

تشخیص چهره:

تابع `loadImage` مطابق صورت مسئله نوشته شد. تابعی به نام `normalImage` برای نرمالایز کردن بردار عکس ها نوشته شد. در تابع `loadFaces`، تمام عکس ها به صورت یک بردار نرمالایز شده ۹۰۰ تایی در ستون های یک ماتریس ذخیره میشوند و خروجی ای با ابعاد ۹۰۰ در ۴۰۰ خواهیم داشت. سپس در تابع `cov`، ماتریس کواریانس محاسبه میشود. برای پیدا کردن `eigenaFaces` ابتدا باید مقدار ویژه و بردار ویژه های ماتری کواریانس را حساب کنیم. `eigenFaces`، بردارهای ویژه ای که بیشترین مقدار ویژه ی متاظر را دارا هستند، خواهند بود. بنابر این لازم است مقادیر ویژه و بردار های ویژه را `sort` کنیم و بردارهایی که بیشترین مقدار ویژه را دارند به عنوان آیکن `eigenFaces` انتخاب کنیم. برای `sort` کردن از تابع `argsort` روی مقادیر ویژه استفاده شده، سپس بردار های ویژه هم با همان ترتیب `sort` میشوند و ۲۵ تایی که بزرگترین مقدار ویژه را داشتند به عنوان آیکن فیسز انتخاب میشوند. در تابع `show` هم هریک از آیکن فیس ها به عکسی ۳۰ در ۳۰ تبدیل شده و نمایش داده میشوند.

در تابع `convertFaces`، عکس یا عکس هایی که به تابع داده شده اند به بردار هایی با ۲۵ المان تبدیل میشوند بدین صورت که هر المان از آنها حاصل ضرب داخلی آن با یکی از آیکن فیس هاست.

در تابع `دیتاست`، تمامی عکس ها به صورت بردار هایی کانورت شده، در کنار اسم صاحب عکس در یک لیست ذخیره میشوند. در نهایت در تابع `knn`، با استفاده از تابع `distance`، فاصله ی عکس ورودی (که ابتدا باید کانورت شود!) از تمامی عضوهای دیتاست محاسبه میشود و فاصله عکس ورودی با هر یک از اعضا به همراه اسم آن عضو ذخیره میشوند. سپس از بین داده های ذخیره شده، آنهایی که کمترین فاصله را با ورودی دارند جدا شده و بررسی میشود کدام اسم در بین آنها از همه بیشتر تکرار شده است. در نهایت اسم پرتکرار و تمامی `k` همسایه ی نزدیک به ورودی به عنوان خروجی تابع بازگردانده میشوند.

بازی ریاضی:

در تابع `generate`، آرایه ای ۱۰ در ۱۰ از اعداد رندم سه رقمی توسط تابع `random.randint` ساخته میشود و سپس تمامی اعضای که مکانشان مضرب ۹ است را دارای مقدار یکسان میکنیم. سپس با تابع `to_str`، آرایه ای ۱۰ در ۱۰ میسازیم که اعداد از ۱ تا ۱۰۰ را به فرم مطلوب صورت سوال (12:743) به کدهای رندم تولید شده ی خود متصل کنند تا برای نمایش در جدول استفاده کنیم.. در نهایت در تابع `play` با استفاده از ابزار `plt.table` جدول را ترسیم میکنیم. یک آپشن اضافه هم که تعبیه شده این است که اگر در تابع `play` از آرگومان `pink=1` استفاده کنید، میتوان جدول را با تم صورتی خیلی خوشگلی مشاهده کنید. 😊