

# Facial Expression Recognition (FER2013)

Utku Murat ATASOY

211401019 / Yapay Zeka Mühendisliği

TOBB Ekonomi ve Teknoloji Üniversitesi

u.atasoy@etu.edu.tr

**Abstract**—Bu projede, FER2013 veri seti kullanılarak çok sınıflı yüz ifadesi tanıma (Facial Expression Recognition) problemi ele alınmıştır. Veri setindeki belirgin sınıf dengesizlikleri, özellikle *Disgust* sınıfının düşük temsili, model başarımını olumsuz yönde etkilemektedir. Bu durumu düzeltmek amacıyla üç farklı örnekleme stratejisi (*undersampling*, *oversampling*, *hybrid sampling*) ile çeşitli kayıp fonksiyonları (Cross Entropy, Class Weighted, Focal Loss) kullanılmıştır. Transfer learning yaklaşımıyla XceptionNet, MobileNet-V3-Large-100, EfficientNet-B0 ve ResNet-18 mimarileri eğitilmiş ve kıyaslanmıştır. En başarılı kombinasyon olan XceptionNet + Cross Entropy Loss + Hibrit Sampling (Target=8989) için yüz hizalama (face alignment) amacıyla FaceNet MTCNN yapısı kullanılmış ve bu sayede modelin sınıf bazlı başarımları önemli ölçüde iyileştirilmiştir. Elde edilen modeller ONNX formatına dönüştürülerek gerçek zamanlı performans testleri yapılmış ve sistemin %66.76 doğrulukla, düşük gecikme süreleri ile çalışabildiği gözlemlenmiştir.

**Index Terms**—Yüz ifadesi tanıma, FER2013, veri dengesizliği, transfer learning, sampling stratejileri, derin öğrenme.

## I. GİRİŞ

Yüz ifadesi tanıma (Facial Expression Recognition - FER), insan-makine etkileşimi, güvenlik, sürücü davranış analizi, eğitim ve sağlık gibi birçok alanda önemli uygulamalara sahiptir. Görüntü işleme ve derin öğrenme tekniklerindeki hızlı gelişmeler, yüz ifadelerinin otomatik olarak doğru sınıflandırılmasını mümkün kılmıştır. Bununla birlikte, veri setlerinde karşılaşılan sınıf dengesizlikleri (örneğin, *Disgust* sınıfının örnek sayısının azlığı) model başarımını olumsuz yönde etkilemektedir.

Bu çalışma kapsamında, çok sınıflı bir yüz ifadesi tanıma problemine yaklaşım getirilmiş; hedef, tüm duygu sınıflarını kapsayacak dengeli ve yüksek performanslı bir model geliştirmektir. Veri üzerinde hem sayısal hem de görsel kalite açısından dengeleme işlemleri uygulanmış; özellikle yüz hizalama adımı, MTCNN algoritması kullanılarak otomatik ve tutarlı bir şekilde gerçekleştirilmiştir.

Transfer learning tabanlı pre-trained derin öğrenme modelleri (CNN mimarileri) üzerinde farklı örnekleme ve loss stratejileri denenmiş; elde edilen en iyi sonuç, XceptionNet modeli kullanılarak hibrit örnekleme ve Cross Entropy Loss kombinasyonu ile sağlanmıştır. Modelin sınıf bazlı başarımları (Precision, Recall, F1 Score ve Accuracy) değerlendirilmiş, ayrıca ONNX formatına dönüştürülerek gerçek zamanlı uygulamalarda test edilmiştir. Test aşamasında XceptionNet, hizalama sonrası elde edilen test seti üzerinde **Accuracy = 66.76%** performansa ulaşmış; ONNX model-

leriyle yapılan gerçek zamanlı testlerde ise ortalama ~140 FPS gözlemlenmiştir.

Proje kapsamında geliştirilen kodlar ve sunum detaylarına ilişkin bağlantılar, raporun referans bölümünün en başında yer almaktadır ([1], [2]).

## II. LİTERATÜR TARAMASI

Yüz ifadesi tanıma (Facial Expression Recognition - FER) üzerine yapılan önceki çalışmalar incelendiğinde, özellikle FER2013 veri seti yaygın olarak kullanılmakta olup çoğu çalışmada transfer learning yöntemlerinin tercih edildiği görülmektedir. Literatürde sık karşılaşılan yaklaşımlardan biri ise *Disgust* gibi az temsil edilen sınıfların analiz dışı bırakılmasıdır. Bu sınıfı dışlamak model doğruluğunu artırsa da gerçek dünya uygulamaları açısından sınırlı bir çözüm sunmaktadır.

Khairuddin ve Chen [3], FER2013 üzerinde çeşitli transfer learning modelleriyle yaptıkları karşılaştırmalı analizde *Disgust* sınıfının dahil edilmesinin doğruluk oranlarını düşürdüğünü göstermiştir. Benzer şekilde Kusuma ve Lim [4], VGG-16 modeli ile yapılan ince ayarların performansa etkisini değerlendirerek veri dengesizliğinin sorun teşkil ettiğini vurgulamışlardır. Oguine ve arkadaşları [5], gerçek zamanlı yüz ifadesi tanıma sistemlerinde hibrit mimarilerin avantajlarını ortaya koymuş, Zahara ve ekibi [6] ise düşük güçlü cihazlarda CNN tabanlı modellerin mikro ifade tespiti üzerindeki başarımını analiz etmiştir.

Yeni literatür çalışmaları ise yüz tespiti ve hizalama adımlarında MTCNN ve FaceNet tabanlı yaklaşımların öne çıktığını göstermektedir. Qi ve arkadaşları [7], MTCNN'in P-Net, R-Net ve O-Net yapılarını kullanarak yüz bölgesi çıkarımını gerçekleştirmiş, ardından FaceNet aracılığıyla vektör bazlı yüz karşılaştırmaları yaparak kendi problemlerinde %86 doğruluk elde etmişlerdir. Wu ve Zhang [8], klasik algoritmaların (AdaBoost, Haar-like, DPM) hem yavaş hem de düşük doğrulukta olduğunu belirtmiş, MTCNN ve FaceNet'in bu sorunları ortadan kaldırarak gerçek zamanlı sistemlerde bazı problemlerde %99.85 doğruluğa ulaşabildiğini göstermiştir. Roy ve arkadaşları [9] ise MTCNN destekli FaceNet yapısını devreye alarak katılım takip sistemleri için %99.87 doğruluğa ulaşan bir model sunmuşlardır.

Projemizde, bu literatürdeki yöntemlerle uyumlu olarak MTCNN mimarisi kullanılarak yüz hizalama (alignment) gerçekleştirilmiş ve bu sayede modelin daha sağlıklı öğrenmesi sağlanmıştır. Ayrıca eğitim sonrası modeller ONNX formatına dönüştürülerek gerçek zamanlı testler yapılmış, lit-

eraturde bahsi geçen canlı uygulama senaryoları birebir gerçekleştirilmiştir. Bu bağlamda, literatürdeki birçok yaklaşımı aynı projede birleştirerek kapsamlı bir sistem sunulmuştur.

Aşağıdaki tablo, literatürdeki bazı çalışmaların *Disgust* sınıfını dahil edip etmemelerine göre doğruluk sonuçlarını özetlemektedir:

Metot	Data with Disgust	Data without Disgust
Transfer Learning using Xception	0.4078	0.4149
Transfer Learning + Fine Tuning using Xception	0.6485	0.6517
Transfer Learning using ResNet152V2	0.3798	0.3898
Transfer Learning + Fine Tuning using ResNet152V2	0.5551	0.2714
Transfer Learning using MobileNetV2	0.4013	0.3941
Transfer Learning + Fine Tuning using MobileNetV2	0.4771	0.4675
Transfer Learning using EfficientNetB0	0.4084	0.4122
Transfer Learning + Fine Tuning using EfficientNetB0	0.5856	0.5961
Transfer Learning using InceptionResNetV2	0.4153	0.4235
Transfer Learning + Fine Tuning using InceptionResNetV2	0.4138	0.4154
Self Made CNN	0.5857	0.5888

Bu tablo, *Disgust* sınıfının dahil edilmesi durumunda genel doğruluğun düşme eğiliminde olduğunu açıkça ortaya koymaktadır. Ancak gerçek dünya senaryolarında bu tür ifadelerin göz ardı edilmesi mümkün değildir. Bu nedenle kendi çalışmamızda tüm sınıflar modele dahil edilmiş ve veri dengesizliğini aşmak adına sampling stratejileri, weighted/focal loss uygulamaları ve yüz hizalama yöntemleri kullanılarak başarı seviyesi artırılmıştır.

Ayrıca açık kaynaklı projeler [10], [11] de hem mimari tercihler hem de uygulama örnekleri açısından incelenmiş ve bazı uygulama pratikleri bu projeye entegre edilmiştir.

### III. VERİ SETİ, VERİ ÖZELLİKLERİ VE ÖZNİTELİKLER

Bu bölümde, projede kullanılan FER2013 veri setinin kaynak bilgileri, öznitelik tanımları, veri ön işleme adımları, veri dengesizliği ve öznitelik açıklamaları detaylı şekilde sunulmaktadır. Ayrıca ham verinin normalize edilmesi, yüz hizalama işlemleri (FaceNet MTCNN tabanlı) ve modelin açıklanabilirliğine yönelik (XAI – LIME) yapılan analizler de burada anlatılmaktadır.

#### A. Veri Kaynağı ve Veri Kümesi Tanımı

Projede kullanılan veri seti, **FER2013** olup, Kaggle üzerinden erişilebilen ve 48x48 piksel boyutunda gri tonlamalı yüz görüntülerini içeren önemli bir kaynaktır. Veri seti, yedi duygu sınıfı (Angry, Disgust, Fear, Happy, Sad, Surprise, Neutral) barındırmaktadır. Detaylı bilgi ve indirme bağlantısı için buraya tıklayabilirsiniz.



Fig. 1. FER2013 veri setinden örnek görüntü.

#### B. Veri Kaynağı ve Veri Kümesi Tanımı

Toplamda **35.887** yüz görüntüsü içeren bu veri kümesi, özellikle sınıflar arası dengesizlik barındırması açısından dikkat çekicidir. Detaylı bilgi ve indirme bağlantısı için buraya tıklayabilirsiniz.

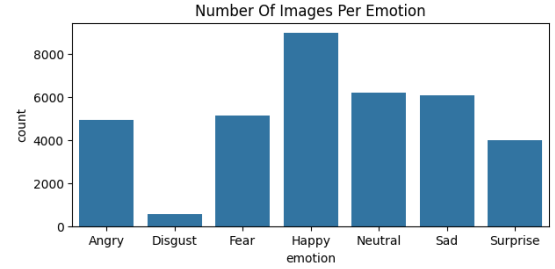


Fig. 2. Veri setinin işlenmemiş halindeki veri dağılımı

Bu veri setinde 7 farklı duyguya ait veriler bulunmaktadır ve sınıf başına dağılımları aşağıdaki gibidir:

- **Happy:** 8989 örnek
- **Neutral:** 6198 örnek
- **Sad:** 6077 örnek
- **Fear:** 5121 örnek
- **Angry:** 4953 örnek
- **Surprise:** 4002 örnek
- **Disgust:** 547 örnek

Bu dağılım, özellikle *Disgust* sınıfının diğer sınıflara kıyasla çok az örnek içermesi sebebiyle, model eğitimi sırasında dengesizlik sorunlarına yol açmaktadır. Diğer duygu sınıflarına göre çok daha düşük örnek sayısına sahip olan *Disgust* sınıfı, eğitim sürecinde modelin bu duyguya yeterince odaklanamamasına ve dolayısıyla istenen başarıyı göstermemesine sebep olmuştur. Literatürde ise yüksek doğruluk oranlarına ulaşan çalışmalarda sıklıkla *Disgust* sınıfının analiz dışı bırakıldığı ve 6 sınıf üzerinden ilerlenildiği gözlemlenmektedir.

Bu projede, sınıflar arası dengesizliğe rağmen *Disgust* sınıfının da dahil edilerek yüksek doğruluk elde edilmesi hedeflenmiş ve bu amaçla çeşitli örnekleme (sampling) stratejileri uygulanmıştır:

- **Undersampling:** Aşırı temsil edilen sınıflardan rastgele örnekler seçilerek, eğitim verisinde sınıf dengesizliği azaltılmıştır. Böylece modelin sık görülen sınıflara eğilim göstermesi engellenmiştir.
- **Oversampling:** Az sayıda örneğe sahip olan *Disgust* sınıfı çoğaltılarak modelin bu sınıfı daha iyi öğrenmesi sağlanmıştır. Bu sayede sınıf dengesinin artırılması hedeflenmiştir.

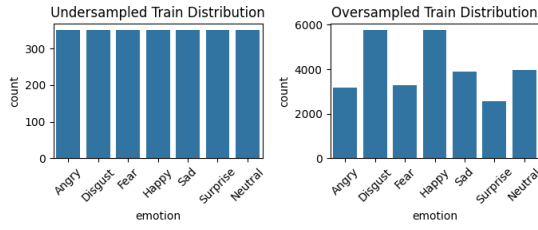


Fig. 3. Undersampling ve Oversampling uygulamaları sonrası sınıf dağılımları.

- **Hibrit Sampling:** Oversampling ve undersampling yöntemleri birleştirilerek, tüm sınıfların örnek sayısı belirli bir hedef değerde eşitlenmiştir. İlk stratejide tüm sınıflar yaklaşık 3000 örneğe, ikinci stratejide ise 8989 örneğe eşitlenmiştir.

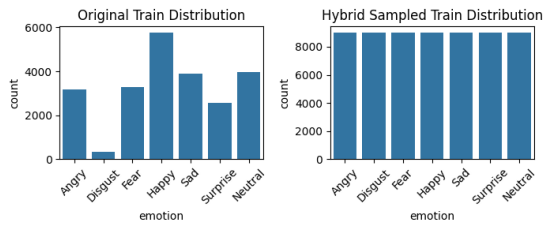


Fig. 4. Hibrit sampling sonrası tüm sınıflar dengeli hale getirilmiştir.

### C. Veri Ön İşleme, Normalizasyon ve Öznitelik Açıklamaları

Tüm veriler, aşağıdaki ön işleme adımlarıyla işlenmiştir:

- **Normalization:** Ham görüntüler, 0-255 aralığındaki piksel değerlerinden 0-1 aralığına normalize edilmiştir.
- **Görüntü İşleme:** Her görüntü, MTCNN ile yüz hizalama işlemi öncesinde RGB formatına dönüştürülüp, 96x96 boyutuna resize edilmiştir.
- **Data Augmentation:** Rotasyon, kırpma, yatay çevirme, parlaklık ve kontrast ayarı gibi veri artırma yöntemleri uygulanarak veri çeşitliliği sağlanmıştır.

Öznitelikler, her bir görüntüdeki piksel değerlerini temsil eden sayısal veriler olup, etiketler nominal (sırasız, çok sınıflı) kategorik veriler olarak yer almaktadır. Veri setinde özellikle *Disgust* sınıfı ciddi şekilde az temsil edilmektedir; bu durum modelin öğrenmesini zorlaştırmaktadır.

### D. Açıklanabilir Yapay Zeka (XAI) ile Model Yorumlanabilirliği: LIME

Daha önce hizalama işlemleri yapılmadan ve sadece sampling yöntemleri uygulanarak eğitilmiş modellerin karar verme mekanizmasını daha iyi anlamak için, **LIME (Local Interpretable Model-agnostic Explanations)** yöntemi uygulanmıştır. Bu yöntemle, modelin tahmin yaptığı örneklerde hangi görüntü bölgelerinin pozitif veya negatif katkı sağladığı renk kodlarıyla görselleştirilmiştir.

### Özellikle:

- **Angry:** Model, yüzün büyük kısmının gölgede kalması ve gözlük nedeniyle göz bölgesinin net görülmemesi sonucunda, belirgin olmayan yüz ifadelerini *Sad* olarak sınıflandırmıştır. Pozitif katkılar daha çok burun ve gözlük çevresinde toplanmıştır.
- **Disgust:** Burun çevresi, alın çizgileri ve kaşların ortasında yer alan bölgeler *Disgust* sınıfına ait pozitif katkılar taşımaktadır. Ancak bu bölgeler, *Angry* ifadesiyle de örtüştüğü için model yanlış tahminde bulunmuştur.
- **Surprise:** Ağız açıklığı ve göz çevresi *Surprise* ifadesi için önemli olsa da, model bu bölgeleri *Sad* ifadesiyle karıştırmış ve düşük ağırlıklı pozitif katkılar sunmuştur. Genel pozitif katkılar geniş bir alana yayılmıştır.
- **Neutral:** Yüzün simetrik ve gevşek kas yapısına sahip olması nedeniyle, model yanak ve göz çevresindeki hafif kıvrımları *Happy* sınıfı ile ilişkilendirmiştir. Bu nedenle ağız kenarındaki yumuşak hatlar pozitif katkı sağlamış, model tahmininde yanılığa neden olmuştur.

Aşağıdaki figürler, LIME analizine ait örnekleri göstermektedir:

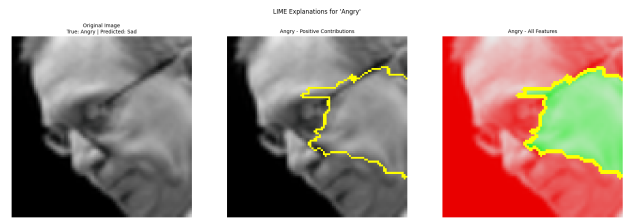


Fig. 5. **Angry** sınıfına ait bir örnekte modelin LIME açıklaması. Gerçek etiket *Angry* olmasına rağmen model *Sad* tahmini yapmıştır. Pozitif katkıların göz çevresi ve burun kısmında yoğunlaştığı görülmektedir.

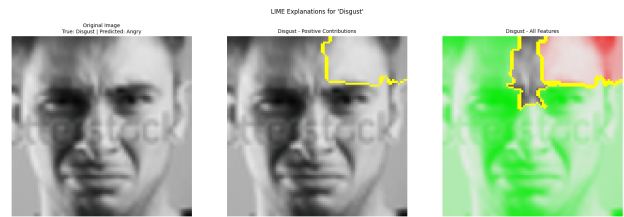


Fig. 6. **Disgust** sınıfına ait bir örnekte modelin LIME açıklaması. Gerçek etiket *Disgust* iken model *Angry* tahmini yapmıştır. Burun, alın ve kaş çevresi belirleyici bölgeler olarak öne çıkmaktadır.



Fig. 7. **Surprise** sınıfına ait bir örnekte modelin LIME açıklaması. Ağız ve göz çevresinin belirgin olmasına rağmen model ifadeyi *Sad* olarak tahmin etmiştir. Yeşil bölgeler pozitif katkı sağlar.

### E. Yüz Hizalama İşlemleri ve FaceNet MTCNN Uygulaması

Sampling ve farklı loss fonksiyonları denenerek elde edilen modellerden en başarılı olan **XceptionNet + Cross Entropy Loss (Hybrid Sampling, Target = 8989)** kombinasyonu, LIME (Local Interpretable Model-agnostic Explanations) yöntemiyle yorumlanabilirlik açısından incelenmiştir. LIME, modelin tahminini yaparken görüntünün hangi bölgelerine ne derecede odaklandığını göstererek, özellikle hatalı sınıflandırmaların nedenlerini analiz etmeyi mümkün kılar.

LIME görselleştirmelerinde fark edildiği üzere, bazı görüntülerde yüz bölgesi tam olarak görünmemekte, hatta model yüz dışındaki alanlara odaklanmaktadır. Bu durum, modelin hatalı tahmin yapmasına neden olmakta ve eğitimi de olumsuz etkilemektedir. Bu sorunu gidermek için verisetine yüz hizalama (*face alignment*) işlemi uygulanmıştır.

Bu işlemde, **FaceNet MTCNN (Multi-task Cascaded Convolutional Networks)** kullanılmıştır. MTCNN, bir görüntüdeki yüzleri tespit etmek ve hizalamak için üç aşamalı (P-Net, R-Net, O-Net) bir yapı kullanır:

- **P-Net:** Yüz aday bölgelerini tespit eder.
- **R-Net:** Aday bölgeleri filtreleyerek daha hassas yüz konumları çıkarır.
- **O-Net:** Nihai yüz sınır kutularını ve yüzün 5 temel noktasını (gözler, burun ucu, ağız köşeleri) belirler.

Projede bu yapı, **96x96 piksel** boyutlu hizalanmış yüz görüntüleri elde etmek amacıyla kullanılmıştır. İşlem, batch modda GPU üzerinde paralel olarak gerçekleştirilmiş ve hız açısından büyük bir avantaj sağlanmıştır.

Her bir örnek aşağıdaki hizalama işlemi adımlardan geçmiştir:

- 1) 48x48 boyutundaki gri görüntü RGB'ye çevrilmiştir.
- 2) MTCNN ile yüz bölgesi tespit edilmiş ve 96x96 boyutunda normalize edilmiştir.
- 3) Yüz tespit edilemeyen örnekler (isteğe bağlı parametre olarak) veri setinden çıkarılmıştır.

Aşağıda hizalama sürecinin bazı özet istatistikleri verilmiştir:

- **Eğitim seti:** 62,923 örnek → 10,463 yüz tespit edilemedi → **52,460** örnek kaldı.
- **Doğrulama seti:** 5,742 örnek → 928 yüz tespit edilemedi → **4,814** örnek kaldı.
- **Test seti:** 7,178 örnek → 1,188 yüz tespit edilemedi → **5,990** örnek kaldı.



Fig. 8. (a) Alignment Fig. 9. (b) FaceNet MTCNN ile uygulanmamış örnekler align edilmiş örnekler

Fig. 10. Yüz hizalamanın (face alignment) veri üzerindeki etkisi. (a) hizalanmamış görüntülerde kafa açısı, pozisyon ve çerçeveleme farklılıkları gözlemlenebilirken; (b) hizalanmış görüntülerde yüzler merkezi, düzgün ölçeklenmiş ve daha simetrik. Bu durum, modelin eğitimi ve genel başarımı açısından büyük avantaj sağlamaktadır.

Bu sonuçlar göstermektedir ki FER2013 veri setinde önemli bir kısımda yüz, model tarafından doğrudan algılanamamakta veya hatalı hizalanmaktadır. Bu nedenle yapılan bu ön işleme adımı modelin genel doğruluğunu ve kararlılığını artırmada oldukça etkili olmuştur.

### F. Yüz Hizalamanın LIME Açıklamaları Üzerindeki Etkisi

Yüz hizalama işleminin modelin karar mekanizması üzerindeki etkisini daha somut gösterebilmek amacıyla, LIME (Local Interpretable Model-agnostic Explanations) yöntemi kullanılarak elde edilen eski (Şekil 5, Şekil 6 ve Şekil 7) ve yeni açıklamalar karşılaştırılmıştır.

Alignment uygulanmadan önceki modellerde, LIME açıklamalarında modelin odaklandığı bölgelerin sınırlı olduğu ve bazen yüz dışındaki alanlara da dikkat verdiği gözlemlenmiştir. Örneğin Şekil 5'de, **Angry** sınıfına ait bir örnekte model, yalnızca göz çevresi ve burun gibi küçük bir alana odaklanmış; Şekil 6'te ise **Disgust** ifadesine ait örnekte, belirli alanlarda doğru odaklanma olmasına rağmen modelin **Sad** tahmini yaptığı görülmektedir. Şekil 7'de ise **Surprise** ifadesinde ağız ve gözler belirgin olmasına rağmen sınıflandırma hatası yapılmıştır.

Yüz hizalama (Face Alignment) uygulandıktan sonra ise modelin dikkatinin daha geniş ve anlamlı bir yüz alanına yayıldığı gözlemlenmiştir. Şekil 11, Şekil 12 ve Şekil 13 üzerinde görülebileceği gibi, model artık kaşlar, gözler, burun ve ağız gibi ifadeyi belirleyen tüm kritik bölgeleri daha dengeli şekilde dikkate almakta ve yüz dışı bölgeleri negatif katkı olarak doğru bir şekilde değerlendirmektedir.

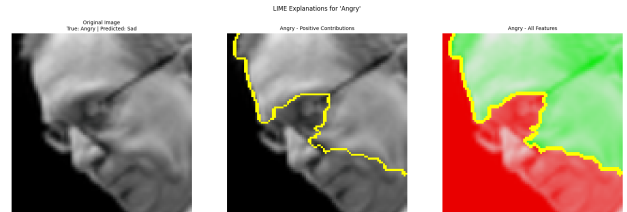


Fig. 11. Face alignment sonrası **Angry** sınıfına ait bir örnekte modelin LIME açıklaması. Model yüzün daha geniş bir bölgesine odaklanmıştır.



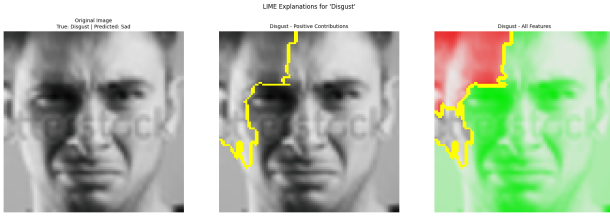


Fig. 12. Face alignment sonrası **Disgust** sınıfına ait bir örnekte modelin LIME açıklaması. Burun, alın ve kaş çevresine yapılan dengeli odaklanma sayesinde modelin dikkati daha doğru yönlendirilmiştir.

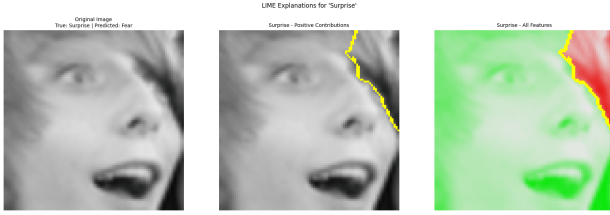


Fig. 13. Face alignment sonrası **Surprise** sınıfına ait bir örnekte modelin LIME açıklaması. İfadenin belirleyici bölgelerine daha dengeli bir odaklanma görülmektedir.

Bu sonuçlar, yüz hizalamanın modelin yalnızca doğruluğunu artırmakla kalmayıp aynı zamanda karar mekanizmasını daha güvenilir hale getirdiğini göstermektedir.

#### IV. KULLANILAN MODELLER

Bu projede, çok sınıflı (multi-class) yüz ifadesi tanıma problemi için transfer learning yaklaşımını temel alan dört farklı pre-trained CNN mimarisi kullanılmıştır. Modellerin eğitim, doğrulama ve test ayırımı %70, %10, %20 olarak belirlenmiştir. Her bir model, literatürdeki benzer çalışmalarla karşılaştırıldığında FER2013 veri seti üzerinde elde edilen sonuçların iyileştirilmesi amacıyla fine-tuning uygulanarak eğitilmiştir.

##### A. Model Seçimi ve Tanımları

- **XceptionNet:** Derin, modüler yapısı sayesinde geniş veri setlerinde üstün özellik çıkarımı yapabilen bir mimaridir. Literatürde, FER problemlerinde elde edilen yüksek başarımların bir kısmı XceptionNet üzerinde fine-tuning ile raporlanmıştır. Projemizde farklı sampling stratejileri ve kayıp fonksiyonları denemeleri sonucunda, hibrit sampling (Target = 8989) ve Cross Entropy Loss kombinasyonu ile en iyi sonuçları vermiştir.
- **MobileNet-V3-Large-100:** Hafif ve mobil cihazlar için optimize edilmiş bu model, düşük hesaplama gücü gereksinimiyle gerçek zamanlı uygulamalara uygunluk sağlamaktadır. Ancak, modelin mimarisinde bazı karmaşık ifadelerin ayırımında XceptionNet'e kıyasla düşük performans gözlemlenmiştir.
- **EfficientNet-B0:** Verimliliği ve parametre sayısı açısından optimize edilmiş olan EfficientNet-B0, ölçeklenebilirlik ve hızlı inference süreleri açısından avantajlıdır. Projemizde hem modelin genel başarısı

hem de gerçek zamanlı uygulamalardaki FPS/latency performansı değerlendirilmiştir.

- **ResNet-18:** Daha basit ve hafif bir yapıya sahip olan ResNet-18, residual bağlantılar sayesinde derin modellerde görülen gradyan kaybını minimize eder. Model, tutarlı ve istikrarlı sonuçlar vermesine rağmen diğer modellerle kıyaslandığında genel doğruluk oranında biraz geride kalmıştır.

##### B. Eğitim ve Değerlendirme Süreci

Tüm modeller, FER2013 veri seti üzerinde %70 eğitim, %10 doğrulama ve %20 test bölümü ayrımı kullanılarak eğitilmiştir. Çok sınıflı sınıflandırma problemi kapsamında her modelin performansı Precision, Accuracy, Recall ve F1 Score metrikleri ile değerlendirilmiştir. Eğitim sürecinde veri dengesizliğini gidermek amacıyla undersampling, oversampling ve hibrit sampling stratejileri uygulanmış; ayrıca Cross Entropy, Class Weighted ve Focal Loss gibi farklı kayıp fonksiyonları denenmiştir. Bu sayede, modellerin az temsil edilen *Disgust* sınıfı gibi zorlu sınıflarda dahi daha iyi performans sergilemesi sağlanmıştır.

##### C. Model Seçimindeki Gerekeç

Literatürde, transfer learning tabanlı modellerin FER problemlerinde yüksek başarı elde ettiği ve pre-trained ağların, özellikle veri miktarı kısıtlı durumlarda, üstün genelleme yeteneği sunduğu raporlanmıştır [3], [4], [5].

- **XceptionNet**, karmaşık yapıdaki ifadelerin ayırımında detaylı öznitelik çıkarımı yapabilmesi nedeniyle tercih edilmiştir.
- **MobileNet-V3-Large-100** ve **ResNet-18**, gerçek zamanlı uygulamalar ve düşük hesaplama gücü gerektiren senaryolar için seçilmiştir.
- **EfficientNet-B0**, verimliliği ve optimize parametre sayısı ile, hem yüksek doğruluk hem de hızlı inference süreleri sunarak uygulama esnasında avantaj sağlamaktadır.

Bu model seçimleri, hem eğitim sırasında elde edilen sonuçlarla hem de literatürde raporlanan değerlerle karşılaştırıldığında, projemizin gerçek dünya uygulamalarında kullanılabilirliğini artırmaya yönelik stratejiler geliştirmemizi sağlamıştır.

#### V. TEST SONUÇLARI

Projede, farklı örnekleme stratejileri ve kayıp fonksiyonları kullanılarak eğitim süreci detaylı olarak incelenmiştir. Aşağıda, proje planı, akış şeması, deneysel sonuçlar ve model test çıktıları yer almaktadır.

Projenin temel aşamaları Tablo I'da özetlenmiştir.

TABLE I  
PROJE AŞAMALARI VE AÇIKLAMALARI

Aşama	Açıklama
Veri Seti Temini	Kaggle'dan FER2013 veri seti temin edilmiştir. Veri seti, 48x48 piksel gri tonlamalı yüz görüntüleri ve yedi duygu sınıfını içermektedir.
Veri Ön İşleme	Görüntüler, ham veriden normalize edilip, 48x48 boyutuna getirilmiş; rotasyon, kırpma, parlaklık ve kontrast ayarları gibi veri artırma (augmentation) yöntemleri uygulanmıştır.
Sampling Stratejileri	Veri setindeki sınıf dengesizliğini gidermek için undersampling, oversampling ve hibrit sampling yöntemleri uygulanmıştır.
Kayıp Fonksiyonları	Model eğitimi sırasında Cross Entropy, Class Weighted ve Focal Loss gibi farklı kayıp fonksiyonları kullanılarak, veri dengesizliğinin etkileri minimize edilmeye çalışılmıştır.
Model Eğitimi	Transfer learning yaklaşımıyla, XceptionNet, MobileNet-V3-Large-100, EfficientNet-B0 ve ResNet-18 modelleri üzerinde fine-tuning gerçekleştirilmiştir. Eğitim/ doğrulama/test ayırımı %70/10/20 şeklinde yapılmıştır.
Performans Değerlendirme	Eğitim sonrası modellerin performansı Precision, Accuracy, Recall ve F1 Score metrikleri kullanılarak hesaplanmış; ayrıca confusion matrix ve sınıf bazlı classification report'lar oluşturulmuştur.
XAI ile Model Yorumlanabilirliği	LIME yöntemi kullanılarak, modelin hangi görüntü bölgelerine odaklandığı görselleştirilmiş; böylece hatalı sınıflandırmaların nedenleri analiz edilmiştir.
Yüz Hizalama ve Yeniden Eğitim	LIME analizlerinden elde edilen bulgular doğrultusunda, FaceNet MTCNN mimarisi ile yüz hizalama (face alignment) uygulanmıştır. Bu hizalanmış veri seti ile modeller yeniden eğitilmiş ve sınıf bazlı başarımlarda önemli iyileşmeler gözlemlenmiştir.
Test	Yeniden eğitilen modeller, test seti üzerinde değerlendirilmiştir. Modelin genel başarısı ve sınıf bazındaki performansları detaylı olarak analiz edilmiştir.
Gerçek Zamanlı Uygulama	En iyi modeller, ONNX formatına dönüştürülmüş ve gerçek zamanlı canlı video akışı üzerinde, FaceNet MTCNN destekli yüz tespiti, preprocess ve softmax ile sınıflandırma yapıpıacak şekilde test edilmiştir. Sistem düşük gecikme süresi ve yüksek FPS ile çalışacak şekilde optimize edilmiştir.

## A. Elde Edilen Sonuçlar

1) *Sampling Uygulanmadan (Baseline)*: Tablo II'da, sampling uygulanmadan elde edilen modellerin performans metrikleri sunulmuştur.

TABLE II  
SAMPLING UYGULANMADAN ELDE EDİLEN PERFORMANS METRİKLERİ

Model Adı	Loss Tipi	Epoch	Batch Size	Precision	Accuracy	Recall	F1 Score
XceptionNet	Cross Entropy	30	64	0.6692	0.6708	0.6708	0.6692
MobileNet-V3-Large-100	Cross Entropy	30	64	0.6441	0.6461	0.6461	0.6442
EfficientNet-B0	Cross Entropy	30	64	0.6442	0.6442	0.6442	0.6436
ResNet-18	Cross Entropy	30	64	0.6437	0.6454	0.6454	0.6431
XceptionNet	Class Weighted	30	64	0.6590	0.6602	0.6602	0.6588
MobileNet-V3-Large-100	Class Weighted	30	64	0.6252	0.6301	0.6301	0.6263
EfficientNet-B0	Class Weighted	30	64	0.6365	0.6369	0.6369	0.6365
ResNet-18	Class Weighted	30	64	0.6330	0.6339	0.6339	0.6322
XceptionNet	Focal	30	64	0.6568	0.6565	0.6565	0.6556
MobileNet-V3-Large-100	Focal	30	64	0.6315	0.6282	0.6282	0.6292
EfficientNet-B0	Focal	30	64	0.6259	0.6269	0.6269	0.6263
ResNet-18	Focal	30	64	0.6227	0.6265	0.6265	0.6224

2) *Undersampling Uygulandığında*: Tablo III'da undersampling stratejisi ile elde edilen sonuçlar yer almaktadır. Bu yöntemde, veri miktarının aşırı azaltılmasının model performansına olumsuz etkileri gözlenmiştir.

TABLE III  
UNDERSAMPLING İLE ELDE EDİLEN PERFORMANS METRİKLERİ

Model Adı	Loss Tipi	Epoch	Batch Size	Precision	Accuracy	Recall	F1 Score
XceptionNet	Cross Entropy	30	64	0.5334	0.5199	0.5199	0.5218
MobileNet-V3-Large-100	Cross Entropy	30	64	0.4231	0.3831	0.3831	0.3967
EfficientNet-B0	Cross Entropy	30	64	0.4287	0.4061	0.4061	0.4131
ResNet-18	Cross Entropy	30	64	0.5139	0.4858	0.4858	0.4951
XceptionNet	Class Weighted	30	64	0.5077	0.4784	0.4784	0.4872
MobileNet-V3-Large-100	Class Weighted	30	64	0.3885	0.3150	0.3150	0.3336
EfficientNet-B0	Class Weighted	30	64	0.3892	0.3313	0.3313	0.3476
ResNet-18	Class Weighted	30	64	0.4195	0.2751	0.2751	0.2958
XceptionNet	Focal	30	64	0.5136	0.4827	0.4827	0.4914
MobileNet-V3-Large-100	Focal	30	64	0.3679	0.2888	0.2888	0.3088
EfficientNet-B0	Focal	30	64	0.3722	0.3218	0.3218	0.3352
ResNet-18	Focal	30	64	0.4270	0.2871	0.2871	0.3155

3) *Disgust Sınıfı İçin Oversampling Uygulandığında*: Tablo IV'da, özellikle az temsil edilen *Disgust* sınıfının oversampling yöntemiyle sayısının artırılması sonucu elde edilen sonuçlar gösterilmiştir.

TABLE IV  
OVERSAMPLING (DISGUST) İLE ELDE EDİLEN PERFORMANS METRİKLERİ

Model Adı	Loss Tipi	Epoch	Batch Size	Precision	Accuracy	Recall	F1 Score
XceptionNet	Cross Entropy	30	64	0.6599	0.6627	0.6627	0.6605
MobileNet-V3-Large-100	Cross Entropy	30	64	0.6347	0.6400	0.6400	0.6369
EfficientNet-B0	Cross Entropy	30	64	0.6296	0.6339	0.6339	0.6310
ResNet-18	Cross Entropy	30	64	0.6366	0.6407	0.6407	0.6377
XceptionNet	Class Weighted	30	64	0.6528	0.6580	0.6580	0.6539
MobileNet-V3-Large-100	Class Weighted	30	64	0.6266	0.6241	0.6241	0.6240
EfficientNet-B0	Class Weighted	30	64	0.6270	0.6291	0.6291	0.6276
ResNet-18	Class Weighted	30	64	0.6219	0.6232	0.6232	0.6216
XceptionNet	Focal	30	64	0.6569	0.6555	0.6555	0.6559
MobileNet-V3-Large-100	Focal	30	64	0.6181	0.6197	0.6197	0.6177
EfficientNet-B0	Focal	30	64	0.6264	0.6254	0.6254	0.6242
ResNet-18	Focal	30	64	0.6193	0.6154	0.6154	0.6156

4) *Hibrit Sampling Uygulaması*: Hibrit sampling yöntemi iki farklı target değer (3000 ve 8989) ile uygulanmış; Tablo V ve Tablo VI'da sırasıyla sonuçlar sunulmuştur. Özellikle Target = 8989 uygulamasında, XceptionNet modeli ile Cross Entropy loss kullanımı en başarılı kombinasyonu ortaya koymuştur.

TABLE V  
HIBRIT SAMPLING (TARGET = 3000) İLE ELDE EDİLEN PERFORMANS METRİKLERİ

Model Adı	Loss Tipi	Epoch	Batch Size	Precision	Accuracy	Recall	F1 Score
XceptionNet	Cross Entropy	30	64	0.6487	0.6439	0.6439	0.6450
MobileNet-V3-Large-100	Cross Entropy	30	64	0.6232	0.6134	0.6134	0.6174
EfficientNet-B0	Cross Entropy	30	64	0.6223	0.6165	0.6165	0.6187
ResNet-18	Cross Entropy	30	64	0.6245	0.6226	0.6226	0.6231
XceptionNet	Class Weighted	30	64	0.6509	0.6357	0.6357	0.6409
MobileNet-V3-Large-100	Class Weighted	30	64	0.6002	0.5952	0.5952	0.5971
EfficientNet-B0	Class Weighted	30	64	0.6088	0.5957	0.5957	0.6007
ResNet-18	Class Weighted	30	64	0.5974	0.5910	0.5910	0.5916
XceptionNet	Focal	30	64	0.6556	0.6521	0.6521	0.6525
MobileNet-V3-Large-100	Focal	30	64	0.6232	0.6134	0.6134	0.6174
EfficientNet-B0	Focal	30	64	0.6219	0.6163	0.6163	0.6184
ResNet-18	Focal	30	64	0.6278	0.6293	0.6293	0.6280

TABLE VI  
HIBRIT SAMPLING (TARGET = 8989) İLE ELDE EDİLEN PERFORMANS METRİKLERİ

Model Adı	Loss Tipi	Epoch	Batch Size	Precision	Accuracy	Recall	F1 Score
XceptionNet	Cross Entropy	30	64	0.6630	0.6668	0.6668	0.6636
MobileNet-V3-Large-100	Cross Entropy	30	64	0.6377	0.6402	0.6402	0.6378
EfficientNet-B0	Cross Entropy	30	64	0.6332	0.6351	0.6351	0.6326
ResNet-18	Cross Entropy	30	64	0.6298	0.6293	0.6293	0.6290
XceptionNet	Class Weighted	30	64	0.6523	0.6545	0.6545	0.6523
MobileNet-V3-Large-100	Class Weighted	30	64	0.6295	0.6291	0.6291	0.6289
EfficientNet-B0	Class Weighted	30	64	0.6247	0.6241	0.6241	0.6229
ResNet-18	Class Weighted	30	64	0.6314	0.6262	0.6262	0.6283
XceptionNet	Focal	30	64	0.6535	0.6500	0.6500	0.6508
MobileNet-V3-Large-100	Focal	30	64	0.6298	0.6271	0.6271	0.6275
EfficientNet-B0	Focal	30	64	0.6190	0.6213	0.6213	0.6194
ResNet-18	Focal	30	64	0.6316	0.6269	0.6269	0.6284

## B. Test Sonuçları ve Analizler

Hibrit Sampling (Target = 8989) ve Cross Entropy Loss kullanılarak eğitilmiş XceptionNet modelinin test sonuçları değerlendirilmiştir.

1) *Örnek Test Görüntüleri*: Gerçek ve tahmin edilen etiketler açısından örnek test batch'leri Şekil 14'de sunulmuştur.



Fig. 14. Örnek Test Batch – Gerçek ve Tahmin Edilen Etiketler

### C. Açıklanabilir Yapay Zeka (XAI) ve Yüz Hizalama + FaceNet MTCNN işlemleri sonrası Sonuçlar

1) *Alignment Sonrası Sınıf Bazlı Başarı Karşılaştırması (XceptionNet)*: Yüz hizalama (alignment) işleminin model performansı üzerindeki etkisini daha net göstermek adına, XceptionNet modeline ait sınıf bazlı başarı metrikleri karşılaştırılmıştır. Aşağıdaki görsellerde alignment işlemi **öncesi** ve **sonrası** için her bir sınıfa ait Precision, Recall ve F1-Score değerleri sunulmuştur.

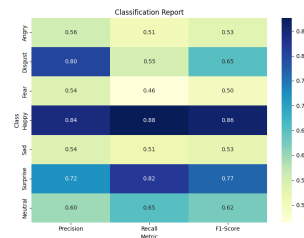
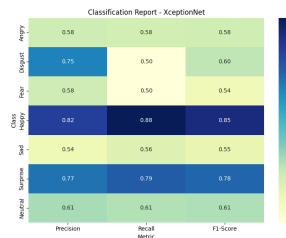


Fig. 15. (a) Alignment öncesi  
XceptionNet sınıf bazlı başarı

Fig. 16. (b) Alignment sonrası  
XceptionNet sınıf bazlı başarı

Fig. 17. Yüz hizalama işlemi öncesi ve sonrası XceptionNet modeli ile elde edilen sınıf bazlı classification report karşılaştırması.

Yukarıdaki Şekil 15 ve Şekil 16 görselleri karşılaştırıldığında, align edilmiş veri kullanıldığında modelin daha istikrarlı ve dengeli sonuçlar verdiği görülmektedir. Özellikle daha önce düşük başarı gösteren sınıflarda belirgin artışlar kaydedilmiştir:

- **Disgust** sınıfında:
  - Precision: **0.75** → **0.80**
  - Recall: **0.50** → **0.55**
  - F1-Score: **0.60** → **0.65**
- **Fear** sınıfında F1-Score: **0.54** → **0.50** (küçük bir düşüş olsa da, diğer sınıflardaki kazanımlar daha anlamlıdır)
- **Sad** sınıfı F1-Score: **0.55** → **0.53**
- **Neutral** sınıfında F1-Score: **0.61** → **0.62**
- **Surprise** sınıfı F1-Score: **0.78** → **0.77** (denge korunmuş)

artırarak sınıf ayrımını kolaylaştırdığını göstermektedir. Bu iyileştirme, hem genel başarıyı hem de sınıflar arası adaleti artırmıştır.

TABLE VII  
FACE ALIGNMENT (MTCNN) SONRASI MODELLERIN GENEL  
PERFORMANS KARŞILAŞTIRMASI

Model	Precision	Accuracy	Recall	F1 Score
XceptionNet	0.6647	0.6676	0.6676	0.6644
MobileNet-V3-Large-100	0.5959	0.5920	0.5920	0.5929
EfficientNet-B0	0.6116	0.6070	0.6070	0.6077
ResNet-18	0.6238	0.6255	0.6255	0.6238

Bu tabloda, MTCNN ile yüz hizalama (face alignment) işlemi uygulandıktan sonra eğitilen modellerin genel başarı metrikleri sunulmaktadır. Özellikle **XceptionNet** modeli, tüm metriklerde en yüksek performansa ulaşarak hizalama işleminin model başarısını artırmada etkili olduğunu göstermiştir.

2) *Gerçek Zamanlı Uygulama ve ONNX Performans Karşılaştırması*: Eğitim süreci tamamlandıktan sonra, modellerin gerçek zamanlı yüz ifadesi tanıma uygulamalarında kullanılabilirliğini değerlendirmek amacıyla tüm modeller **ONNX (Open Neural Network Exchange)** formatına dönüştürülmüştür. ONNX, farklı framework'ler arası birlikte çalışabilirlik sağlayan ve düşük seviyeli inference motorları (örneğin: ONNX Runtime, TensorRT) ile optimize çalışan açık bir formattır. Aşağıda, önce normal ONNX modellerinin performans metrikleri, ardından **simplified** (sadeleştirilmiş) ONNX modellerinin performans metrikleri sunulmakta ve sonrasında elde edilen sonuçlar yorumlanmaktadır.

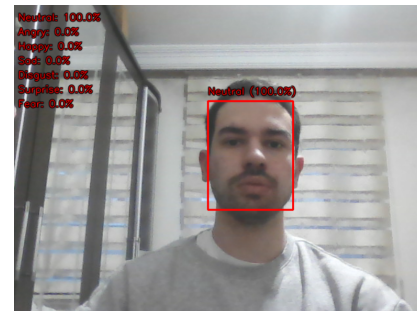


Fig. 18. Gerçek zamanlı yüz ifadesi tanıma uygulamasından bir ekran görüntüsü.

Şekil 18’de görüldüğü üzere, geliştirilen sistem gerçek zamanlı kamera görüntüsü üzerinden çalışmakta ve tespit edilen yüz bölgesine karşılık gelen duygu sınıfını anlık olarak tahmin etmektedir. Bu örnekte model, %100 güvenle yüz ifadesini **Neutral** olarak sınıflandırmıştır.

Uygulama süreci şu adımları içermektedir:

- Kamera görüntüsünden alınan kareler üzerinde **MTCNN** kullanılarak yüz tespiti yapılır.
- Elde edilen yüz, normalize edilerek **96x96** boyutuna getirilir ve **ONNX** modeline input olarak verilir.

- Model çıktısı softmax işlemiyle yorumlanarak, en olası duygu sınıfı ekran üzerine yüz kutusuyla birlikte yazdırılır.
- Bu işlemler saniyede onlarca kez tekrarlandığı için sistem oldukça akıcı çalışmaktadır.

Geliştirilen bu sistem sayesinde, farklı derin öğrenme mimarileri ONNX formatında çalıştırılarak gerçek zamanlı yüz ifadesi tanıma gerçekleştirilmiş; düşük gecikme süreleri ve yüksek FPS değerleri ile uygulamanın pratik sahalarda kullanılabilirliği gösteril

Normal ONNX Modelleri Performansı Modeller, PyTorch üzerinden eğitildikten sonra, 96x96 boyutunda dummy input kullanılarak ONNX formatına export edilmiştir. Aşağıdaki tablo, normal ONNX modelleri için elde edilen performans metriklerini göstermektedir:

TABLE VIII  
NORMAL ONNX MODELLERİNİN GERÇEK ZAMANLI PERFORMANS METRİKLERİ

Model	FPS Mean	FPS Min	FPS Max	Latency Mean (ms)	Latency Min (ms)	Latency Max (ms)	Frame Count
XceptionNet	140.64	105.17	199.78	7.11	5.01	9.51	152
MobileNet-V2	423.53	124.98	500.27	2.36	2.0	8.0	152
EfficientNet-B0	318.17	199.59	499.32	3.14	2.0	5.01	153
ResNet-18	245.48	166.5	333.41	4.07	3.0	6.01	152

Simplified ONNX Modellerinin Performansı Aynı modeller, onnx-simplifier kütüphanesi yardımıyla sadeleştirilmiş ONNX formatına dönüştürülmüştür. Bu sadeleştirme işlemi, inference sırasında modelin daha az kaynak tüketmesini sağlayarak, performansı optimize etmeye yöneliktir. Aşağıdaki tablo, simplified ONNX modelleri için elde edilen performans metriklerini sunmaktadır:

TABLE IX  
SIMPLIFIED ONNX MODELLERİNİN GERÇEK ZAMANLI PERFORMANS METRİKLERİ

Model	FPS Mean	FPS Min	FPS Max	Latency Mean (ms)	Latency Min (ms)	Latency Max (ms)	Frame Count
XceptionNet	139.2	99.89	166.63	7.18	6.0	10.01	150
MobileNet-V2	432.59	124.98	500.10	2.31	2.0	8.0	156
EfficientNet-B0	322.68	249.44	499.86	3.10	2.0	4.01	153
ResNet-18	245.48	166.09	333.36	4.07	3.0	6.02	152

Yorum ve Karşılaştırma Normal ONNX modelleri (Tablo VIII) ile simplified ONNX modellerinin (Tablo IX) performansları karşılaştırıldığında aşağıdaki hususlar öne çıkmaktadır:

- **XceptionNet:** Normal ONNX versiyonunda ortalama FPS 140.64 ve latency ortalama 7.11 ms iken, simplified versiyonda FPS ortalaması 139.2 ve latency 7.18 ms olarak kaydedilmiştir. Küçük farklar mevcut olup, genel performans benzerlik göstermektedir.
- **MobileNet-V2:** Normal versiyonda ortalama FPS 423.53 iken, simplified versiyonda FPS değeri 432.59 olarak artmıştır. Latency değerlerinde de ufak bir iyileşme gözlenmiştir (2.36 ms'den 2.31 ms'ye).
- **EfficientNet-B0:** Her iki versiyonda da ortalama FPS ve latency değerleri benzer kalmıştır; model sadeleştirme sonrası performansını korumaktadır.
- **ResNet-18:** Normal ve simplified versiyonlar arasında fark neredeyse gözle görülür düzeyde değildir.

Sonuç olarak, simplified ONNX modelleri, inference sırasında kaynak kullanımında küçük iyileştirmeler sağlamakta ancak genel olarak modellerin FPS ve latency gibi temel performans metriklerini önemli ölçüde değiştirmemektedir. ONNX-simplifier işleminin, model karmaşıklığını azaltırken performansı korumada etkili olduğunu göstermektedir. Gerçek zamanlı uygulamalarda, yüksek FPS ve düşük latency değerleri sayesinde tüm modellerin uygulanabilir olduğu tespit edilmiştir.

## VI. SONUÇLAR

Bu projede, FER2013 veri seti kullanılarak çok sınıflı yüz ifadesi tanıma problemi ele alınmıştır. Veri setindeki belirgin dengesizlikler – özellikle *Disgust* sınıfının az temsil edilmesi – nedeniyle, farklı örnekleme stratejileri (undersampling, oversampling, hibrit sampling) ve çeşitli kayıp fonksiyonları (Cross Entropy, Class Weighted, Focal Loss) uygulanarak model başarımının artırılması hedeflenmiştir. Ayrıca, FaceNet MTCNN kullanılarak gerçekleştirilen yüz hizalama işlemleri, ham ve düzensiz veriler yerine daha tutarlı, normalize edilmiş yüz görüntülerinin elde edilmesini sağlamıştır. Transfer learning yaklaşımıyla XceptionNet, MobileNet-V3-Large-100, EfficientNet-B0 ve ResNet-18 modelleri üzerinde fine-tuning gerçekleştirilmiş; test sonuçları ve sınıf bazlı değerlendirmeler ile XceptionNet modeli özellikle üstün performans sergilemiştir. ONNX formatına dönüştürülen modellerin gerçek zamanlı uygulamalarda yüksek FPS ve düşük latency değerleri sunması, sistemin pratik kullanıma uygun olduğunu göstermiştir. Ayrıca, LIME yöntemi ile modelin karar verme süreçleri analiz edilerek hatalı sınıflandırmaların nedenlerine dair önemli içgörüler elde edilmiştir.

### Öğrendiklerimiz ve Kazandıklarımız:

- Tüm duygu sınıflarını içeren dengeli bir yüz ifadesi tanıma sistemi geliştirmek için veri dengesizliğine yönelik örnekleme stratejileri ve özel kayıp fonksiyonları uygulanmasının etkili olduğu;
- Yüz hizalama (face alignment) işleminin, özellikle düşük temsiliyet gösteren sınıflarda (örneğin, *Disgust*) modelin başarımını artırdığı,
- ONNX formatına geçiş ve sadeleştirme (simplify) işlemlerinin, gerçek zamanlı uygulamalarda modelin performansını (FPS, latency) optimize ettiğini gözlemledik;
- LIME tabanlı açıklanabilirlik analizi, modelin hangi özelliklere odaklandığını ortaya koyarak, gelecekte yapılabilecek iyileştirmeler için yol gösterici oldu.

### Yapamadıklarımız ve Kısıtlamalar:

- Veri setinin çözünürlüğü ve tek kanallı (grayscale) yapısı, bazı ince özneliliklerin yeterince öğrenilememesine neden olmuştur.
- *Disgust* gibi sınıfların çok az temsil edilmesi, bazı modellerde hala karışıklıklara yol açmaktadır.
- Modelin gerçek dünya koşullarında daha geniş ve çeşitlendirilmiş veri setleriyle test edilmesi, ileri aşamalarda yapılabilecek geliştirmeler arasındadır.

### Karşılaştırmalı Model Performansı:



- Literatürde yer alan ve benzer veri seti kullanılarak gerçekleştirilen bir çalışmada, Transfer Learning + Fine Tuning yöntemiyle Xception modeli kullanılarak elde edilen en iyi accuracy skoru [11] 0.6485 olarak bildirilmiştir. Buna karşın, bu çalışmada geliştirilen model; XceptionNet + Cross Entropy Loss + MTCNN FaceNET Alignment kombinasyonu ile test seti üzerinde **0.6676** accuracy skoru elde etmiştir. Bu sonuç, önerilen yaklaşımın modelin genel doğruluk performansını artırmada etkili olduğunu göstermektedir.

Gelecek araştırmalarda, yüksek çözünürlüklü ve çok kanallı yüz görüntüleri kullanılarak modelin öznetelik çıkarım kapasitesinin artırılması; farklı yüz hizalama algoritmalarının ve veri artırma stratejilerinin incelenmesi; ayrıca ONNX modellerinin donanım optimizasyonları (örneğin, TensorRT) ile gerçek zamanlı uygulamalara entegrasyonunun detaylı olarak ele alınması önerilmektedir. Böylece gerçek dünya uygulamalarında daha yüksek doğruluk, daha düşük gecikme ve daha geniş kapsama alanı sağlanabilir.

Sonuç olarak bu çalışma, FER2013 veri seti üzerinde tüm duygu sınıflarını dikkate alarak veri dengesizliği ve yüz hizalama gibi zorluklara karşı uygulanabilir çözümler sunmakta, gerçek zamanlı yüz ifadesi tanıma sistemlerinde kullanılabilecek dengeli ve yüksek performanslı bir model geliştirilmesine önemli katkılar sağlamaktadır.

## REFERENCES

- [1] Utku.Murat.Atasoy, "Kod linki," [https://drive.google.com/drive/folders/1Nr4n\\_tVyyQNQR8y6CTdmh4-xb0j9qJMR?usp=sharing](https://drive.google.com/drive/folders/1Nr4n_tVyyQNQR8y6CTdmh4-xb0j9qJMR?usp=sharing), rapora gömülü kod klasörleri için bağlantı.
- [2] Utku.Murat.Atasoy, "Sunum videosu," <https://youtu.be/779Kehtc7i8?si=jvS4lshXKJmPJ-Kr>, youTube üzerinden sunum.
- [3] Y. Khairuddin and Z. Chen, "Facial emotion recognition: State of the art performance on fer2013," <https://paperswithcode.com/sota/facial-expression-recognition-on-fer2013>, 2021, submitted on 8 May 2021.
- [4] G. P. Kusuma and J. A. P. Lim, "Emotion recognition on fer-2013 face images using fine-tuned vgg-16," 2020, submitted on 10 November 2020.
- [5] O. C. Oguine, K. A. Kinfu, K. J. Oguin, and D. Ofuani, "Hybrid facial expression recognition (fer2013) model for real-time emotion classification and prediction," 2022, june 2022.
- [6] L. Zahara, P. Musa, E. P. Wibowo, I. Karim, and S. B. Musa, "The facial emotion recognition (fer-2013) dataset for prediction system of micro-expressions face using the convolutional neural network (cnn) algorithm based raspberry pi," *IEEE Xplore*, 2020, date Added to IEEE Xplore: 24 December 2020.
- [7] S. Qi, X. Zuo, W. Feng, and I. G. Naveen, "Face recognition model based on mtcnn and facenet," *IEEE*, 2022, trained on LFW dataset, 86.0% accuracy.
- [8] C. Wu and Y. Zhang, "Mtcnn and facenet based access control system for face detection and recognition," *Automatic Control and Computer Sciences*, vol. 55, no. 1, pp. 102–112, 2021.
- [9] M. K. Roy, P. Dwibedi, A. Singh, R. P. Chakraborty, and M. K. H. Mondal, "Mtcnn and facenet-based face detection and recognition model for attendance monitoring," in *Conference Paper*, 2024, first Online: 18 February 2024.
- [10] Codejay12, "Facial-expression-recognition," <https://github.com/codejay12/Facial-Expression-Recognition>, 2020.
- [11] L. Rose, "Facial-expression-recognition," <https://github.com/leorrose/Facial-Expression-Recognition>, 2020.