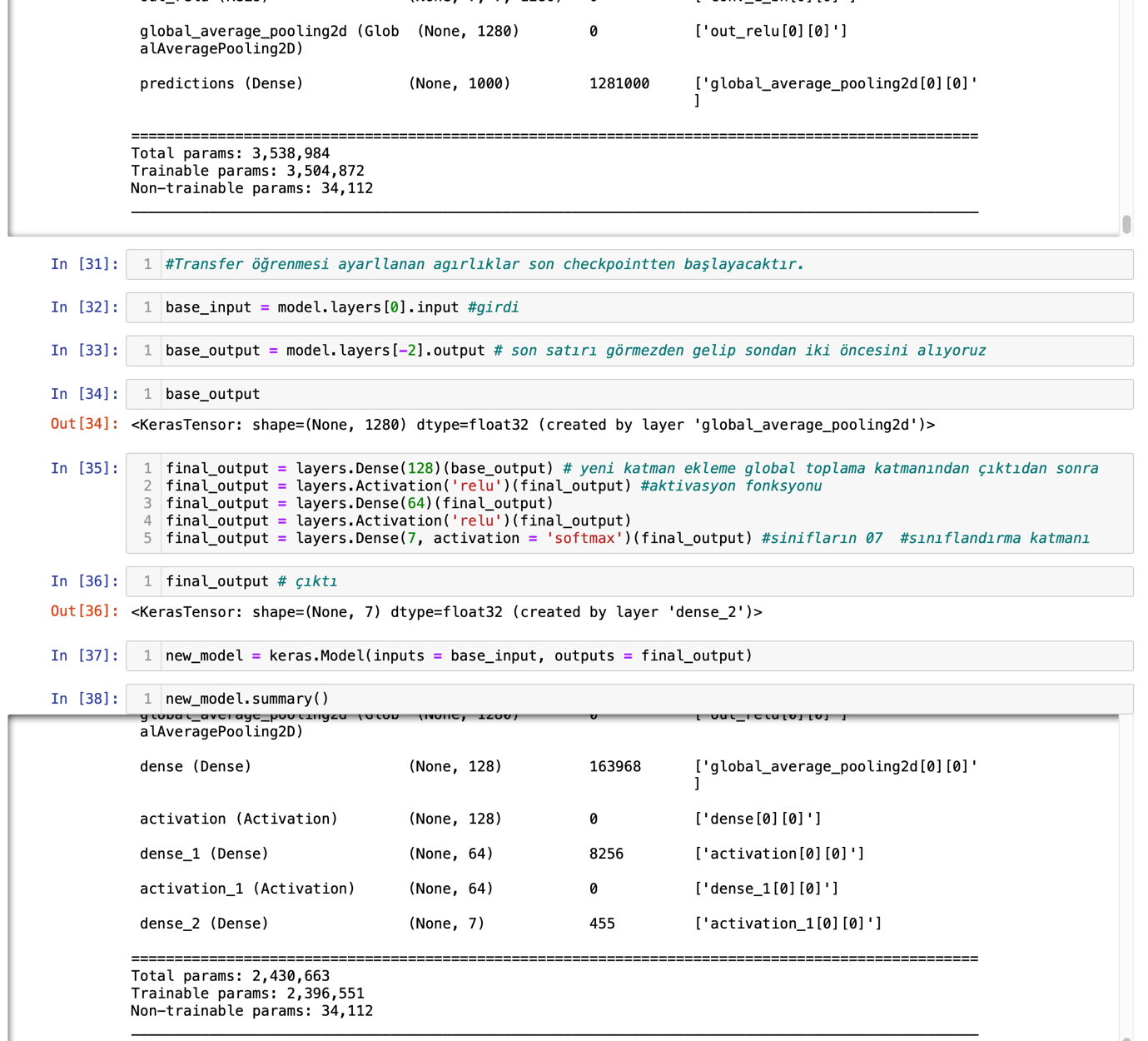
Yüz tanıma projesi

Programı yaparken kullanılan OpenCV, TensorFlow, Mathplotlib ve Numpy kütüphanelerden oldukça faydalandım ve kodumun en tepesine ekledim. Ardından eğitim setinden bir görüntü ekleyerek çözünürlerini görüntüledim. Ardından eğitim dataseti veri yolunu ve farklı duyguların barındırıldığı klasörleri Siniflar olarak ekledim. Sonrasında eğitim setinde kullanılacak resimlerin boyutlarını 224x224 olacak sekilde değiştirip görüntüledim.



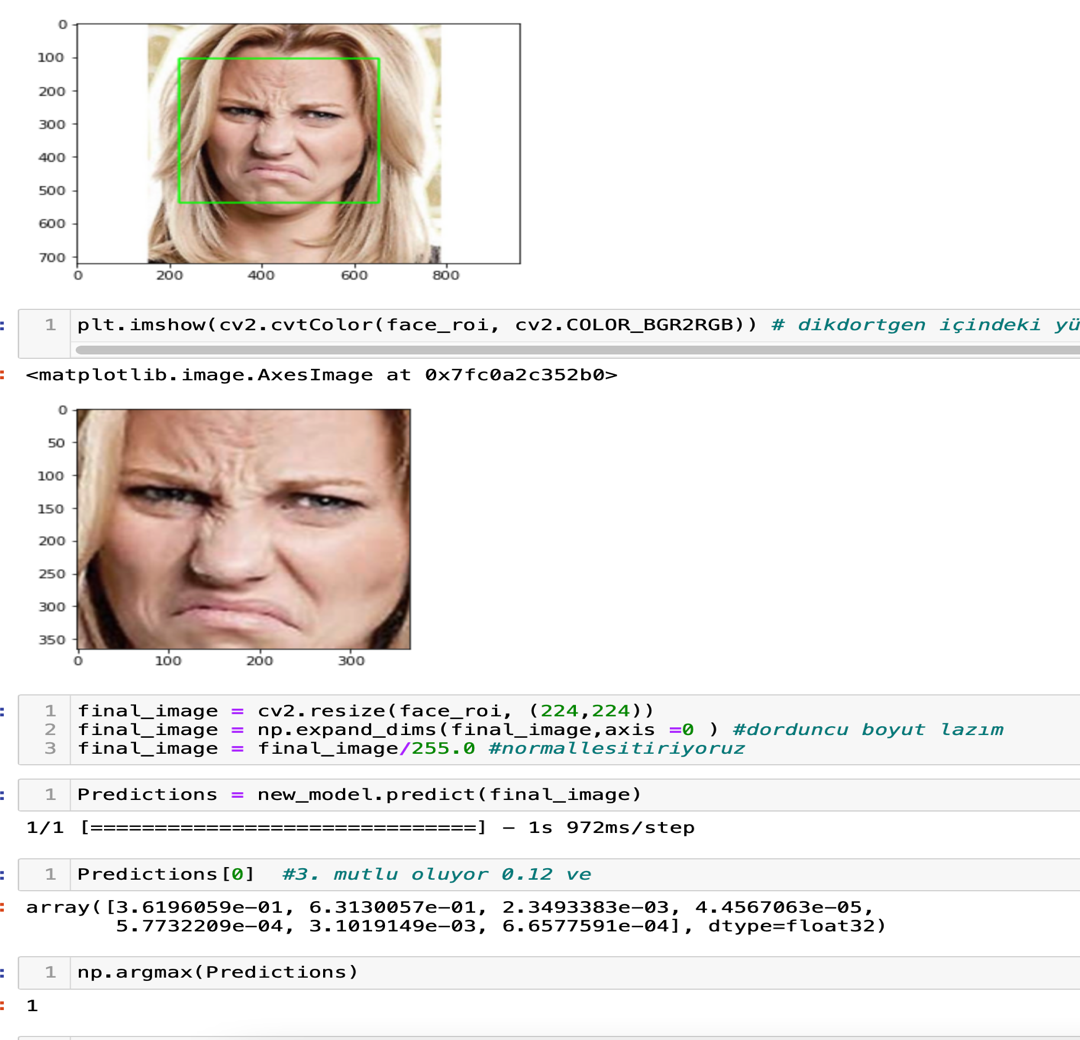
**Şekil.3.1** Eğitim datasei veriyolu klasör ve siniflar ve yeniden boyutlandırma

Bir sonraki adımda ise create \_training\_Data() adında bir fonksyon oluşturup bunların sınıflarına etiketler atıyoruz dosya yolunu takip edip openCV kutuphanesini kullanarak boyutlarını yeniden belirliyoruz.Bu işlemde veri setindeki tüm resimleri işleyecek bir şekilde ayarlıyoruz. Veri seti fonksyonunu oluşturduktan sonra random import ederek kamera örneği için eğitiyoruz. Ardından X ve y eksenlerine veri etiketleri özelllikleri eklenecek şekilde scikit öğrenmesi yapacak şekilde bir for döndüğüsü ve matematiksel işlemlere okuyoruz. Bu ayarlamaları bitirdikten sonra eğitim için derin öğrenme ve transfer ögrenmesi işlemlerini gerçekleştiriyoruz. Devamında, TensorFlow ve Keras kütüphanelerini içeriye aktardıktan sonra ön eğitim modelini seçiyoruz. Ön eğitim özetinden ne kadar parametre olduğuna eğitilebilir ve eğitillimez parametreleri görüntülüyoruz ve versiyon kontrolu yapıyoruz. Transfer öğrenmesi için ayarlanmış checkpointleri girdi ve son satırları/çıktıları model summaryi göre kodluyouz . Yeni model özetinde yeni parametre değerlerini görüntülüyoruz. 

**Şekil.3.2** eski/yeni Parametreler, transfer öğrenmesi işlemi ve yeni model özeti

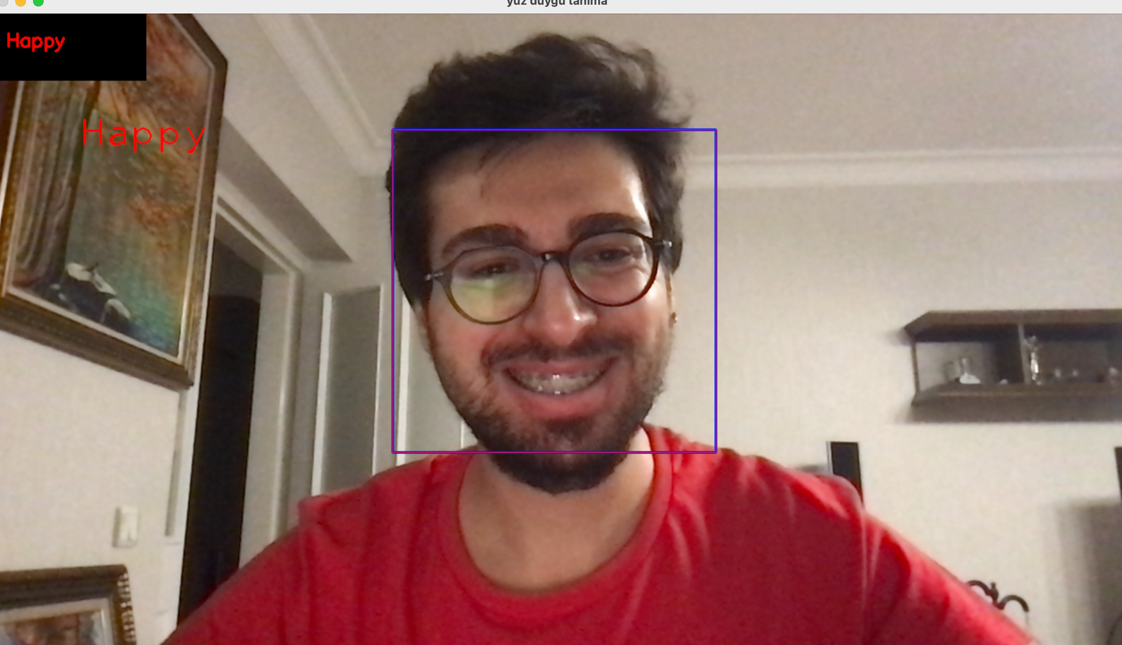
Dahası, 15 epockhlu modelini alıyoruz ve sonrasında bu yeni modeli kaydediyoruz. Bu uzun işlem bittikten sonra final adımına gelmeden önce veri setindeki gri resimleri renkledirme işlemi yapıyoruz. Dosya yolundan bir stock resim almamızın ardından yüzü algılayacağı yeşil kutuyu belirliyoruz tekrar boyutları belirlenen resize ediyoruz ve normalleştiriyoruz . En sonunda , resimi eğittiğimiz modele göre tahmine sokuyoruz bu tahminlerde 0 dan 6 ya kadar bir değer çıkartyor . Bu değer daha önce sınıflarda belirlediğimiz veri setlerinin klasorlerin adlarıyla ayrı her farklı klasörde farklı duygu barındıran resimler olduğunu belirtmiştik.

* 0 = Sinirli
* 1 = Tiksinme
* 2 = Korku
* 3 = Mutlu
* 4 = Nötr
* 5 = Hüzünlü
* 6 = Şaşırmış

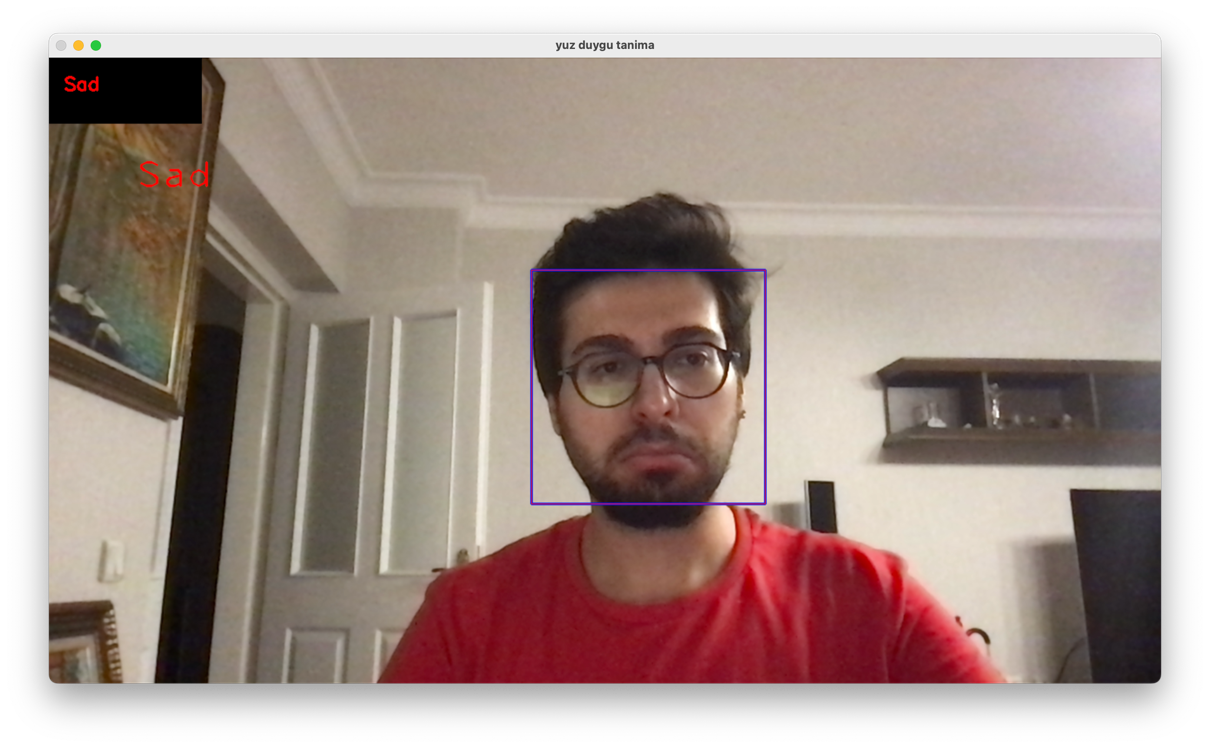


**Şekil3.3** Yapılan tahmin sonucunu 1 yani tiksinme olarak yapılan tahmin

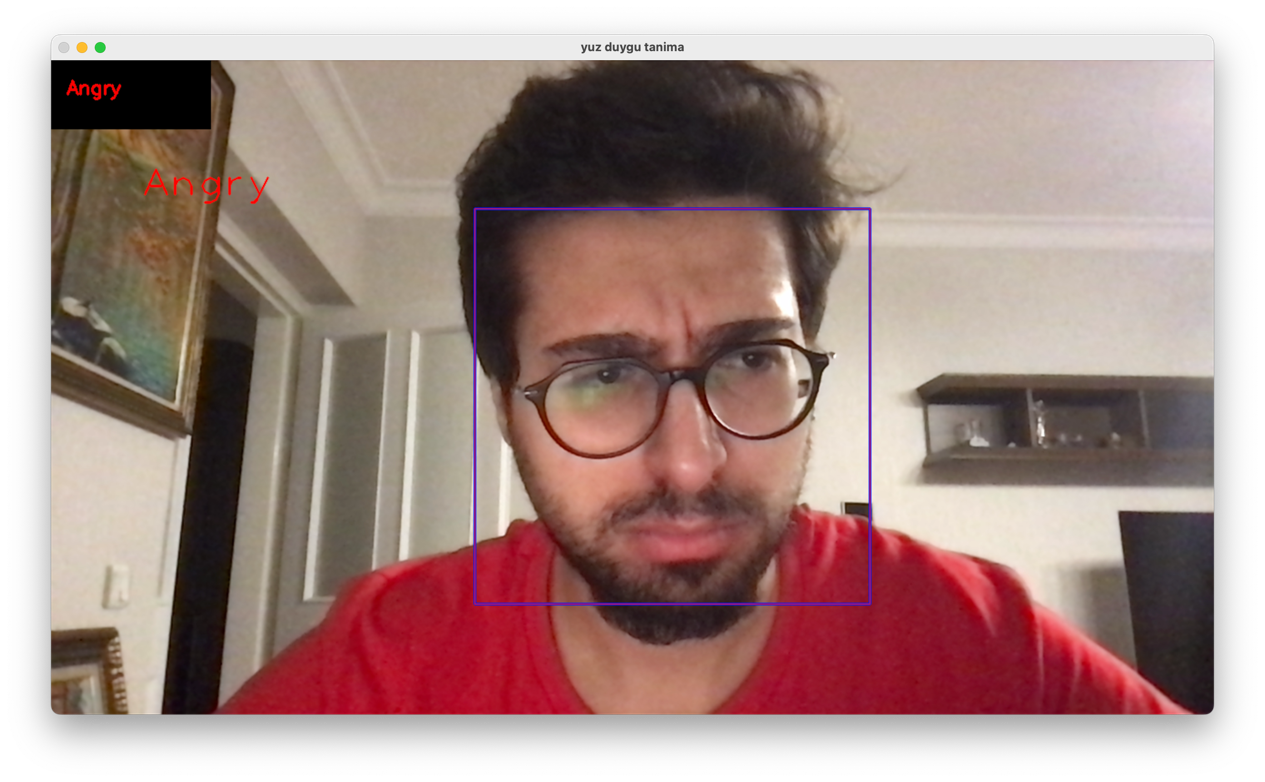
Final olarak, eş zamanlı webcamden görüntü alıp bu tahminleri canlı değerlendirilebilmesi için gereken kodlamalar yazılmış oldu. Genellikle burda openCv kütüphanesinin özelliklerinden çoğunlukta tarafımca fayda sağlanmış oldu. Arka plan yazılarını ve renklerini başlangıç fonksyonlarını oluşturduktan sonra haarscascade ön yüz algoritmasını kullandım. Bundan sonra çerçeve içindeki çok yüzleri tanımlama kodunu yazdım. Tahminler model ile Predictions olarak kullanarak her duygu için ayrı if döngüsü oluşturdum. Aşağıdaki ekran görüntülerinde farklı duyguların anlık tahmini görünmektedir.



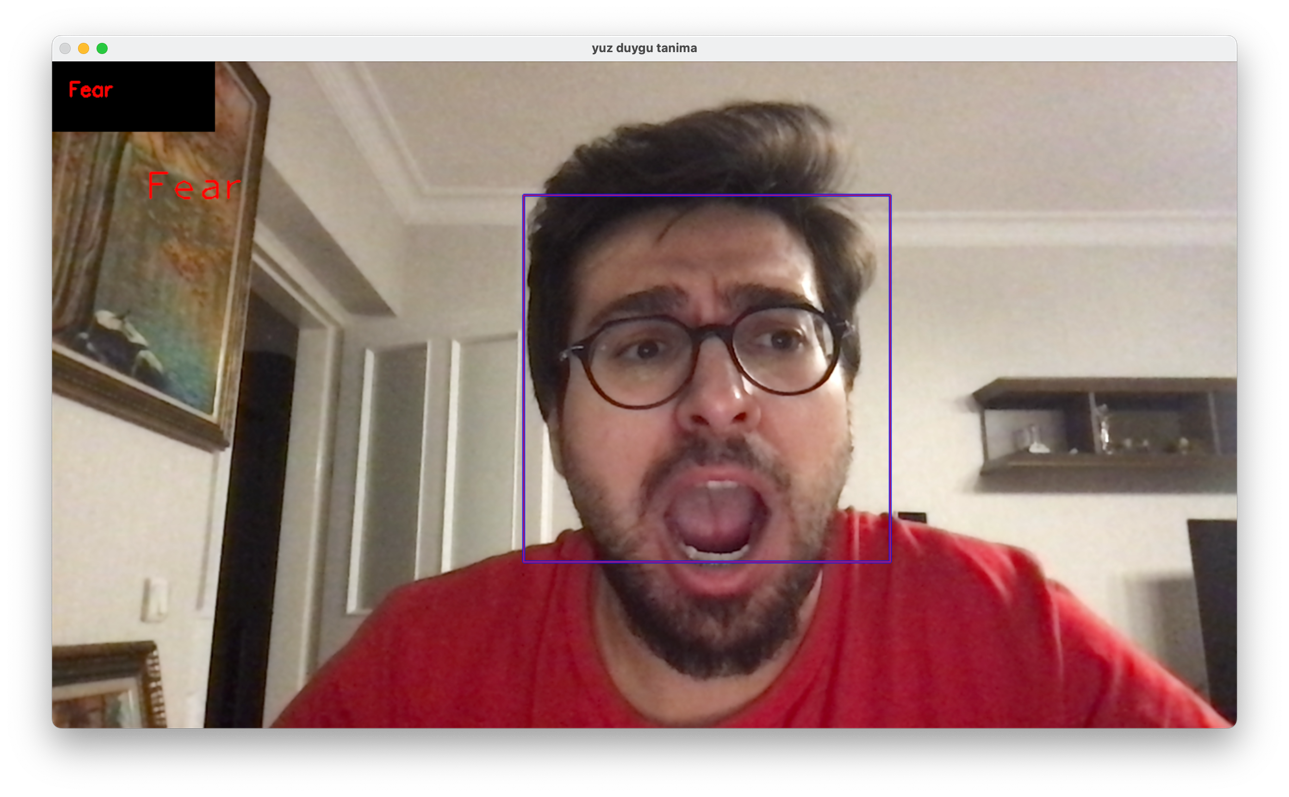
**Şekil.3.4** Mutlu ifade tahmini



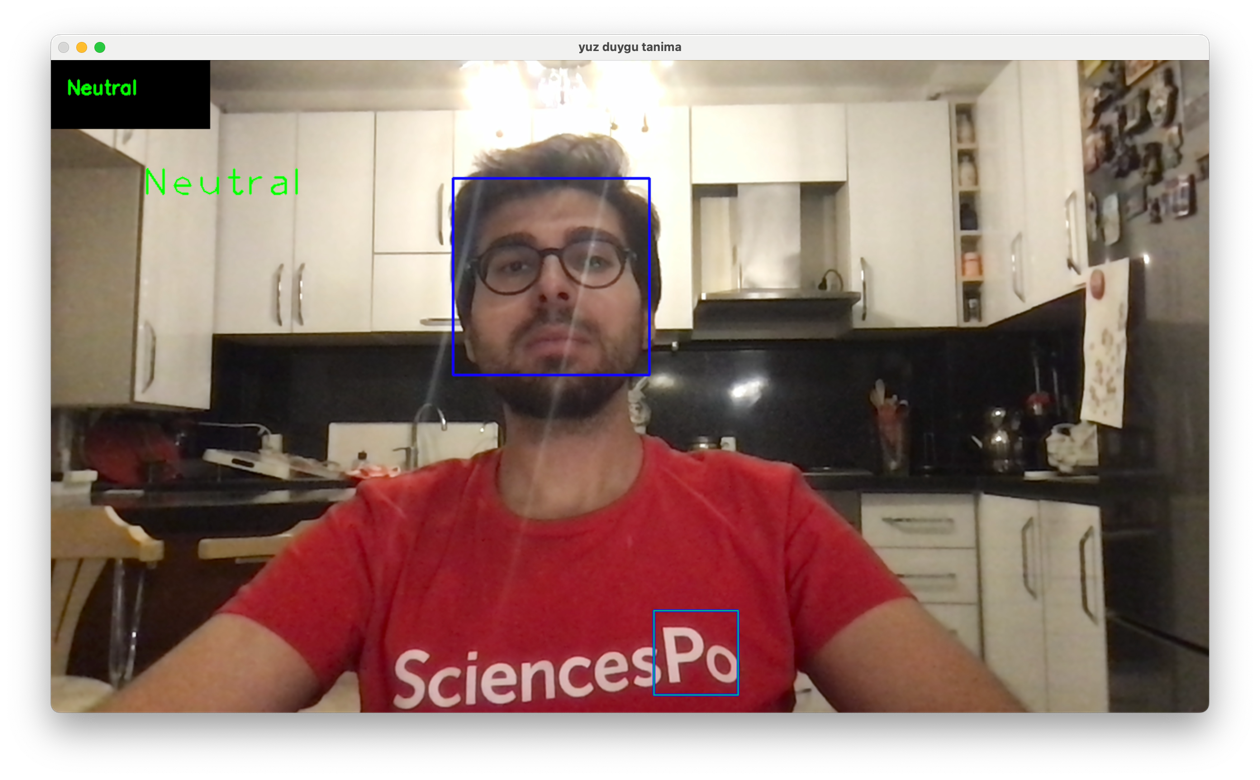
**Şekil.3.5** Üzgün ifade tahmini



**Şekil.3.6** Sinirli ifade tahmini



**Şekil.3.7** Korku ifade tahmini



**Şekil.3.8** Nötr ifade tahmini

3.1 DENEMELER

Belirlenen ifade biçimlerini birçok kez deneyim üç farklı skala oluşturdum bunlar başlıca Başarısız Başarılı ve Kısmen Başarılı şekilde bize verilen çıktılara göre bir deneme tablosu oluşturdum. Hem kamera hem de ortam ışık ekran kartı faktörlerinden dolayı değerler farklılık gösterebileceği için böyle bir deneye başvurdum.

**Tablo 3.1.1** Başarı Durumu Tablosu

|  |  |
| --- | --- |
| **İfade Biçimi** | **Başarı Durumu** |
| Sinirli | Başarılı |
| Tiksinme | Kısmen Başarılı |
| Korku | Başarılı |
| Mutlu | Başarılı |
| Nötr | Başarılı |
| Hüzünlü | Başarılı |
| Şaşırmış | Kısmen Başarılı |

4.KAPANIŞ VE ÖNERİLER

Bilgisayar görüşü günümüzde ekstra önem kazandığı ek olarak da bir çok alanda ve sektörde kullanılabilen yüz duygu tahmini programı yapay zeka alanında bilgisayar bilime büyük bir basamak atlatabileceği yüksek ihtimaldir. Bu güne kadar hem devletlerin hem de bireylerin mobil cihazlarında veya büyük verilerin süper bilgisayarlar tarafından işlenerek güvenlik kameralarından insanların duygu durumunu anlamak hayal gözükmediği yadsınamaz bir gerçektir. Gelecekten ışık tutan bu program modülü bir çok farklı birimle ve yapıyla birleşme yapısından dolayı geriye kalan tek şey insanın hayal gücü ve isteklerine göre şekillenebilir. Mesela güvenlik veya bir mobil uygulamada eğlence filtresi bu proje temelinde bilgisayar biliminin yapay zeka alanında kendine eşsiz bir yer alır.

5.KAYNAKÇA

[1] “B. Venners - The Making of Python,” www.artima.com. 13 Jan 2003.https://www.artima.com/articles/the-making-of-python

[2] M. Abadi et al., “TensorFlow: Large-scale machine learning on heterogeneous distributed systems,” Preliminary White Paper [cs.DC], 2016. <http://download.tensorflow.org/paper/whitepaper2015.pdff>

[3] Quotes about Python,” Python.org. [Online]. Available: https://www.python.org/about/quotes/. [Accessed: 27-May-2022].

[4] ["NumFOCUS Sponsored Projects"](https://numfocus.org/sponsored-projects). NumFOCUS. Retrieved 2021-10-25.

[5] A. Kaehler and G. Bradski, *Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library*. “O’Reilly Media, Inc.,” 2016. Accessed: May 23, 2022. [Online]. Available:<https://books.google.com.tr/books?id=SKy3DQAAQBAJ&pg=PT26&redir_esc=y#v=onepage&q&f=false>

[6]TensorFlow: Open source machine learning. Google. 2015. Archived from the original on November 11, 2021.

[7] C. R. Harris *et al.*, “Array programming with NumPy,” *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020, doi: 10.1038/s41586-020-2649-2.