بسمه تعالی

گزارش پروژه پردازش تصویر

پروژه سوم: تشخیص خواب آلودگی

مهربد اعتمادی (1400012268036) پارسا امیدی (1400012268028) هانیه وطنی (1400012268035)

- **Downloading the dataset**

دیتاسِت مورد استفاده در این پروژه دیتاست nthuddd2 می‌باشد که از Kaggle دریافت شده است و از کیفیت بسیار بالایی برخوردار است. این دیتاسِت در پژوهش و مقالات زیادی استفاده شده و شامل حالت‌های مختلف چهره، چشم و دهان می‌باشد که همگی این تصاویر grayscale می باشند.

- **Directory Specification**

1. Specifying a directory for split files

هدف این قسمت ایجاد یک دایرکتوری تمیز و منظم شده از دیتاست است. یک پوشه دیتاست اصلی داریم که به عنوان سورس فایل خام در نظر گرفته‌شده‌است و حاوی عکس‌ها قبل از پردازش است. این پوشه دو زیرپوشه تقسیم‌شده‌است که دو حالت «drowsy» و «notdrowsy» را در بر می گیرد.

یک دایرکتوری خروجی (Output directory) نیز برای ذخیره دیتای پراسس شده و مرتب ساخته شده است که آماده استفاده برای مدل هستند. این پوشه حاوی سه زیرپوشه training, validation, test می‌باشد که دیتاهایی که برای هر یک استفاده شده است را ذخیره می کند. هر زیرپوشه خود حاوی دوبخش است که حالات «drowsy» و «notdrowsy» را نشان می‌دهند.

2. Organizing the data into train, validation, test

دیتاست به سه بخش تقسیم می‌شود: Training set, Validation set, Test set. برای این پروژه 80% دیتا برای ترین کردن، 10% برای ولیدیشن و 10% برای تست در نظر گرفته شده‌است تا مدل را بتوان به خوبی ترین کرد. این بخش‌ها سپس در پوشه‌های مربوطه ذخیره می‌شوند.

3. Defining the output path for divided dataset

مسیر پوشه‌های مختلف معلوم و در برنامه تعریف شده‌اند تا مدل با استفاده از مسیرها ترین شود. یک image data generator نیز اعمال شده‌است تا تکنیک‌های data augmentation را اعمال کند. این image data generator پروسه‌ی بارگیری و انتقال دیتا به مدل را انجام می‌دهد و هندل کردن دیتاست‌های بزرگ و تکنیک‌های data augmentation را ممکن می‌کند. data augmentation تکنیکی در ماشین لرنینگ می‌باشد که تغییراتی را بر دیتای موجود اعمال کرده و سایز و چگالی دیتاست را افزایش می‌دهد. این تغییرات ورژن‌های دیگری از تصاویر موجود می‌سازد که برای بهبود کارایی مدل نقش به‌خصوصی دارند. این تغییرات می‌توانند شامل موارد زیر باشند:

- Geometric Transformations
- Color Transformations
- Noise Addition
- Other Transformations

این تکنیک به دلایل مختلفی استفاده می‌شود؛ مثلاً بزرگ کردن اندازه دیتاست، جامع‌تر کردن مدل، کم کردن overfit و بالانس کردن توزیع کلاس‌های مختلف (در این پروژه «notdrowsy»و «drowsy»).

تکنیک‌های استفاده شده در این پروژه بدین صورت هستند که در همان ابتدا اعمال می‌شوند:

- Rescaling: با این‌کار تصاویر را نرمالایز می‌کنیم.
- Rotation: به طور رندوم تصویر را با یک زاویه مشخص (30 درجه) میچرخاند بدین معنی که تصویر می‌تواند به طور ساعت‌گرد یا غیر ساعت‌گرد تا زاویه مشخص شده بچرخد.
- Horizontal flip: تصویر را به صورت افقی می‌چرخاند.

- **Samples**

1. **Train samples**
2. **Validation samples**
3. **Test samples**

در تمامی این سمپل‌ها سایز (224 و 224) بوده و اندازه batch به 32 تنظیم شده‌اند. این batch size بدین معنی است که مدل هر چند عکس را به طور همزمان دریافت و پردازش می‌کند. هم‌چنین shuffle به روی True تنظیم شده تا مدل عکس‌ها را به طور تصادفی انتخاب کند.

- **Model defining**

1. **Loading the base MobileNet model**

در این پروژه از مدل MobileNet استفاده شده است که یک مدل پری - ترینند می‌باشد و CNN-based است. این مدل مدلی محبوب در مقالات پژوهشی است. قبل از اضافه کردن لایه‌های دیگر باید لایه‌های مدل پری - ترینند شده را در طول ترین کردن به اصطلاح «فریز» کنیم. فریز کردن بدین معنی است که وزن این لایه‌ها در طول ترین کردن تغییر نمی‌کند.

2. **Adding custom layers on model**

این لایه‌ها باید بر روی مدل موبایل‌نت اضافه شوند تا مدل را برای تشخیص خستگی آماده کنند. این لایه‌ها عبارت‌اند از:

- Global Average Pooling Layer
- Dense Layer
- Dropout Layer
- Output Layer

3. **Training the model**

دیتاست ترینینگ را به صورت بچ به مدل می‌دهیم و مدل یاد می‌گیرد چگونه تصاویر را کلاس‌بندی کند. سپس دیتاست validation استفاده می‌شود تا کارایی مدل را در طور پروسه یادگیری اندازه‌گیری کند. این مسئله کمک می‌کند تا overfit را تشخیص دهد. Epoch پروژه نیز با توجه به اندازه دیتاست یک در نظر گرفته‌شده‌است.

4. Fine-tuning the model

بعد از ترین شدن اولیه، مدل از fine-tuning استفاده می‌شود تا کارایی را بهبود بخشد. این پروسه شامل آنفریز کردن تعدادی از لایه‌های اصلی مدل MobileNet و ترین کردن آن‌ها با یک نرخ یادگیری پایین‌تر است. در این پروژه از callback استفاده شده که ابزاری است برای مشاهده و کنترل پروسه‌ی ترین کردن. Callback ها در اقدامات مشخصی استفاده می‌شوند:

- Early Stopping: متوقف کردن پروسه‌ی ترین کردن در صورتی که کارایی مدل بر روی validation set بهبود نیابد؛ این موضوع از overfit شدن جلوگیری کرده و در زمان صرفه‌جویی می‌کند.
- Saving the Best Model: ذخیره کردن مدلی که در طول پروسه‌ی ترین کردن دارای بالاترین کارایی است.

مدل در طول fine-tuning با پنج epoch ترین خواهد شد.

- **Predicting the labels for the test data**

بعد از پروسه‌ی ترین کردن و fine-tuning، نوبت به اعتبارسنجی کارایی بر روی دیتاست تست می‌باشد که شامل پیش‌بینی کردن لیبل‌های دیتاست تست توسط مدل و مقایسه‌ی آن با لیبل‌های واقعی می‌باشد. این پروسه شامل این مراحل است:

- Loading the Model: مدل ترین و fine-tuned شده از فایل سیو شده لود می‌شود.
- Generating Predictions: از مدل برای پیش‌بینی لیبل‌ها استفاده می‌شود. خروجی این مرحله یک Probability Score است که بین صفر و یک است و نشان‌دهنده احتمال «drowsy» بودن عکس است.
- Converting Predictions to Binary Labels: اعداد بدست آمده در مرحله قبل به یک عدد باینری (0 یا 1) تبدیل می‌شود. اعداد بالای 0.5 در دسته drowsy و اعداد پایین آن در دسته «notdrowsy» قرار می‌گیرند.

- True Labels: لیبل‌های واقعی دیتاست تست از test data generator استخراج می‌شوند.

1. Classification Report

این گزارش شامل اطلاعات کاملی از کارکرد مدل برای هر کلاس است که بخش‌های متعددی مثل precision, recall, F1-score, support دارد.

- Precision: معیاری است که اندازه می‌گیرد که مدل چه مقدار دقت در پیش‌بینی درستی برای کلاس «drowsy» داشته‌است. فرمول آن به صورت زیر است:

$$\text{Precision} = \frac{True\ Positives}{True\ Positives\ +\ False\ Positives}$$

- Recall: این معیار در واقع مشخص می‌کند که مدل، چه مقدار از کلاس «drowsy» را از دست نداده و درست پیش‌بینی کرده است. فرمول آن به صورت زیر است:

$$\text{Recall} = \frac{True\ Positives}{True\ Positives\ +\ False\ Negatives}$$

- F1-Score در بخش بعدی بررسی شده است.

2. F1-Score

این معیار جفت precision و recall را ترکیب کرده و به یک مقدار تبدیل می‌کند. این معیاری در واقع یک میانگین هارمونیک از precision و recall است که به صورت زیر محاسبه می‌شود:

$$\text{F1-Score} = \frac{Precision*Recall}{Precision+\ Recall}$$

از مزایای این معیار می‌توان به این اشاره کرد که تناسب بین Precision و Recall را برقرار می‌سازد و مخصوصاً زمانی مناسب است که توزیع کلاس‌ها نامساوی است. این معیار نیز مانند دو معیاری قبلی بر کلاس positive که در اینجا همان کلاس «drowsy» است تمرکز دارد.

3. Confusion Matrix

جدولی است که پیش‌بینی‌های مدل را با مقایسه آن‌ها با لیبل‌های اصلی خلاصه کرده و نشان می‌دهد. این جدول شامل موارد زیر است:

- True Positives: پیش‌بینی شده‌های درست کلاس «drowsy»
- True Negatives: پیش‌بینی‌شده‌های درست کلاس «notdrowsy»

- False Positives: پیش‌بینی شده‌های غلط کلاس «drowsy»
- False Negatives: پیش‌بینی شده‌های غلط کلاس «notdrowsy»

حال به پیش‌پردازش‌های انجام شده به روی دیتاست و نتایج آن‌ها می‌پردازیم.

• CLAHE

1. Apply CLAHE

برای اعمال CLAHE به روی تصاویر باید آن‌ها را به فرمت LAB تبدیل کنیم. الگوریتم CLAHE بر روی کانال L که نشان‌دهنده Lightness است اعمال می‌شود و A و B کانال‌های رنگی آن هستند. سپس پس از اعمال تغییر بر کانال L آن‌ها با دو کانال دیگر ادغام شده و نتیجه نهایی را حاصل می‌کنند. پس از تمامی این مراحل تصویر به فرمت RGB بازگشته و نرمال می‌شود.

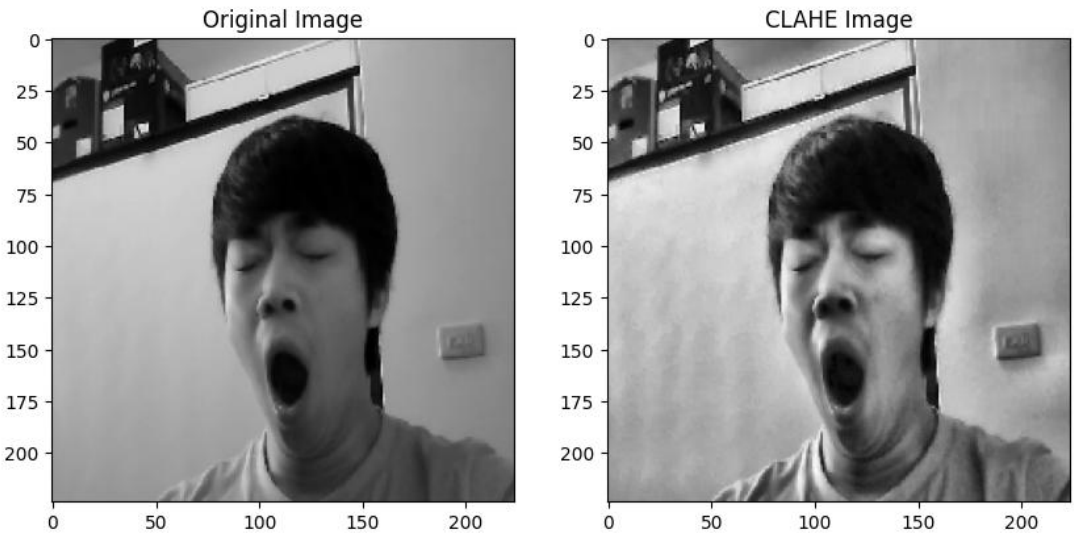
2. Apply on all images

این روند برای تمامی پیش‌پردازش‌ها یکسان بوده و در اینجا برای نمونه توضیح داده می‌شود و در ادامه به منظور پرهیز از تکرار بازگو نمی‌شود.

برای اعمال CLAHE روی تمام تصاویر این روند انجام می‌شود:

- ریست کردن test generator
- Iterate Through the Test Generator: الگوریتم روی تمام تصاویر یک batch اعمال می‌شود.
- مدل روی تصاویر تغییر یافته پیش‌بینی انجام می‌دهد.
- Probability Scoreها به مقادیر باینری تبدیل شده تا دو کلاس را مشخص کنند.
- کلاس‌های پیش‌بینی شده و کلاس‌های صحیح جفت ذخیره میشوند.
- مدل اعتبارسنجی می‌شود.

3. Displaying a sample



همان‌گونه که مشاهده می‌شود کنتراست تصویر به خوبی تغییر یافته و با تصویر باکیفیت‌تری مواجه هستیم.

4. Classification Report

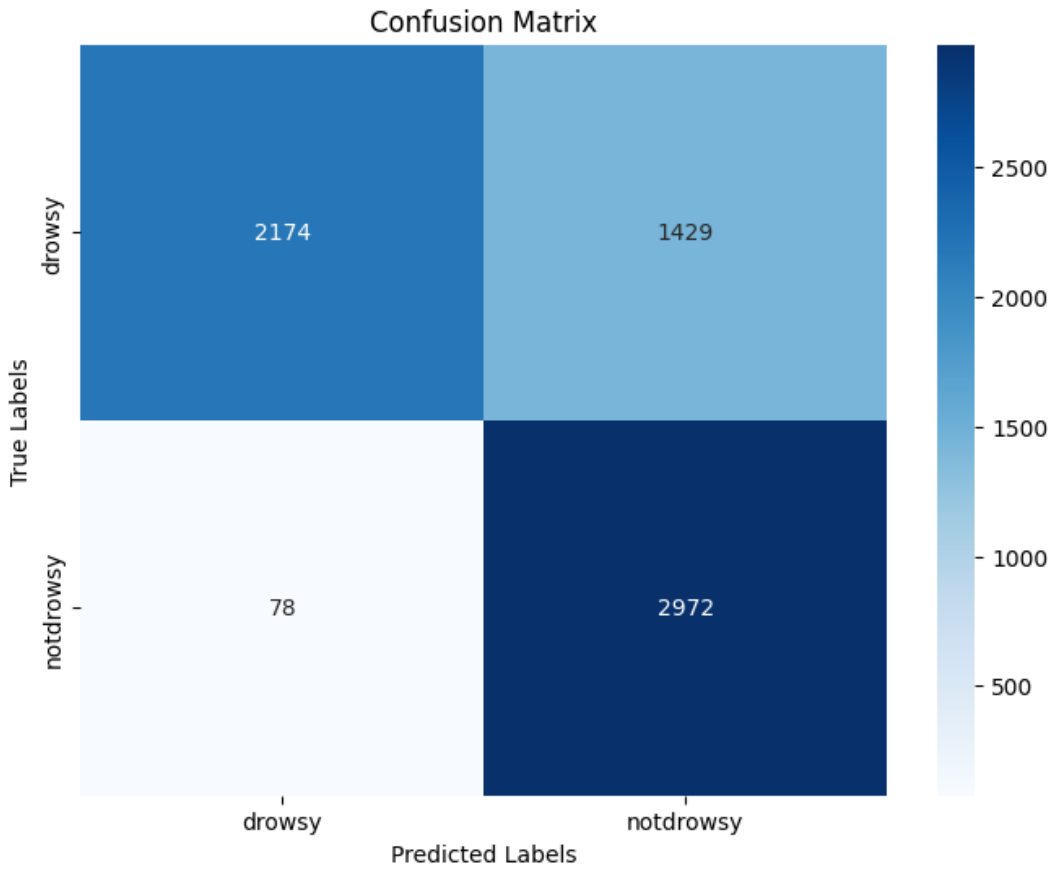
Classification Report

	Precision	Recall	F1-Score	Support
drowsy	0.97	0.60	0.74	3603
notdrowsy	0.68	0.97	0.80	3050
accuracy			0.77	6653
macro average	0.82	0.79	0.77	6653
weighted average	0.83	0.77	0.77	6653

5. F1-Score

F1-Score: 0.7977452690913971

Confusion Matrix .6

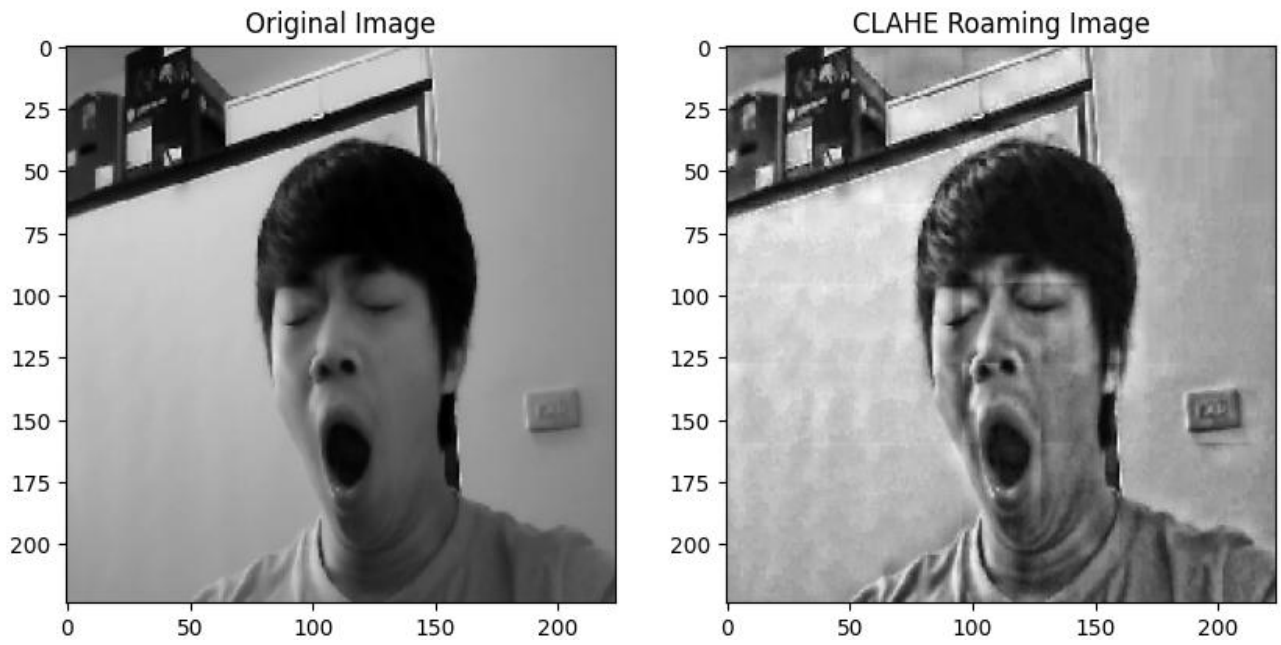


• CLAHE Roaming

1. Apply CLAHE Roaming

این تابع الگوریتم CLAHE را با استفاده از یک Sliding Window به روی تصویر اعمال می‌کند و بر خلاف الگوریتم اولیه که CLAHE را به طور همزمان بر کل تصویر اعمال می‌کند؛ این تابع تصویر را به چند قسمت تقسیم کرده (این چند قسمت هم‌پوشانی دارند) و CLAHE را به روی هر قسمت اعمال می‌کند. پنجره استفاده شده در این پروژه سایز 64*64 است.

2. Displaying a sample



3. Classification Report

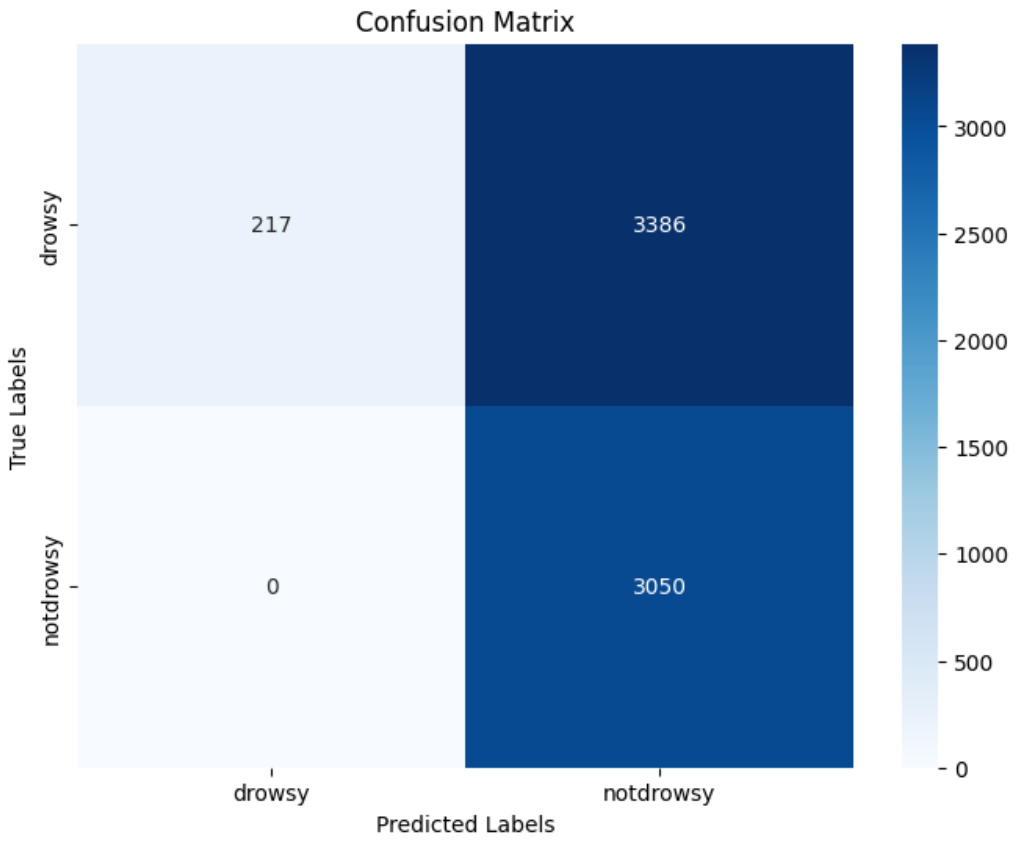
Classification Report

	Precision	Recall	F1-Score	Support
drowsy	1	0.06	0.11	3603
notdrowsy	0.47	1	0.64	3050
accuracy			0.49	6653
macro average	0.74	0.53	0.38	6653
weighted average	0.76	0.49	0.36	6653

F1-Score .3

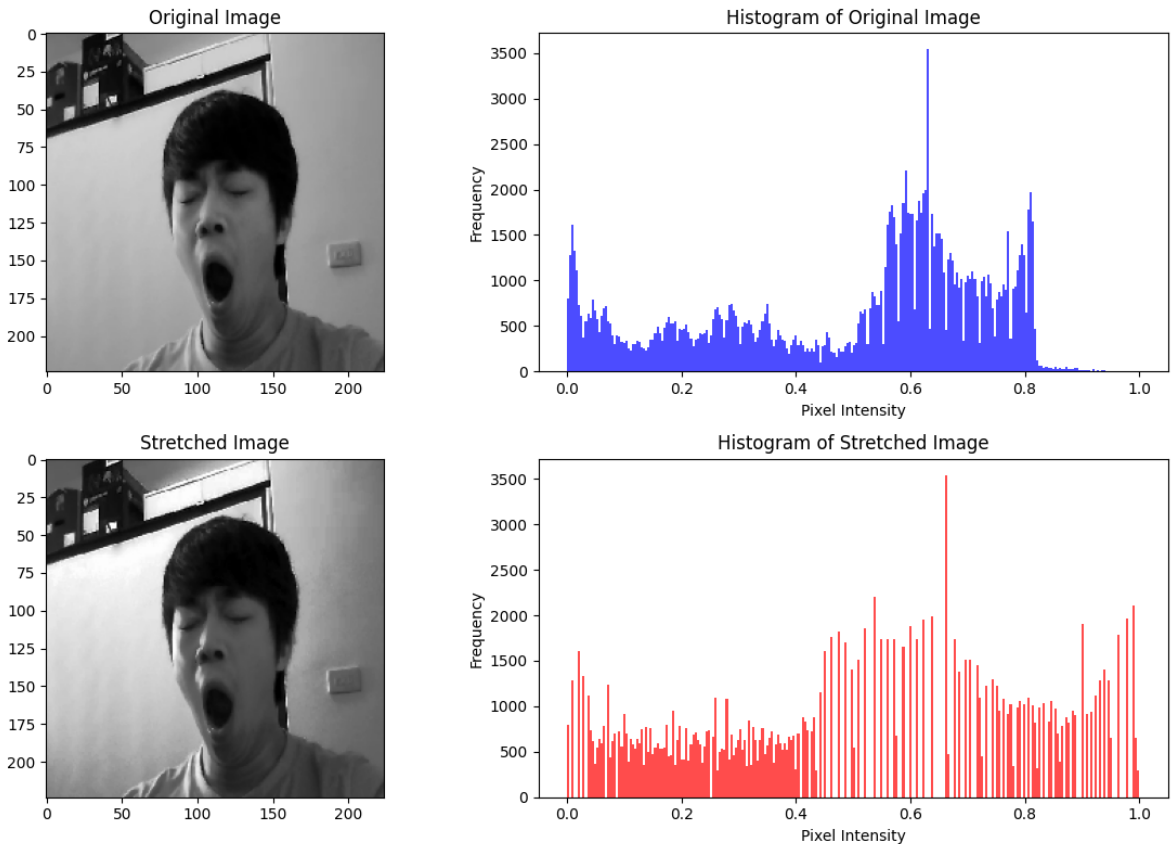
F1-Score: 0.6430529200927683

Confusion Matrix .4



Histogram stretching •

Displaying a sample .1



Classification Report .2

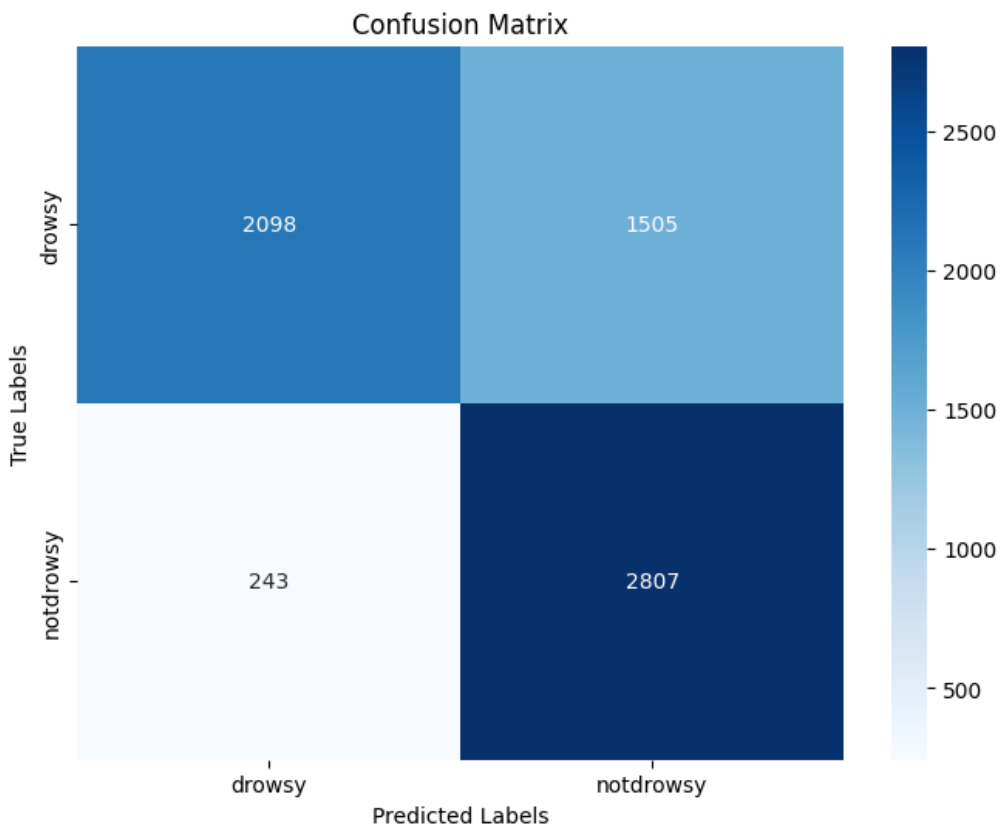
Classification Report

	Precision	Recall	F1-Score	Support
drowsy	0.90	0.58	0.71	3603
notdrowsy	0.65	0.92	0.76	3050
accuracy			0.74	6653
macro average	0.77	0.75	0.73	6653
weighted average	0.78	0.74	0.73	6653

F1-Score .3

F1-Score: 0.7625645205107308

Confusion Matrix .4

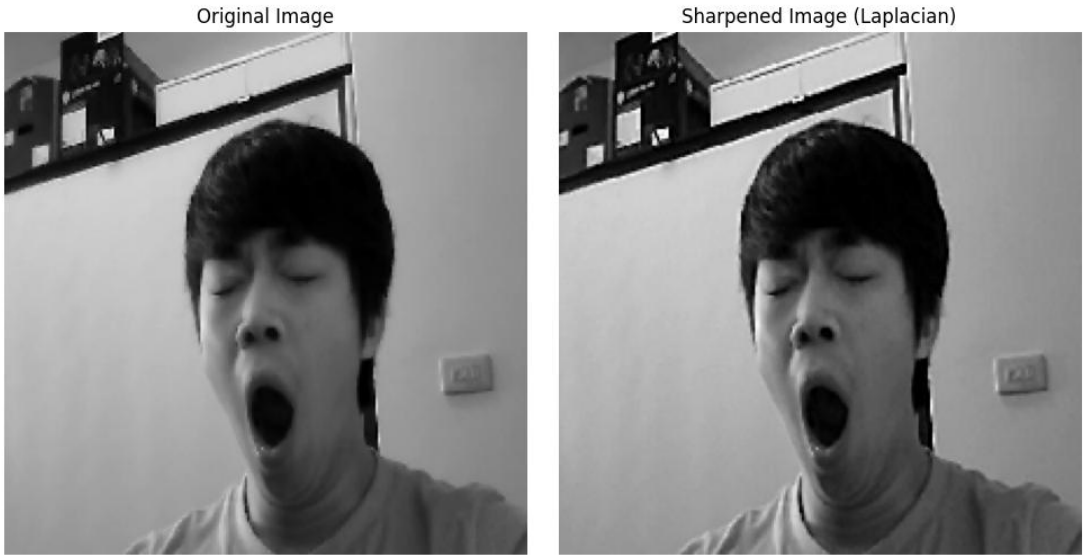


• Laplacian

1. Apply Laplacian

این فیلتر ابتدا یک تصویر RGB را دریافت کرده، آن‌را به فرمت Grayscale تبدیل می‌کند. سپس فیلتر گاوسین را به روی آن اعمال می‌کند تا نویز تصویر را کاهش دهد و در نهایت لاپلاسین تصویر را محاسبه می‌کند تا آن را تیز کند. پس از تمامی این اقدامات تصویر به حالت RGB باز می‌گردد.

2. Displaying a sample



3. Classification Report

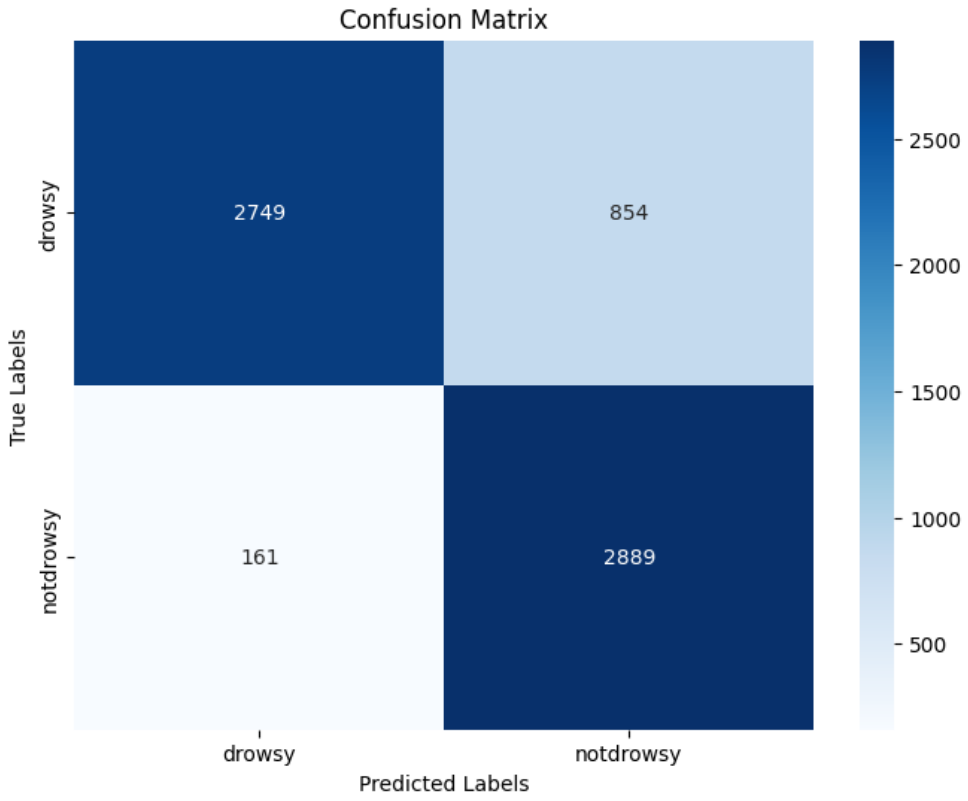
Classification Report

	Precision	Recall	F1-Score	Support
drowsy	0.94	0.76	0.84	3603
notdrowsy	0.77	0.95	0.85	3050
accuracy			0.85	6653
macro average	0.86	0.86	0.85	6653
weighted average	0.87	0.85	0.85	6653

F1-Score .4

F1-Score: 0.850581480936258

Confusion Matrix .5

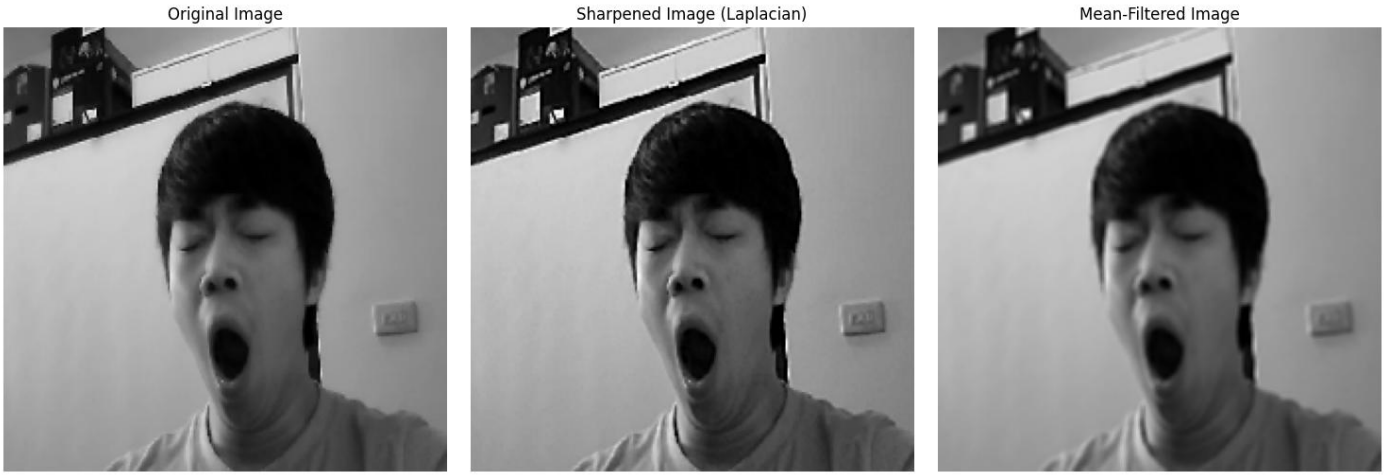


• Mean on Laplacian

1. Apply Mean on Laplacian

این نتیجه حاصل از اعمال فیلتر میانگین به روی نتیجه فیلتر لاپلاس که در قسمت قبل آمده است می‌باشد.

2. Displaying a sample



3. Classification Report

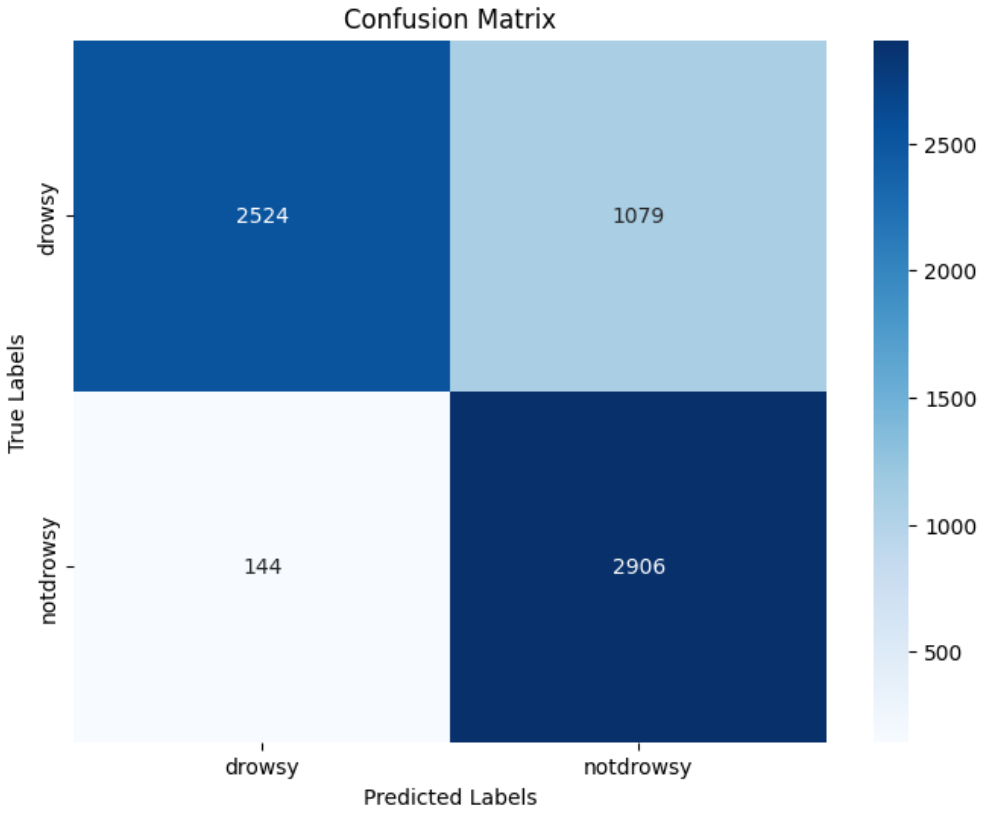
Classification Report

	Precision	Recall	F1-Score	Support
drowsy	0.95	0.70	0.80	3603
notdrowsy	0.73	0.95	0.83	3050
accuracy			0.82	6653
macro average	0.84	0.83	0.82	6653
weighted average	0.85	0.82	0.81	6653

4. F1-Score

F1-Score: 0.8261549395877754

Confusion Matrix .5

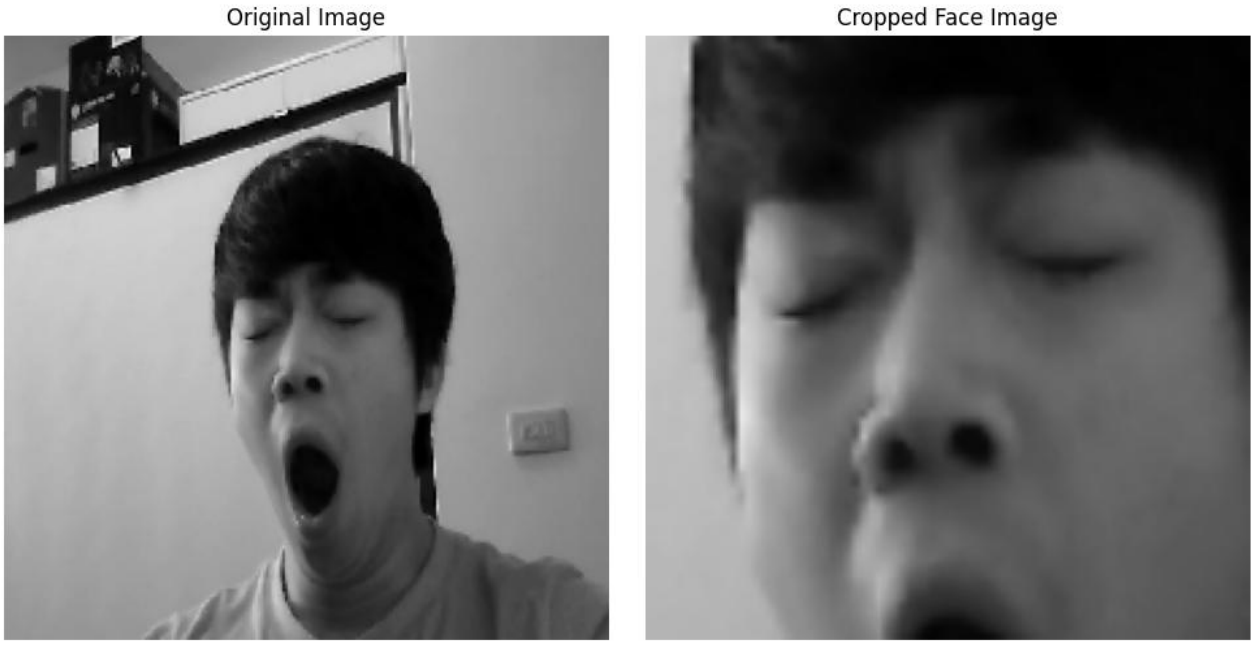


• Face Recognition

1. Apply Face Recognition

این عملیات، تصویر RGB را دریافت کرده و آن را به فرمت Grayscale تبدیل می‌کند. سپس با استفاده از Haar Cascade صورت را شناسایی و crop می‌کند که روشی مبتنی بر ماشین لرنینگ برای شناسایی اجسام می‌باشد. اگر هیچ صورتی شناسایی نشود تابع تصویر اصلی را باز می‌گرداند.

2. Displaying a sample



3. Classification Report

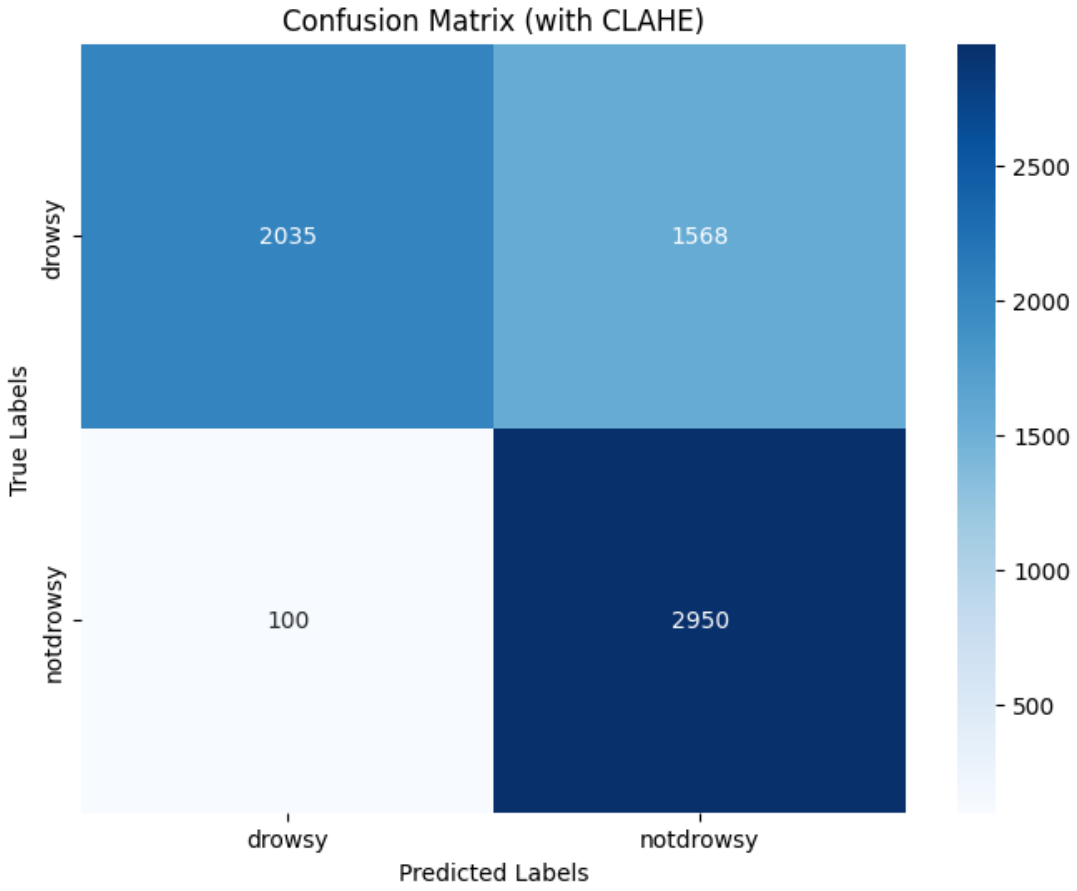
Classification Report

	Precision	Recall	F1-Score	Support
drowsy	0.74	0.77	0.75	3603
notdrowsy	0.71	0.67	0.69	3050
accuracy			0.72	6653
macro average	0.72	0.72	0.72	6653
weighted average	0.72	0.72	0.72	6653

F1-Score .4

F1-Score: 0.6912333838128891

Confusion Matrix .5

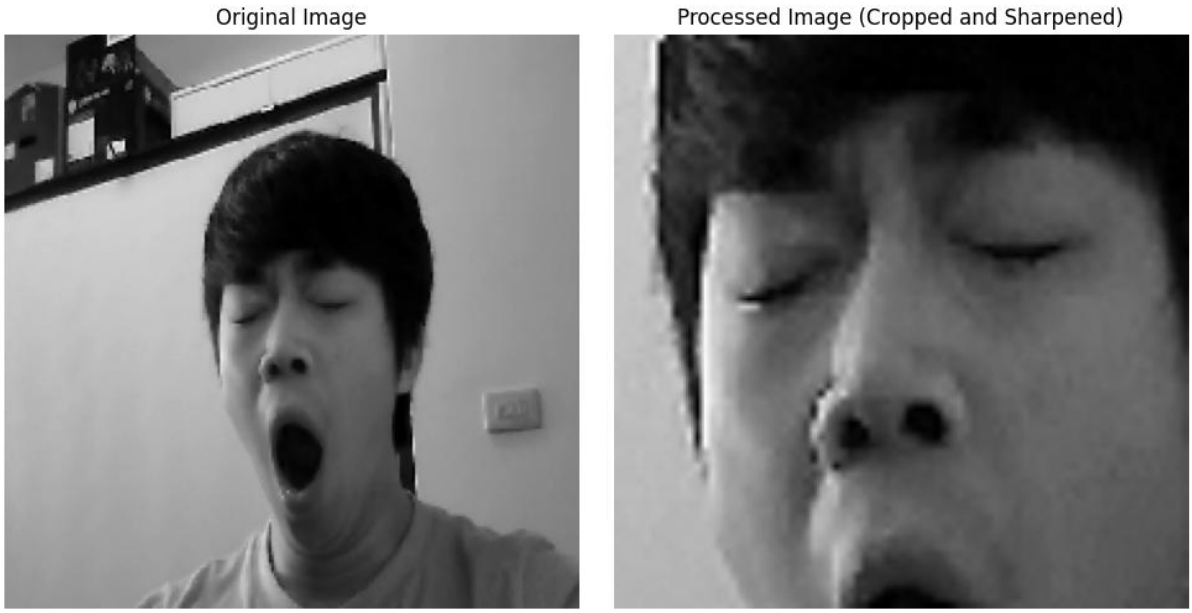


• Face Recognition with Laplacian

1. Apply Face Recognition with Laplacian

حاصل اعمال فیلتر لاپلاسین بر نتایج Face Recognition می‌باشد.

2. Displaying a sample



3. Classification Report

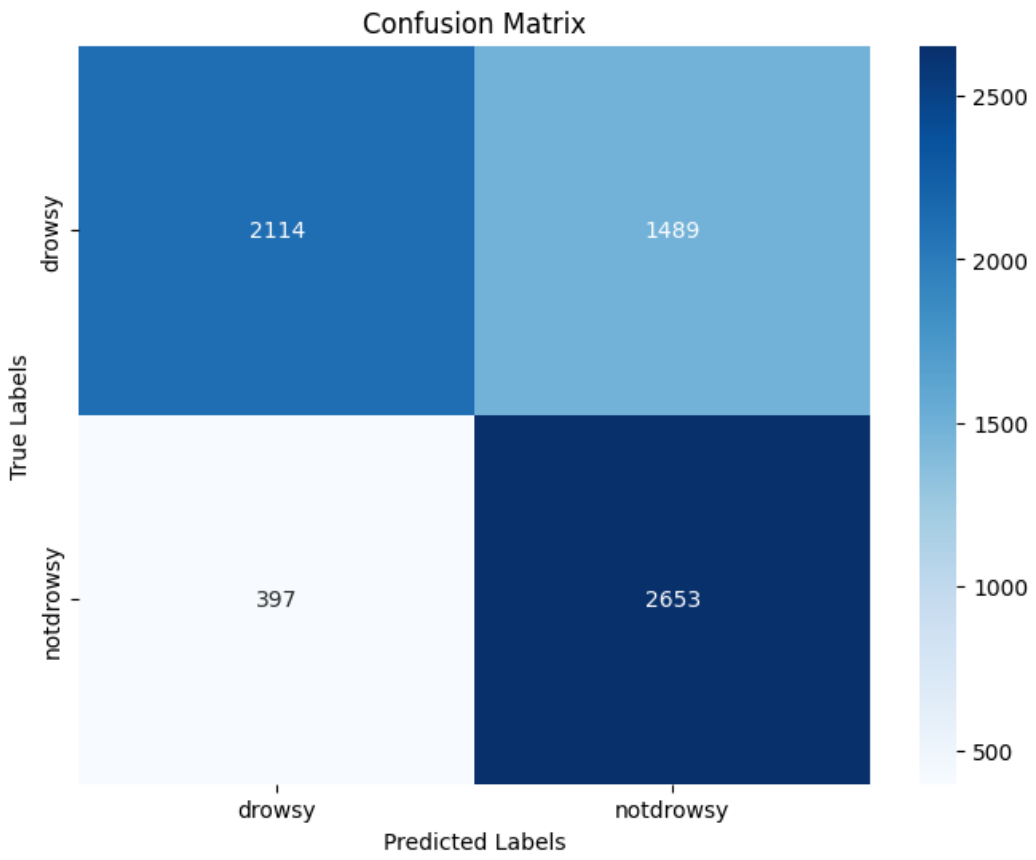
Classification Report

	Precision	Recall	F1-Score	Support
drowsy	0.84	0.59	0.69	3603
notdrowsy	0.64	0.87	0.74	3050
accuracy			0.72	6653
macro average	0.74	0.73	0.71	6653
weighted average	0.75	0.72	0.71	6653

F1-Score .4

F1-Score: 0.73776418242491 66

Confusion Matrix .5

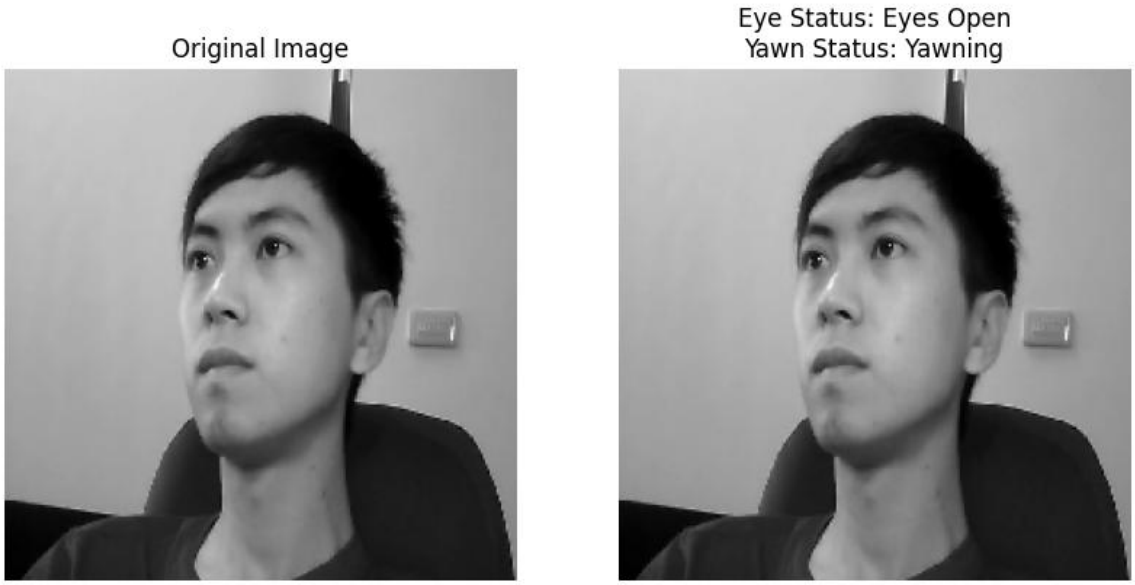


• Eye and yawn detection

1. Apply Eye and yawn detection

این عملیات با استفاده از Dlib's face detector المان‌های صورت را استخراج کرده و EAR و MAR را محاسبه می‌کند. بر اساس این زوایا تصمیم می‌گیرد که چشم‌ها باز یا بسته و یا فرد در حال خمیازه کشیدن است.

2. Displaying a sample



3. Classification Report

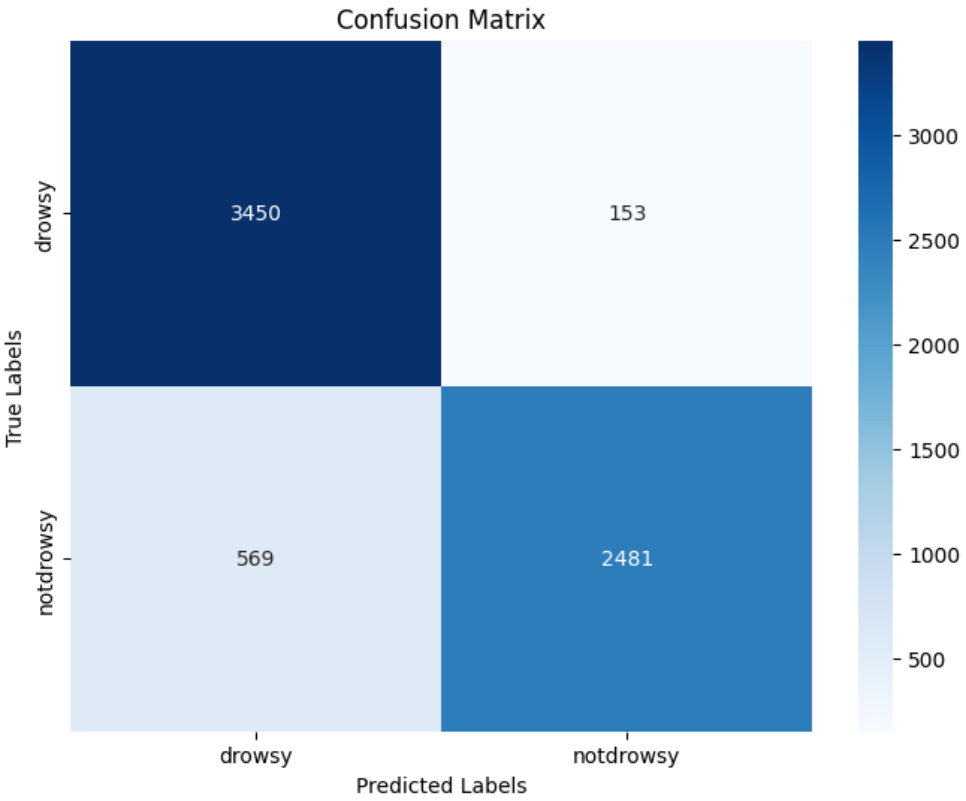
Classification Report

	Precision	Recall	F1-Score	Support
drowsy	0.86	0.96	0.91	3603
notdrowsy	0.94	0.81	0.87	3050
accuracy			0.89	6653
macro average	0.90	0.89	0.89	6653
weighted average	0.90	0.89	0.89	6653

F1-Score .3

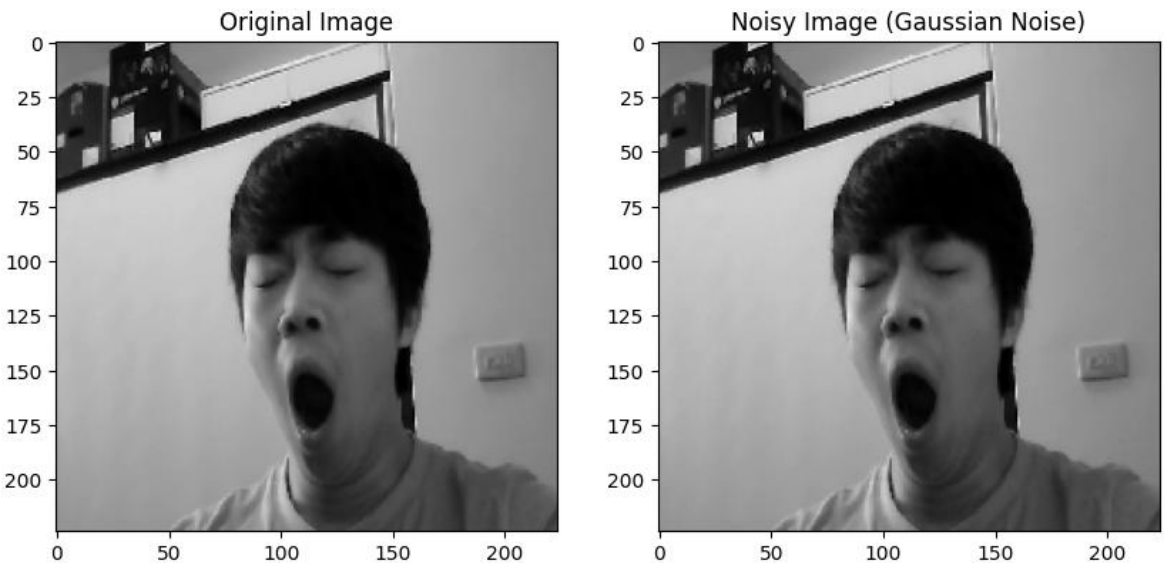
F1-Score: 0.8729767769176636

Confusion Matrix .4



Gaussian Noise •

Displaying a sample .1



Classification Report .2

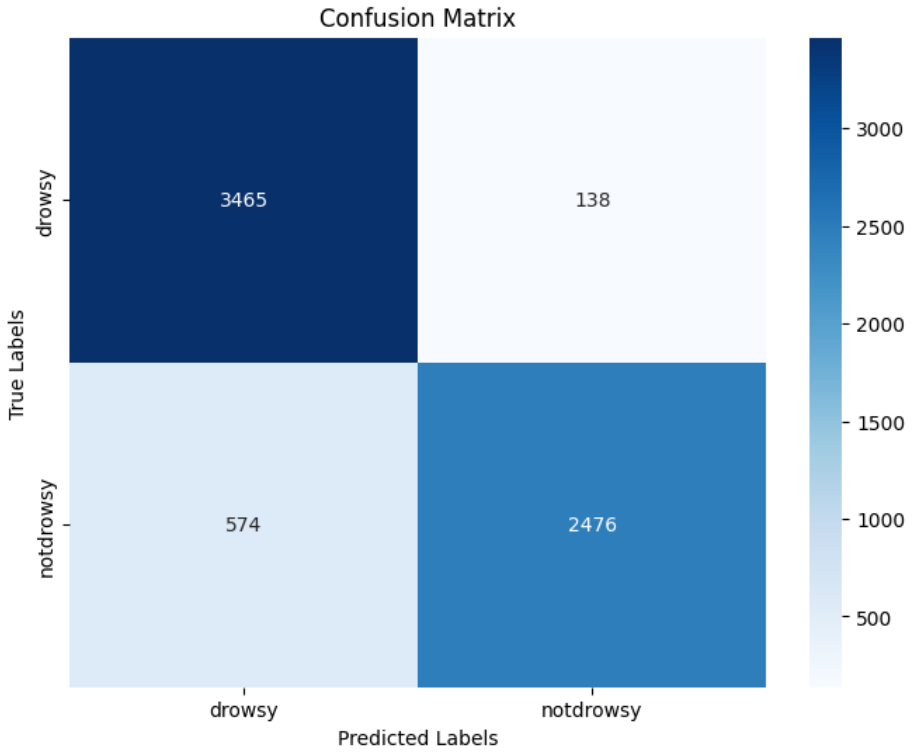
Classification Report

	Precision	Recall	F1-Score	Support
drowsy	0.86	0.96	0.91	3603
notdrowsy	0.95	0.81	0.87	3050
accuracy			0.89	6653
macro average	0.90	0.89	0.89	6653
weighted average	0.90	0.89	0.89	6653

F1-Score .3

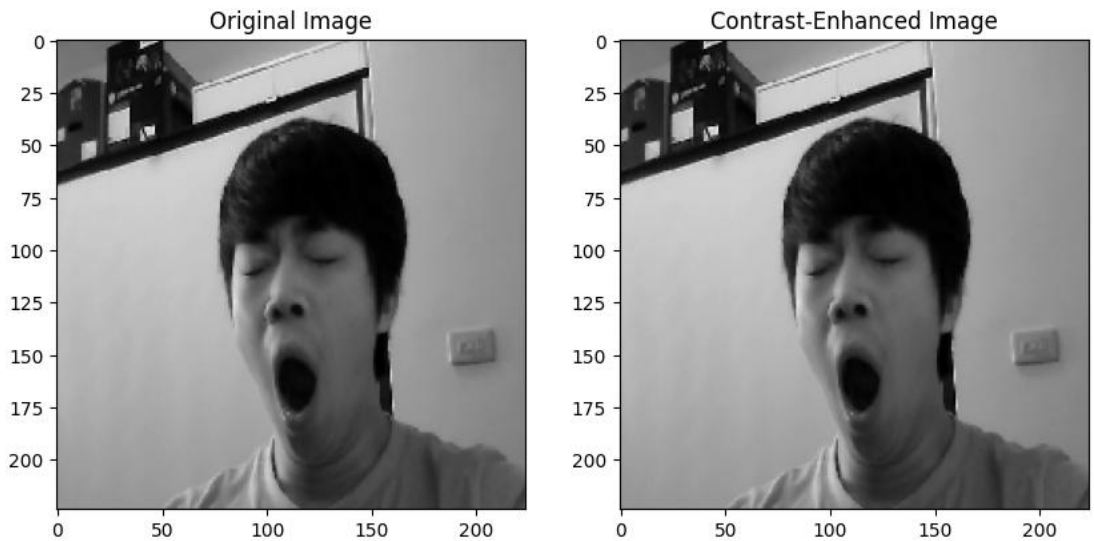
F1-Score: 0.8742937853107344

Confusion Matrix .4



Linear Contrast Enhancement •

Displaying a sample .1



Classification Report .2

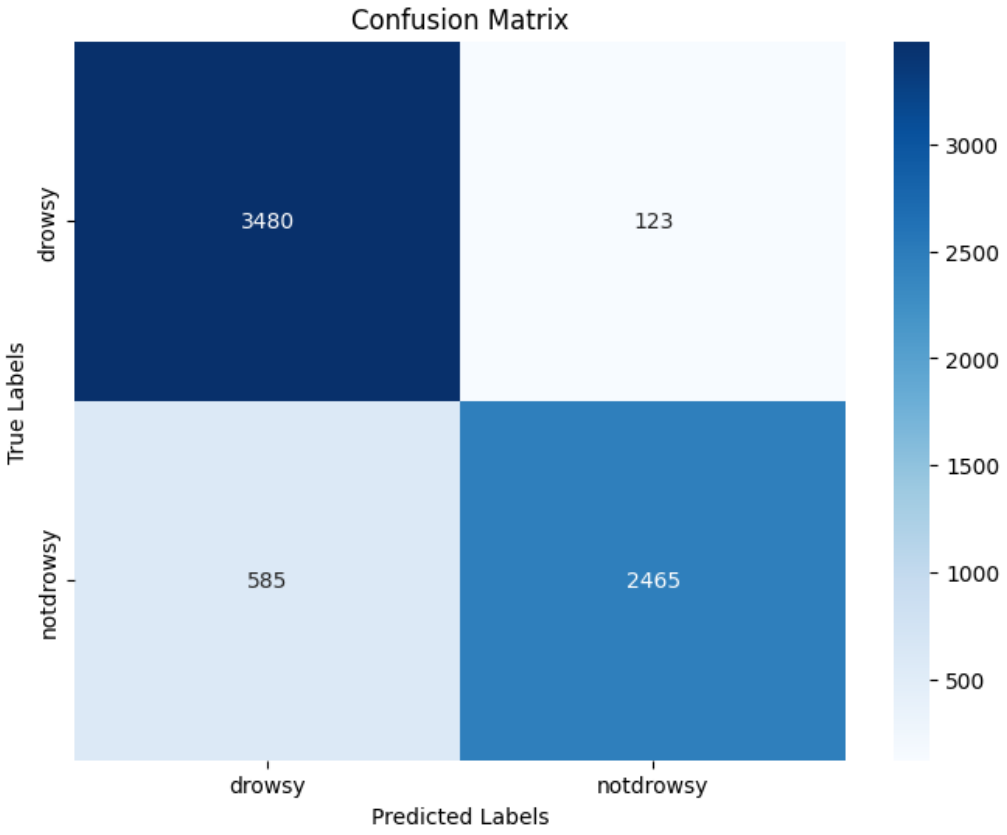
Classification Report

	Precision	Recall	F1-Score	Support
drowsy	0.86	0.96	0.91	3603
notdrowsy	0.95	0.81	0.87	3050
accuracy			0.89	6653
macro average	0.90	0.89	0.89	6653
weighted average	0.90	0.89	0.89	6653

F1-Score .3

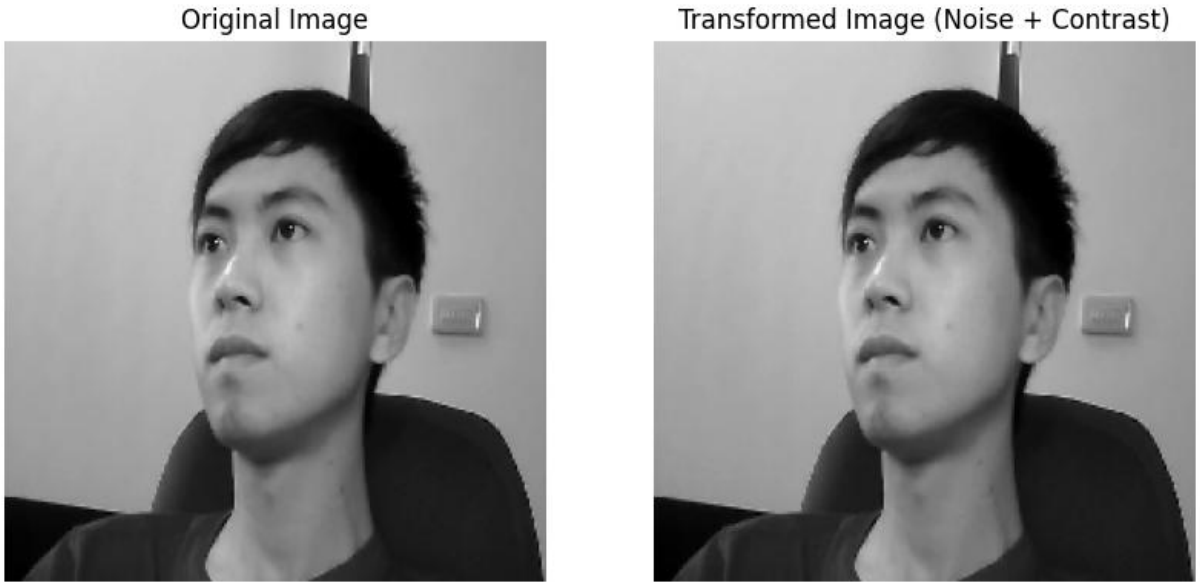
F1-Score: 0.8744235544519333

Confusion Matrix .4



Gaussian Noise on Linear Contrast Enhancement •

Displaying a sample .1

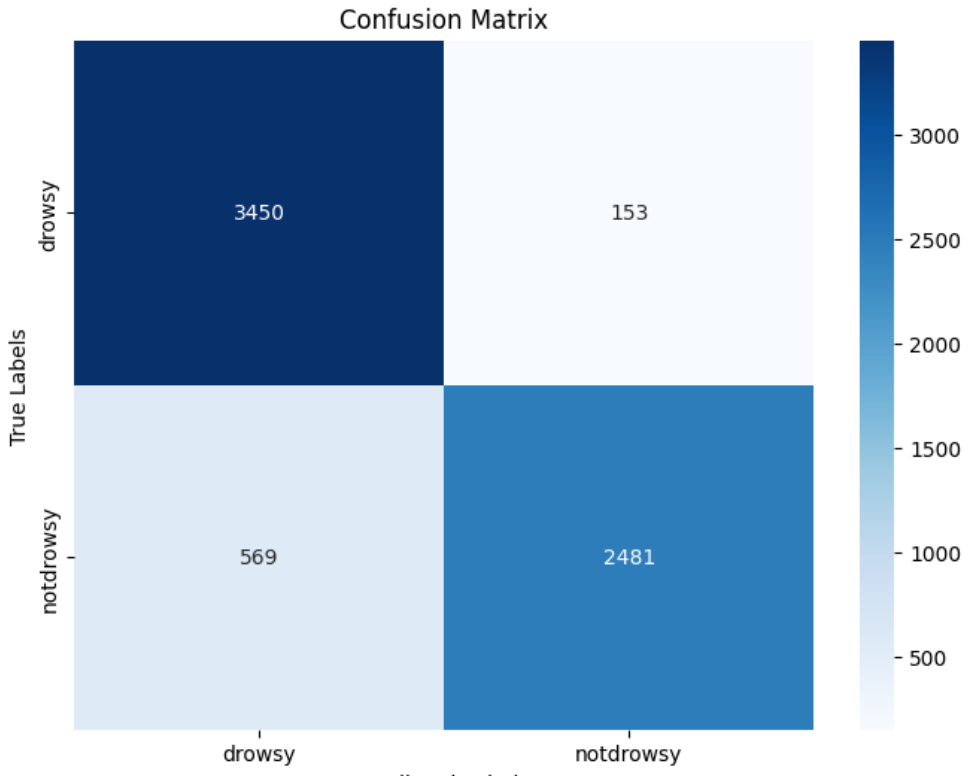


Classification Report .2

Classification Report

	Precision	Recall	F1-Score	Support
drowsy	0.86	0.96	0.91	3603
notdrowsy	0.94	0.81	0.87	3050
accuracy			0.89	6653
macro average	0.90	0.89	0.89	6653
weighted average	0.90	0.89	0.89	6653

F1-Score .3

F1-Score: 0.8729767769176636

Confusion Matrix .4