

پهلوانان در شب یلدا



شب یلدا از راه رسیده؛ همون شبی که از قدیم گفتن بلندترین شبه و بهونه‌ایه برای دور هم جمع شدن. میدان شهر از عصر شلوغ شده، چراغ‌ها روشن، صدای خنده میاد و پهلوانان مثل هر سال آماده‌ان که جشن یلدا رو با مردم برگزار کنن. وسط میدان، سفره‌ی یلدا پهنه، هندونه‌ها قاچ شده‌ان، شمع‌ها روشن و همه منتظرن تا انارهای سرخ و خوش‌رنگ بیان روی سفره.

اما داستان به این سادگی‌ها هم نیست...

برای جشن امسال، از انبارها، بازارها و حجره‌های مختلف کلی عکس گرفته شده تا انارهای مخصوص جشن از توی اون‌ها جدا بشن. مشکل اینجاست که این عکس‌ها پر از چیزای رنگارنگه؛ پارچه‌های قرمز و سبز، لباس‌های یلدایی، سیب، هندونه، چراغ و هزار تا تزئین دیگه. خلاصه هر چی نگاه می‌کنی یه رنگ قرمز یه جایی پیدا می‌شه!



پهلوانان می‌دونن که توی هر عکس فقط یه خوشه‌ی انار واقعی مخصوص جشن وجود داره؛ یه انار درست و حسابی، سرخ، براق و خوش‌حال!

ولی نور بعضی عکس‌ها کمه، تو بعضیا سایه افتاده، تو بعضیا پس‌زمینه شلوغه و تشخیص اینکه کدوم قرمز، «انارِ خودِ اناره» و کدوم فقط یه چیز قرمزه، حسابی سخت شده.

چند تا از پهلوان‌ها نشستن عکس‌ها رو دونه‌دونه نگاه کردن، ولی بعد از چند ساعت چشمشون درد گرفت، سرشون گیج رفت و آخرش هم دیدن این‌جوری نمی‌شه ادامه داد. یکی‌شون گفت:

«این کار، کارِ دست نیست!»

اینجاست که همه‌ی نگاه‌ها به سمت شما برمی‌گرده...

1. معرفی پروژه

در این پروژه‌ی نهایی، شما باید یک سیستم **Image Segmentation** خودکار برای شناسایی یک شیء هدف مشخص (برای مثال انار) در تصاویر دیجیتال طراحی و پیاده‌سازی کنید.

سیستم شما باید از **segmentation مبتنی بر رنگ در فضای HSV** استفاده کند و پارامترهای **segmentation** باید به‌صورت خودکار با استفاده از **Genetic Algorithm** بهینه شوند. هدف اصلی، ساخت سیستمی است که قابلیت **generalization** بالایی داشته باشد و روی تصاویر دیده‌نشده عملکرد مناسبی ارائه دهد.

تنظیم دستی پارامترها مجاز نیست و تمام پارامترهای مهم **segmentation** باید از طریق فرآیند **optimization** یاد گرفته شوند.

2. تعریف مسئله

با دریافت یک تصویر ورودی، سیستم شما باید یک **binary segmentation mask** تولید کند به‌طوری که:

- پیکسل‌های متعلق به شیء هدف با مقدار 1 (سفید) مشخص شوند



- سایر پیکسل ها با مقدار 0 (سیاه) برچسب گذاری شوند

segmentation باید تا حد امکان دقیق، تمیز و مقاوم در برابر تغییرات نور، پس زمینه و ظاهر شیء باشد.

3. الزامات فنی

3.1 نمایش تصویر و فضای رنگی

- تصاویر به صورت استاندارد در قالب RGB ارائه می شوند (توجه داشته باشید که OpenCV تصاویر را به صورت پیش فرض با ترتیب BGR بارگذاری می کند).
- شما باید تصاویر را به فضای رنگی HSV تبدیل کنید.
- منطق segmentation باید عمدتاً مبتنی بر HSV thresholds باشد.

3.2 بهینه سازی با Genetic Algorithm

استفاده از Genetic Algorithm برای بهینه سازی پارامترهای segmentation الزامی است.

طراحی GA شما باید شامل موارد زیر باشد:

1. Genome Design

- یک genome که پارامترهای segmentation (مانند HSV thresholds) را نمایش دهد.

2. Fitness Function

- باید یک Fitness Function تعریف کنید که mask پیش بینی شده را با ground-truth mask مقایسه کند.
- انتخاب Fitness Function باید به طور کامل توجیه شود، به خصوص در صورتی که انواع مختلف خطا اهمیت متفاوتی دارند.



3. Evolutionary Operators

- روش selection
- عملگر crossover
- عملگر mutation

4. آموزش و ارزیابی (مهم)

برای اینکه راه حل شما «واقعاً قابل اعتماد» باشد، کافی نیست فقط روی همان عکس هایی که با آن ها کار کرده اید نتیجه ی خوب بگیرید؛ باید طوری طراحی شود که وقتی یک عکس جدید و ندیده به آن می دهیم هم بتواند انار را درست جدا کند.

به همین دلیل، تصاویر پروژ (مجموعه عکس هایی که ما در اختیارتون می گذاریم و خودتون جمع آوری می کنید) را از همان ابتدا به 2 بخش جدا تقسیم کنید:

- **بخش اول (بزرگتر):** برای اینکه Genetic Algorithm با آن ها کار کند و پارامترهای segmentation را پیدا کند.
- **بخش دوم (کوچکتر):** برای اینکه در طول طراحی و تنظیمات، کیفیت راه حل تان را روی عکس هایی که «مستقیماً با آن ها بهینه سازی نشده» بررسی کنید و تصمیم بگیرید کدام تنظیمات بهتر است.

نمره ی نهایی پروژه بیشتر از هر چیز به این وابسته است که سیستم شما روی عکس های جدید و دیده نشده چقدر خوب و پایدار عمل می کند.

5. توسعه های اختیاری مدل

★ استفاده از Multiple Masks و ترکیب آن ها با AND، OR و NOT (مثال رنگ قرمز که در کلاس صحبت کردیم)

★ برای افزایش کیفیت segmentation پیشنهاد می شود از Pipeline زیر استفاده کنید:



1. Pre-processing

- Blurring (مانند mean یا Gaussian filter)

2. Segmentation

3. Post-processing

- Morphological operations

★ از هر ایده خلاقانه ای به شدت استقبال می کنیم.

6. ارزیابی و رقابت

- ارزیابی نهایی روی یک **hidden test dataset** انجام می شود.
- یک leaderboard بر اساس معیار ارزیابی اعلام شده تشکیل خواهد شد.
- راه حل های برتر نمره امتیازی بیشتری دریافت خواهند کرد.

7. معیارهای نمره دهی

پروژه بر اساس موارد زیر ارزیابی می شود:

- صحت پیاده سازی
- کیفیت طراحی Genetic Algorithm
- اثربخشی Fitness Function
- عملکرد روی داده های دیده نشده
- عمق تحلیل و استدلال
- رتبه در رقابت