



Amirkabir University of Technology
(Tehran Polytechnic)



Electrical Engineering Department

Multimedia – Dr.Sharifian

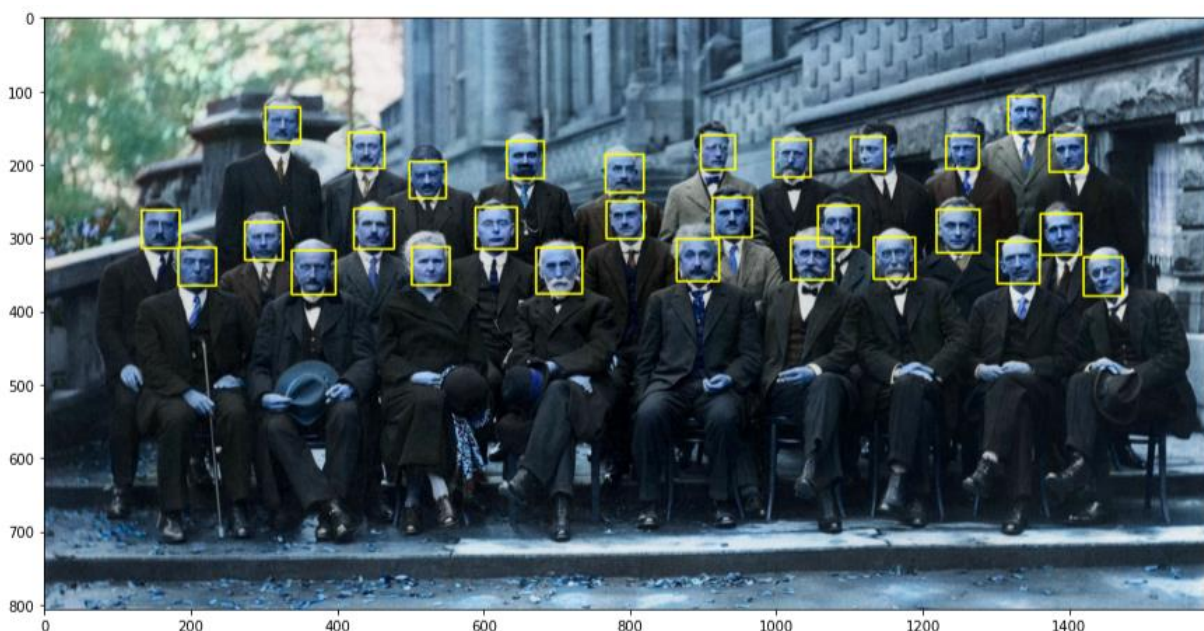
Report HW 2

نام دانشجو: علی بابالو – ۹۸۲۳۰۱۲

ایمیل: alibabaloo@aut.ac.ir

سوال ۱:

برای قسمت اول برای تشخیص تعداد چهره های عکس از مدل HAAR frontface استفاده کردیم که شامل ۳ مدل default, alt, alt2 می باشد که ۲ مدل دیفالت و alt2 را با یکدیگر مقایسه کردم که در مدل alt2 نتایج به درستی و تعداد چهره ها ۲۹ عدد تشخیص داده شد. برای رسیدن به این نتیجه مقدار پارامتر minNeighbour را تغییر میدادم تا به نتیجه مطلوب برسیم.



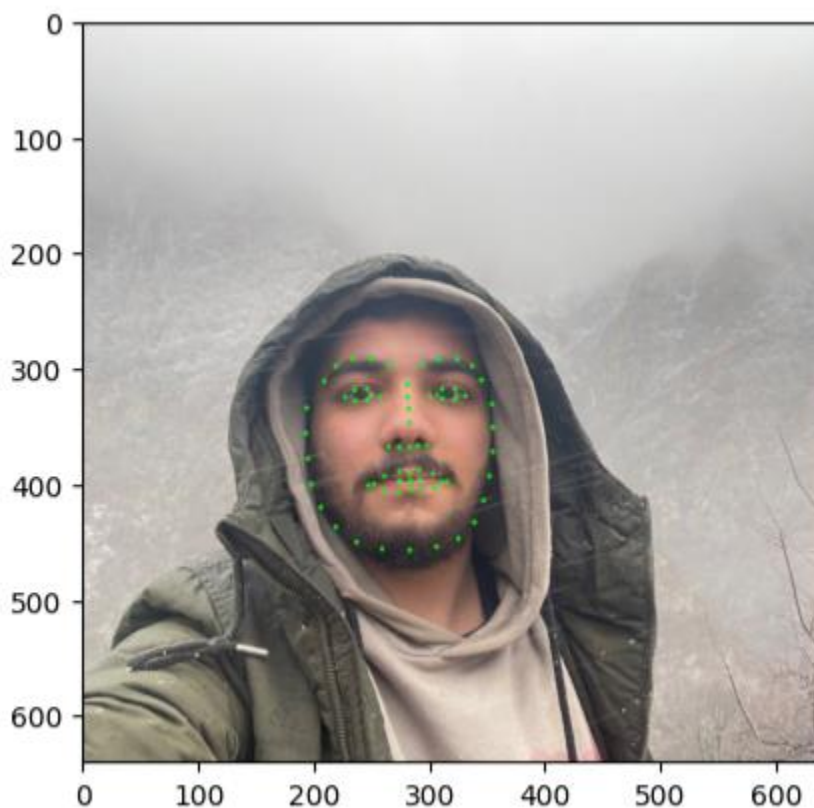
برای قسمت دوم این سوال بعلت اینکه لپتاپ بنده وبکم ندارد از یک ویدیو از پیش ضبط شده استفاده کردم. ابتدا ویدیو را با استفاده از videocapture خوانده و با تابع read هر فریم آن را در حلقه می خوانیم. سپس برای هر فریم از مدلی که در

قسمت قبل استفاده کردیم بهره می‌بریم تا چهره را تشخیص دهد و آن را با مستطیل نشان دهد. برای اتمام ویدیو نیز ۲ روش قرار دادیم یکی بعد از اینکه فریم آخر خوانده شد بصورت خود به خود خارج می‌شود حالت دوم بدین صورت است که با فشردن کلید esc می‌توان خارج شد.

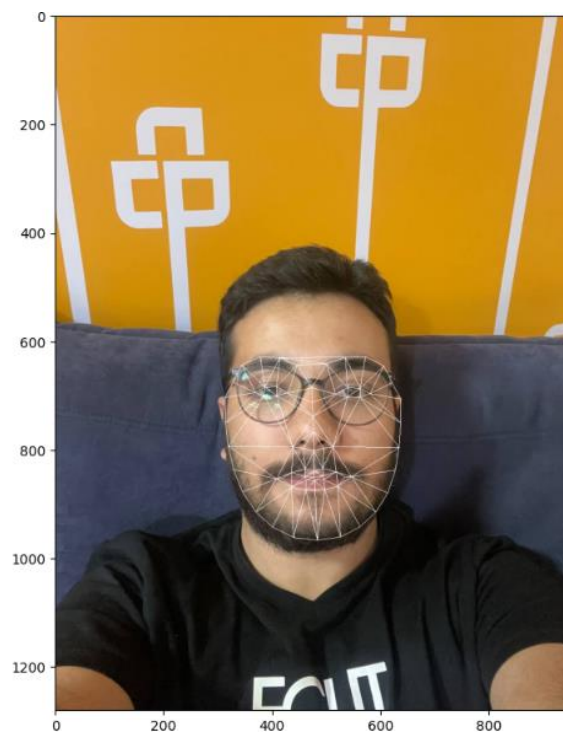
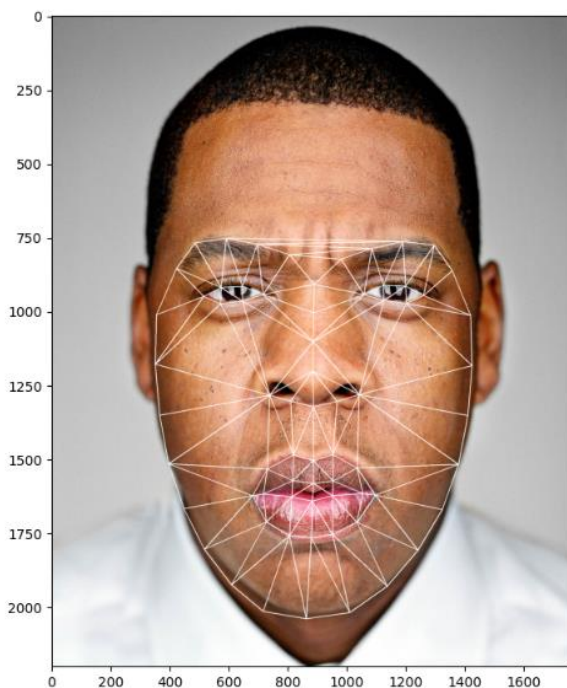
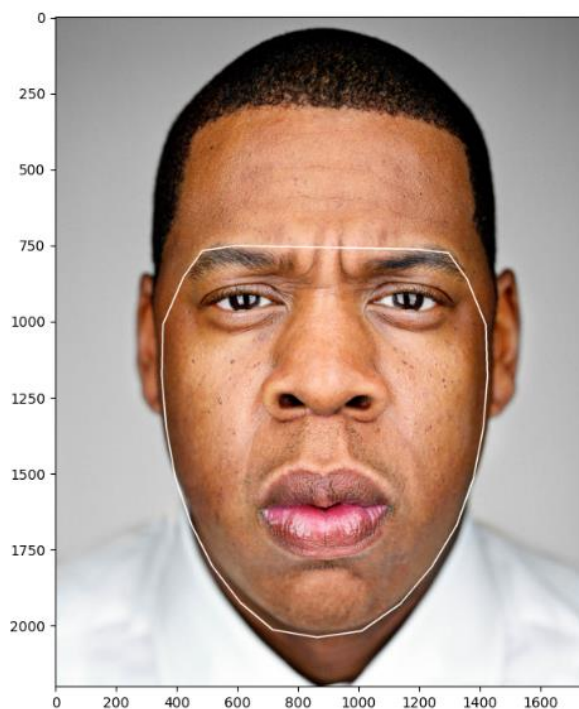
برای سنجش عملکرد کد نیز زمان مورد نیاز برای پردازش 200 فریم محاسبه شده و از روی آن میانگین زمان پردازش هر فریم و نیز FPS به دست می‌آید. مقدار FPS طبیعتاً به میزان پردازش مورد نیاز برای هر فریم بستگی دارد. یکی از عوامل اصلی در این مورد اندازه‌ی تصویر است، هر چه اندازه‌ی تصویر بزرگ‌تر باشد، میزان پردازش مورد نیاز بیش‌تر و FPS کمتر است و برای این ویدیو از سایز 500×720 استفاده کردم که بنسبت سایز بزرگی است و به همین دلیل مقدار FPS کم شده است.

سوال ۲:

برای قسمت اول این سوال بعلت پولی بودن نتوانستم تصویر خوبی از fotor جنریت کنم بهمین علت از تصویر آقای Jay-Z استفاده کردم تا لندمارک‌ها را بر روی آن تشخیص دهم. برای پیدا کردن نقاط اساسی از کتابخانه dlib و مدل `arks_face_68_predictor_s` استفاده کردم که 68 نقطه اساسی بر روی صورت را مشخص می‌کند و دایره‌های آن را بر روی تصویر سلبریتی و همچنین تصویر خودم کشیدم.



برای قسمت دوم و تعویض 2 چهره، ابتدا نقاط کلیدی را مانند قسمت قبل بدست می‌آوریم سپس بدور چهره یک `covex` hull میکشیم و این نقاط را با توجه به درس استاد بصورت مثلی به یکدیگر متصل می‌کنیم.



برای قسمت بعد تنها کافی است آن مثلث هارا از تصویر یک به تصویر دوم منتقل کنیم(البته چون اسکیل تصاویر با هم یکی نبودند تصویر نهایی خیلی خوب نشد) برای این کار از متدهای `getAffineTransform` و `warpAffine` استفاده می کنیم؛ متد اول با در اختیار داشتن دو مثلث تبدیلی را به ما می دهد که با اعمال آن بر یک تصویر، مثلث اول به مثلث دوم تبدیل می شود، متد دوم نیز این تبدیل به دست آمده را بر روی تصویر اعمال می کند. در نهایت نتیجه بدست آمده را روی تصویر دیگر می اندازیم.

پایان