بندی شده است.

در کلاستر ۱، تعداد عکس‌های مربوط به dense-residential بیشتر از بقیه است؛ اما به دلیل وجود شمایل خانه هم در dense-residential هم در intersection کمی اختلاط دارد.

در کلاستر ۲ که بیشتر عکس‌های forest است، به دلیل وجود رنگ سبز در بعضی از عکس‌های beach، این تصاویر هم، در این کلاستر وجود دارند. همچنین به دلیل ساختار درختان کنار هم، که شبیه به ساختار یخ‌ها کنار هم و همچنین ساختمان‌ها کنار هم (از دور) است بعضی از عکس‌های sea-ice و dense-residential هم در این کلاستر وجود دارد.

در کلاستر ۳، بیشتر عکس‌های مربوط به intersection است ولی به دلیل گفته شده در بالا بعضی از عکس‌های dense-residential هم وجود دارد.

کلاستر ۴، مربوط به beach است؛ به دلیل وجود آب در این عکس‌ها و همچنین در sea-ice این دو کلاستر اختلاط دارند. به علاوه شباهت ساحل به صحرا نیز موجب وجود عکس‌هایی از desert در این کلاستر شده است.

کلاستر ۵ هم بیشتر مربوط به sea-ice است که همانطور که گفته شد با کلاستر ۴ اختلاط زیادی دارد.