

TOBB ETÜ Bilgisayar Mühendisliği Bölümü
BİL 468 Dönem Projesi
2023-24 Bahar Dönemi

Proje Grup No: 7

1. Proje tanımı (kısaca özet, problemin açıklanması, girdiler, çıktılar, sınıflar)

Bu proje, bilgisayar görü teknolojilerini kullanarak yüz ifadelerini tanıyarak duygusal durumu analiz etmeyi amaçlar. Temel amaç, girdi olarak alınan görsellerdeki insanların yüz ifadelerinden duygusal durumlarını anlamak ve buna uygun çıktı verebilmektir. Proje kapsamında geliştirilecek model, görüntüdeki insanın yüz ifadesinden mutlu, üzgün, sinirli, şaşkın, normal, iğrenmiş, korkmuş olmak üzere 7 farklı duygudan hangisini ifade ettiğini saptayacaktır.

Geliştirilecek modelin girdileri yakın çekimden yüzün tamamını kaplayan görüntüler olacaktır. Sınıflarımız yukarıda bahsettiğimiz 7 farklı duygu durumu olacaktır. Çıktılar ise görüntülerdeki yüzün bu 7 sınıftan hangisine sahip olduğunun bir

2. Literatür taraması (daha önce benzer projelerde kullanılan metotlar hakkında bilgiler) (kullanılan başlıca yöntemlerin açıklanması, karşılaştırılması ve değerlendirilmesi)

Yüz ifadesi tanıma ve duygusal analiz konusunda çeşitli metotlar ve yaklaşımlar literatürde vardır. Bu alandaki başlıca yöntemler arasında derin öğrenme tabanlı yaklaşımlar, özellik çıkarma ve sınıflandırma yöntemleri bulunmaktadır.

Derin Öğrenme Tabanlı Yaklaşımlar: Son yıllarda derin sinir ağları, yüz ifadesi tanıma ve duygusal analizde büyük başarı elde etmiştir. Özellikle evrişimli sinir ağları (Convolutional Neural Networks - CNN) bu alanda yaygın olarak kullanılmaktadır. CNN'ler, yüz görüntülerinden özellikleri çıkararak sınıflandırma yapmak için kullanılır. Bu yöntem, genellikle end-to-end bir yaklaşımı benimser ve önceden işlenmiş özelliklerin manuel olarak belirlenmesine ihtiyaç duymaz.

Özellik Çıkarma ve Sınıflandırma Yöntemleri: Geleneksel olarak, yüz ifadesi tanıma için özellik çıkarma ve sınıflandırma yöntemleri kullanılmıştır. Bu yöntemlerde, yüz görüntülerinden belirli özellikler (örneğin, yüzün mimiksel özellikleri, gözlerin ve ağızın konumu vb.) çıkarılır ve ardından bu özellikler sınıflandırıcı algoritmalar kullanılarak sınıflandırılır. Örnek olarak, Support Vector Machines (SVM), Decision Trees ve Random Forests gibi algoritmalar kullanılabilir.

Farklı Veri Artırma Yöntemleri: Veri artırma, modelin eğitim veri setini çeşitlendirmek ve genelleştirme yeteneğini artırmak için kullanılan bir yöntemdir. Yüz ifadesi tanıma modellerinde, özellikle az veriye sahip durumlarda, veri artırma teknikleri önemli bir rol oynar. Yatay ve dikey simetri, döndürme, ışıklandırma değişiklikleri gibi teknikler veri setini çeşitlendirmek için yaygın olarak kullanılır.

Bu literatür taraması, projede kullanılacak yöntemlerin belirlenmesinde rehberlik sağlayacaktır. Yapılacak olan yöntem seçimi, kullanılabilir veri miktarı, hesaplama gücü, model performansı ve uygulama gereksinimlerine bağlı olarak yapılacaktır.

3. Projede kullanılan veriler (verilerin boyut, sayı vb. detayları, etiketler, birkaç veri örneği)

1. <https://www.kaggle.com/datasets/msambare/fer2013>

FER-2013 veri seti, yüzleri duygusal ifadelerine dayalı olarak yedi kategoriye (0=Kızgın, 1=İğrenmiş, 2=Korku, 3=Mutlu, 4=Üzgün, 5=Şaşkın, 6=Normal) sınıflandırmak amacıyla kullanılan gri tonlamalı görüntülerden oluşmaktadır.

Yüzler otomatik olarak kaydedilmiştir, böylece yüz neredeyse merkezlenmiş ve her görüntüde yaklaşık olarak aynı miktarda alanı kaplamaktadır.

Eğitim seti, 28,709 örneği içermekte olup, halka açık test seti ise 3,589 örneği içermektedir

Aşağıda veri setinden örnekler görebilirsiniz.



Korku



Üzgün



Şaşkın

2. <https://www.kaggle.com/datasets/jonathanoheix/face-expression-recognition-dataset/data>

Aynı şekilde bu veri seti de yedi katogoriye (0=Kızgın, 1=İğrenmiş, 2=Korku, 3=Mutlu, 4=Üzgün, 5=Şaşkın, 6=Normal) sınıflandırmak amacıyla kullanılan gri tonlamalı görüntülerden oluşmaktadır.

Veri seti, 28,821 eğitim ve 7,066 test örneğinden oluşmaktadır.

Bu iki veri setinin aynı duygusal sınıflandırmalara sahip olması, ayrıca bir avantajdır. Bu, modelin genel duygusal ifadeleri daha iyi anlamasına ve doğru sınıflandırma yapmasına yardımcı olabilir.

Aşağıda veri setinden örnekler bir kaç örnek bulabilirsiniz.



İğrenmiş



Kızgın



Doğal

4. Projede kullanılacak yöntemlerin (en az 3 tane) detaylı açıklaması

Yöntem 1:

Yöntem 2:

Yöntemin Genel Açıklaması ve Önemli Noktaları: Bu metodoloji, yüz ifadelerini sınıflandırmak için kullanılan bir model geliştirmeyi amaçlar. Temel olarak, Histogram of Oriented Gradients (HOG) tekniği kullanılarak özellik çıkarımı yapılır. Bu, görüntülerdeki lokal gradyan yönergelerini vurgular, böylece yüz ifadelerini temsil eden önemli özellikler elde edilebilir. Daha sonra, bu HOG özellikleri SVM, KNN ve RF kullanılarak sınıflandırılır.

Genel Aşamalar:

a. Veri Hazırlığı: İlk olarak, yüz ifadesi veri setinin üzerinde çalışılır. Veri seti, eğitim ve test verileri olarak ayrılır.

b. Özellik Çıkarımı: Her bir görüntü, HOG tekniği kullanılarak özelliklere dönüştürülür. Bu, görüntülerdeki gradyanların histogramlarını oluşturarak, yüz ifadelerini temsil eden önemli özellikleri çıkarır.

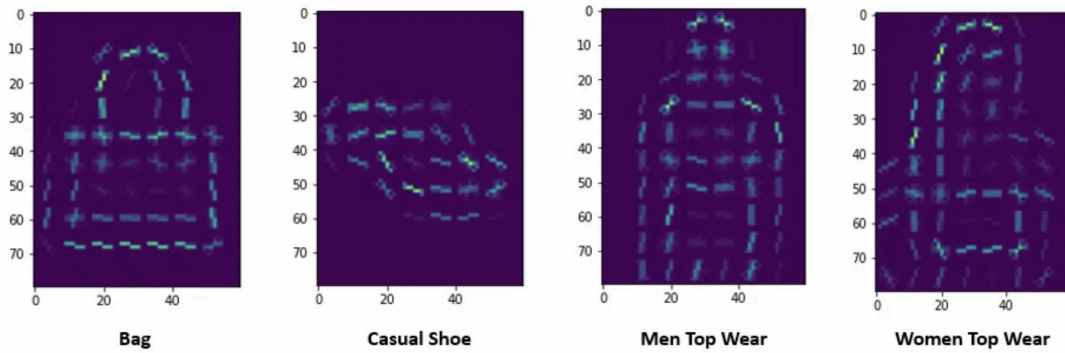
c. Model Eğitimi: Yukarıda bahsedilen 3 farklı model SVM, KNN VE RF kullanılarak, HOG özellikleriyle eğitim verileri üzerinde bir sınıflandırma modeli oluşturulur. Bu model, yüz ifadelerini belirlemek için öğrenilir.

d. Performans Metrikleri: Test verileri kullanılarak modellerin performansları değerlendirilir. F1, precision, accuracy gibi metrikler hesaplanarak kaydedilir.

HOG Tekniği Aşamaları Detayları:

1. Görüntü Boyutlandırma ve Normalizasyon: Görüntülerin boyutlarını birleştirmek ve aydınlatma etkilerinden kaçınmak için boyutlandırma ve normalizasyon işlemleri yapılır.
2. Aydınlatma efektlerini azaltmak ve komşu pikselleri dikkate almak için Gauss filtresi uygulanır.
3. HOG, her bir hücrede gradyanın büyüklüğünü ve yönünü hesaplar. Her bir hücre için bir histogram oluşturulur, ardından bu histogramlar birleştirilerek HOG özellik vektörü oluşturulur.
4. Son olarak, tüm bu histogramlar birleştirilerek 1D vektör olan HOG özellik tanımlayıcısı elde edilir.

Aşağıda HOG görüntülerinin görsel temsillerinden bazıları verilmiştir:



3- Yöntemin Öngörülen Performansı,

Avantajları ve Dezavantajları:

Performans: HOG ve ML kombinasyonu, yüz ifadesi tanıma görevlerinde iyi bir performans sergileyebilir. Ancak, veri setinin kalitesi, boyutu ve çeşitliliği bu performansı etkileyebilir.

Avantajlar:

- **Kaynak Verimliliği:** HOG, hesaplama açısından daha hafif olduğundan, kaynak kısıtlı ortamlar için daha uygun bir seçenektir.
- **Yapılandırılmış Veri:** HOG, yapılandırılmış veya görsel olmayan veri ile çalışırken tercih edilebilir. Yani, sadece görüntülerle sınırlı olmayan veri tiplerine de uygundur.

Dezavantajlar:

- **Kapsam ve Karmaşıklık:** HOG, yüz ifadeleri gibi karmaşık ve değişken veri setleriyle başa çıkmakta zorlanabilir. Özellikle sınıflar arasında belirgin ayrım olmadığı durumlarda performansı azalabilir.

Yöntem Oluşturulurken kullanılan makaleler:

<https://medium.com/swlh/histogram-of-oriented-gradients-hog-for-multiclass-image-classification-and-image-recommendation-cf0ea2caae8>

<https://medium.com/@entrepreneurbilal10/hog-vs-cnn-unveiling-the-power-of-image-classification-a3d1585d76e8>

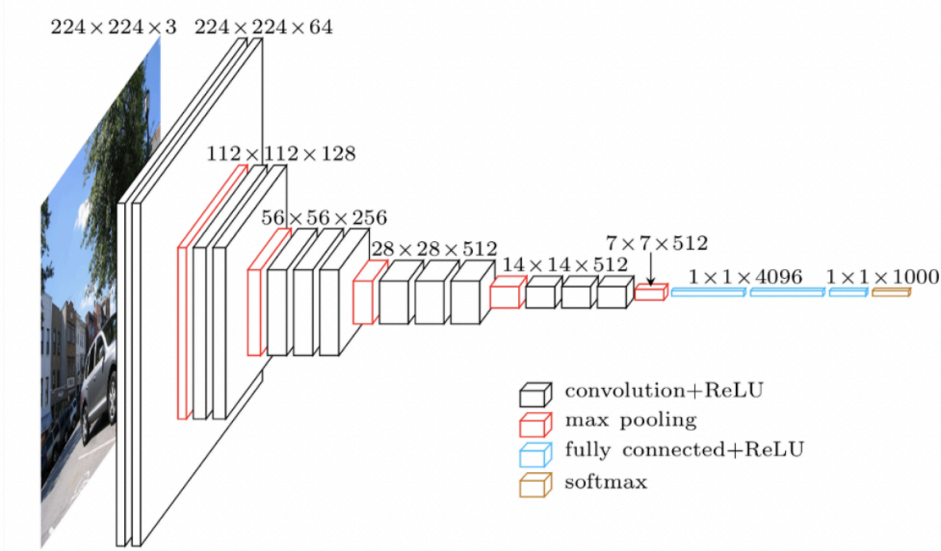
Yöntem 3: Bu metodoloji, yüz ifadelerini sınıflandırmak için kullanılan bir model geliştirmeyi amaçlar. Temel olarak, elimizdeki verileri kullanarak bir CNN yapısı oluşturmak bu etapta amaçlanmıştır. 2 adet aktarmalı 1 adet aktarmasız olmak üzere train veri setiyle 3 farklı CNN yapısı inşa edilecektir. Daha sonra test verisetimizle bu 3 farklı modelin F1, precision, accuracy gibi metrikler hesaplanarak kaydedilecektir. Elde edilen performans çıktıları kıyaslanarak raporlanacaktır.

1) VGG-19

İlk CNN modelimizde VGG-19 mimarisi üzerinden aktarmalı öğrenme (transfer learning) ile kendi verisetimizle tekrardan eğitim yapacağız.

1. Giriş Katmanı (Input Layer):
 - a. Giriş boyutu: 224x224 piksel renkli (RGB) görüntüler.
2. Katmanlar 1-2:
 - a. Convolutional + ReLU + Max Pooling: Convolutional Layer (3x3 çekirdek, 64 filtre, padding=1)
 - b. ReLU aktivasyon fonksiyonu
 - c. Max Pooling (2x2 çekirdek, adım=2)
3. Katmanlar 3-4: Convolutional + ReLU + Max Pooling:
 - a. Convolutional Layer (3x3 çekirdek, 128 filtre, padding=1)
 - b. ReLU aktivasyon fonksiyonu
 - c. Max Pooling (2x2 çekirdek, adım=2)
4. Katmanlar 5-8: 2 adet Convolutional + ReLU + Max Pooling:
 - a. 2 adet Convolutional Layer (3x3 çekirdek, 256 filtre, padding=1)
 - b. ReLU aktivasyon fonksiyonu
 - c. Max Pooling (2x2 çekirdek, adım=2)
5. Katmanlar 9-12: 4 adet Convolutional + ReLU + Max Pooling:
 - a. 4 adet Convolutional Layer (3x3 çekirdek, 512 filtre, padding=1)
 - b. ReLU aktivasyon fonksiyonu
 - c. Max Pooling (2x2 çekirdek, adım=2)
6. Katmanlar 13-16: 4 adet Convolutional + ReLU + Max Pooling:
 - a. 4 adet Convolutional Layer (3x3 çekirdek, 512 filtre, padding=1)
 - b. ReLU aktivasyon fonksiyonu
 - c. Max Pooling (2x2 çekirdek, adım=2)
7. Tam Bağlantı (Fully Connected) Katmanlar:
 - a. 2 adet Fully Connected Layer (Her biri 4096 nöron)
 - b. ReLU aktivasyon fonksiyonu
 - c. Dropout (overfitting önlemek için)
8. Çıkış Katmanı (Output Layer):
 - a. Fully Connected Layer (1000 nöron, ImageNet veri kümesi sınıf sayısı kadar)
 - b. Softmax aktivasyon fonksiyonu (sınıflar arasında olasılık hesaplamak için)

1- VGG-19 mimarisinde katmanları,kullanılan sequential convolution, yapısını aşağıda paylaştığımız görselden inceleyebilirsiniz.

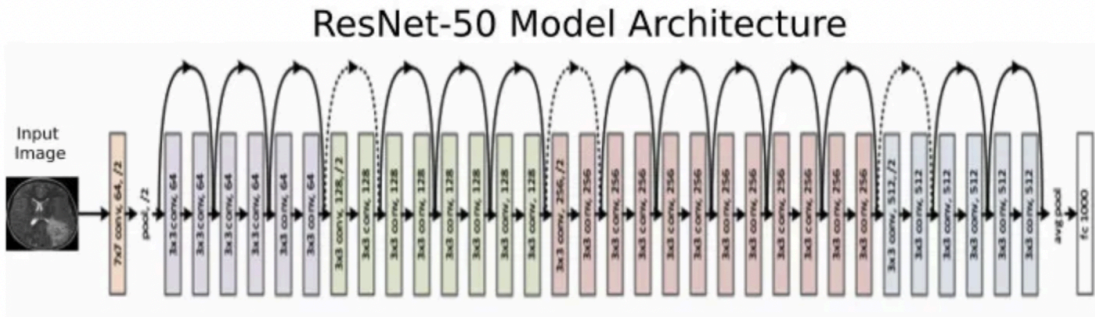


2) RESNET-50

İlk CNN modelimizde RESNET-50 mimarisi üzerinden aktarmalı öğrenme (transfer learning) ile kendi verisetimizle tekrardan eğitim yapacağız. RESNET-50 mimarisinde kullanılan sequential convolution yapısını aşağıda paylaştığımız görselden inceleyebilirsiniz.

ResNet'in genel yapısı residual block olarak adlandırılan temel yapıdan oluşur. Her residual block, iki ana yolu içerir: birincisi, doğrudan giriş (input) değeri olan x , ikincisi ise bu giriş değerinin üzerine eklenen bir artık (residual) blok tarafından üretilen bir çıktıdır. İki yolu birleştirmek için bir bağlantı (skip connection) kullanılır. Bu bağlantı, giriş değerini çıkışa ekler. Bu temel blok şu şekildedir:

1. Giriş (Input): x
2. İlk Katmanlar (Convolution, Batch Normalization, ReLU): $F(x)$
3. İkinci Katmanlar (Convolution, Batch Normalization): $G(F(x))$
4. Çıkış (Output): $G(F(x)) + x$



3) CNN

Son modelimizde kendi yapımızı tasarlayacağız. Bunun için aşağıda sıralayacağımız aşamaları uygulayarak ilerleyeceğiz.

1. Giriş Katmanı: Giriş katmanı, modelinize veri setinizi giriş olarak alacak ve işleyecektir. Tipik olarak, renkli bir görüntü veri seti için (genellikle 3 kanallı) boyutlandırma olarak (genellikle 224x224 piksel) tercih edilir
2. Evrişimli (Convolutional) Katmanlar: Evrişimli katmanlar, modelin görüntü üzerinde özellikleri öğrenmesine yardımcı olur. Farklı filtre sayıları ve boyutları kullanarak modelimizi zenginleştiririz.
3. Tam Bağlantılı Katmanlar: Tam bağlantılı katmanlar, evrişimli katmanlardan gelen özellikleri düzleştirir ve sınıflandırma yapar.
4. Çıkış Katmanı: Çıkış katmanı, modelin çıkışını verir. Sınıflandırma yapılacak sınıf sayısına göre nöron sayısı belirlenmelidir. Çoğu durumda, aktivasyon fonksiyonu 'softmax' olarak tercih edilir.

Bu 4 aşamada belirtilen kriterlere uyularak bir CNN modeli geliştireceğiz.

Avantajlar:

- Son Teknoloji Performansı: CNN'ler, görüntü sınıflandırma doğruluğu konusunda endüstri standardını belirleyerek son teknoloji performansını sergiler.
- Büyük Veri Setleri: CNN'ler, etiketli verinin bol olduğu durumlarda özellikle başarılıdır. Bu, öğrenme sürecinde modelin daha iyi genelleme yapabilmesini sağlar.
- CNN modelleri kendi yapıları gereği özellik çıkarmaya gerek duyulmamaktadır. Model kendi kendine bu özellik çıkarımını yaptığı için ekstra bir feature extraction yapacak iş gücüne ihtiyaç duyulmaktadır.

Dezavantajlar:

- Hesaplama Yüğü: CNN'ler genellikle daha karmaşık ve hesaplama yoğun modellerdir, bu da daha fazla hesaplama kaynağı gerektirir.
- Veri Seti Boyutları: Küçük veri setleri ile çalışırken, aşırı öğrenme riski taşıyabilir ve genelleme performansını olumsuz etkileyebilir. Bu kıyaslama, HOG ve CNN'nin farklı avantaj ve dezavantajlarını vurgulayarak, kullanılacak ortama ve veri setine bağlı olarak tercih edilebileceklerini göstermektedir.

https://www.tensorflow.org/tutorials/images/transfer_learning

<https://medium.com/@entrepreneurbilal10/hog-vs-cnn-unveiling-the-power-of-image-classification-a3d1585d76e8>

<https://medium.com/analytics-vidhya/multi-class-image-classification-using-transfer-learning-with-deep-convolutional-neural-networks-eab051cde3fbaqarvWKL6aJQD-l6g3toFehQaRJOyD0Z0cl/edit>

5. Projenin basit yol haritası (veri seti oluşturma takvimi) (2 kişiyeniz görev dağılımı)

Veri Seti Toplama ve Ön İşleme:

Veri setleri hazır, indireceğiz

Model Geliştirme:

Her biri farklı özellik çıkarımı ve sınıflandırma yöntemi kullanarak üç farklı model geliştirmeye çalışacağız: HOG ve ML, VGG-19, RESNET-50, ve kendi CNN modelimiz.

Her modelin mimarisini tasarlayacağız ve derin öğrenme kütüphaneleri (TensorFlow, Keras) kullanarak eğiteceğiz.

Eğitim sürecinde aşırı uyum (overfitting) problemini önlemek için uygun regularizasyon tekniklerini (dropout vb.) kullanacağız.

Model Değerlendirme:

Eğitilmiş modelleri test veri seti üzerinde değerlendireceğiz.

Doğruluk, hassasiyet, geri çağırma gibi performans metriklerini kullanarak modellerin performansını analiz edeceğiz.

Elde edilen sonuçları karşılaştırarak hangi modelin en iyi performansı gösterdiğini belirleyeceğiz .

Sonuçların Sunumu ve Raporlama:

Proje sonuçlarını raporlayın ve sunum haline getireceğiz.

Hangi modelin en iyi performansı gösterdiğini ve nedenini açıklayacağız..

Görev Dağılımı:

Veri Seti Toplama ve Ön İşleme: Her iki üye

Model Geliştirme: Her bir üye bir model üzerinde çalışacak ve sonuçları bir araya getirecek.

Model Değerlendirme: Her iki üye de modellerin performansını değerlendirecek ve sonuçları karşılaştıracak.

Sonuçların Sunumu ve Raporlama: Her iki üye de sonuçları birlikte sunacak ve raporlayacak