OSMANLICA OPTİK KAREKTER TANIMA(OCR) SİSTEMİNİN DERİN SİNİR AĞLARI İLE GELİŞTİRİLMESİ

📌 Amaç ve Kapsam

- **Osmanlıca**, 13. yüzyıldan 20. yüzyıla kadar Osmanlı İmparatorluğu'nda kullanılan bir yazı dilidir.
- Osmanlı arşivlerinde milyonlarca belge bulunmaktadır ve **bunların çok az bir kısmı dijital metne dönüştürülmüştür**.
- Bu çalışmanın amacı, **derin öğrenme tabanlı bir OCR modeli geliştirerek Osmanlıca nesih fontlu basılı belgeleri tanımaktır**.
- Mevcut Osmanlıca OCR sistemleri genellikle yetersiz sonuçlar vermektedir.
- Çalışmada **Convolutional Neural Networks (CNN) ve Long Short-Term Memory (LSTM)** mimarileri birleştirilerek güçlü bir model oluşturulmuştur.

📌 Yöntem

Makale, üç farklı veri kümesi kullanarak derin öğrenme tabanlı OCR modelleri oluşturmuştur:

1. **Orijinal Veri Kümesi: 1.000 sayfa** basılı Osmanlıca belge içerir.
2. **Sentetik Veri Kümesi: 23.000 sayfa** sentetik olarak üretilmiş Osmanlıca belge içerir.
3. **Hibrit Veri Kümesi:** Orijinal ve sentetik verilerin birleşiminden oluşur.

- **Deneysel çalışmalar**, 21 sayfalık bir test seti üzerinde gerçekleştirilmiştir.
- Model, **Google Docs, Abby FineReader, Miletos, Tesseract (Arapça ve Farsça modelleri)** ile karşılaştırılmıştır.
- **Ham, normalize ve bitişik metin seviyelerinde karakter, bağlanmış harf (ligature) ve kelime tanıma doğrulukları hesaplanmıştır**.

📌 Kullanılan Model ve Mimarisi

- **Derin sinir ağı modeli, CNN+LSTM tabanlı bir CRNN (Convolutional Recurrent Neural Network)** mimarisi kullanmaktadır.

- **CNN bölümü**, görüntüden özellikleri çıkarır.
- **LSTM bölümü**, harf dizilerini öğrenir ve karakterlerin doğru sıralanmasını sağlar.
- **CTC (Connectionist Temporal Classification) kaybı**, OCR çıktısının düzeltilmesine yardımcı olur.
- **Model eğitimi için kullanılan hiper parametreler:**
- **Öğrenme oranı (Learning Rate):** 0.002
- **Momentum:** 0.5
- **Epoch:** 3.000.000
- **Veri ön işleme:** Görüntüler, satırlara ve karakterlere bölünerek modelin eğitimi için hazırlanmıştır.

📌 Sonuçlar ve Karşılaştırma

- **Hibrit model**, **%97.37** karakter tanıma doğruluk oranı ile en iyi sonucu vermiştir.
- **Diğer yöntemlerle karşılaştırma:**
- **Google Docs:** %91.43
- **Abby FineReader:** %81.05
- **Tesseract (Arapça):** %81.27
- **Tesseract (Farsça):** %83.48
- **Miletos:** %86.88
- **Osmanlica.com Hibrit Modeli: %97.37 (en yüksek doğruluk oranı)**

📌 Sonuç ve Katkılar

- **Çalışma, Osmanlıca OCR için en yüksek başarıyı sağlayan modeli sunmuştur**.
- Osmanlıca OCR sürecinde **karakter, bağlanmış harf ve kelime seviyesinde analizler yapılmıştır**.
- **Osmanlıca harf, bağlanmış karakter ve kelime sıklıkları analiz edilmiştir**.
- **Çalışmanın sonucunda geliştirilen model Osmanlica.com üzerinden erişilebilir hale getirilmiştir**.
- **Osmanlıca belgelerin dijitalleştirilmesi sürecinde büyük bir katkı sunmaktadır**.

📌 Kullanılan Yöntem ve Model

Makale, **Osmanlıca metinlerin otomatik olarak dijital metne dönüştürülmesi** için **CNN + LSTM** tabanlı bir **OCR modeli** önermektedir. **CRNN (Convolutional Recurrent Neural Network)** adı verilen bu hibrit model, **hem görüntü tabanlı özellikleri** hem de **karakter sıralamasını** öğrenebilen güçlü bir yaklaşımdır.

📌 Yöntemin Adımları

Önerilen yöntem **dört ana aşamadan** oluşmaktadır:

1. **Veri Kümesi Hazırlama ve Ön İşleme**
2. **Derin Öğrenme Modelinin Eğitimi**
3. **OCR Modelinin Çalışması (Inference)**
4. **Sonuçların Normalizasyonu ve Analizi**

1️⃣ Veri Kümesi Hazırlama ve Ön İşleme

Makale kapsamında üç farklı **veri kümesi** oluşturulmuş ve kullanılmıştır:

Veri Kümesi	Açıklama	Sayfa Sayısı
Orijinal Veri	Gerçek Osmanlıca belgelerinden elde edilen görüntüler	1.000
Sentetik Veri	Algoritmik olarak üretilen Osmanlıca metin görüntüleri	23.000
Hibrit Veri	Orijinal + Sentetik verilerin birleşimi	24.000

📍 Veri Ön İşleme Adımları

- **Metin Görüntülerinin Satırlara Bölünmesi:**

OCR modelinin girdi olarak satır bazında çalışması için, Osmanlıca metin görüntüleri **satır seviyesinde bölümlendirildi**.

- **Karakterlerin Ayrıştırılması ve İşaretlenmesi:**

Modelin eğitimi için, her harf etiketlendi ve uygun veri formatına dönüştürüldü.

- **Görüntü Filtreleme ve Gürültü Temizleme:**

OCR modelinin başarısını artırmak için **OpenCV** ve **ImageMagick** gibi kütüphaneler kullanılarak görüntüler iyileştirildi.

- **Veri Normalizasyonu:**

Harf ve kelime seviyesinde **normalize edilmiş** (standartlaştırılmış) metinler oluşturuldu.

2️ Derin Öğrenme Modelinin Eğitimi

📍 Kullanılan Mimari: CRNN (CNN + LSTM + CTC)

Model, **Convolutional Recurrent Neural Network (CRNN)** mimarisi kullanarak **CNN, LSTM ve CTC (Connectionist Temporal Classification) kaybını** birleştiren hibrit bir yapıdır.

🔹 CNN (Convolutional Neural Network) Bölümü:

- Görüntülerden **harf ve karakter özelliklerini çıkarmak için** kullanılır.
- **Özellik Haritaları (Feature Maps)** üreterek harflerin görsel temsillerini yakalar.
- **3×3 evrişim (convolution) filtreleri** kullanılarak kenar ve desen bilgileri çıkarılır.

🔹 LSTM (Long Short-Term Memory) Bölümü:

- **Karakterlerin sıralı bilgilerini öğrenmek için** kullanılır.
- İleri ve geri yönlü **Bidirectional LSTM (BiLSTM)** ile harflerin **önceki ve sonraki karakterlerle ilişkisini** öğrenir.

🔹 CTC (Connectionist Temporal Classification) Kayıp Fonksiyonu:

- Harflerin **sıralamasını belirlemek ve boşlukları yönetmek** için kullanılır.
- Doğrudan çıktı olarak bir kelime veya cümle oluşturmak yerine, **harf dizilerinin olasılıklarını hesaplar**.

🔹 Model Mimarisinin Detayları:

Katman Türü	Açıklama
CNN Katmanları	Görüntü özelliklerini çıkarır
Bidirectional LSTM	Harflerin sırasını öğrenir
Yoğun Katman (Dense Layer)	Karakter tahminleri yapar
CTC Kayıp Fonksiyonu	Doğru karakter dizisini çıkarır

📌 **Bu model, Osmanlıca karakterlerin bitişik yazılması, harf şekillerinin değişkenliği gibi zorlukları başarılı bir şekilde çözmektedir.**

3️⃣ OCR Modelinin Çalışması (Inference)

- Eğitilmiş model, **yeni Osmanlıca metinleri girdi olarak alır** ve **bu görüntüleri metne çevirir**.
- Modelin çıktısı, **ham metin, normalize edilmiş metin ve bitişik harfli metin** olmak üzere üç farklı formatta olabilir.

📌 Çıktılar şu şekilde sınıflandırılır:

1. **Ham Metin (Raw Text):** Doğrudan OCR çıktısı
2. **Normalize Metin:** OCR hataları düzeltilmiş metin
3. **Bitişik Metin:** Harf bağlantılarının korunarak metnin düzeltilmiş hali

4️⃣ Sonuçların Normalizasyonu ve Analizi

- OCR sonuçları **karakter, bağlı harf (ligature) ve kelime doğruluk oranları** ile değerlendirildi.
- **Osmanlı alfabesinin özelliklerine göre** harfler gruplandırıldı:
- **Bitişebilen / Bitişemeyen Harfler**

- **Noktalı / Noktasız Harfler**
- **Harf Gövdesi Tipine Göre Gruplar**
- **Harf, kelime ve bağlanmış harf sıklıkları** analiz edilerek modelin başarısı detaylıca incelendi.

📌 **Deneysel Sonuçlar**

Model	Ham Metin Doğruluğu (%)	Normalize Metin Doğruluğu (%)	Bitişik Metin Doğruluğu (%)
Hibrit Model (Önerilen)	**88.86**	**96.12**	**97.37**
Orijinal Model	87.73	94.87	96.16
Sentetik Model	73.16	77.64	78.10
Google Docs	83.86	92.02	91.43
Abby FineReader	71.98	80.19	81.05
Tesseract (Arapça)	76.92	82.37	81.27
Tesseract (Farsça)	75.30	83.85	83.48
Miletos	75.76	86.46	86.88

📌 **Önerilen model, karakter, kelime ve bağlanmış harf doğruluk oranlarında diğer yöntemlerden belirgin şekilde daha iyi performans göstermiştir.**

📌 **Sonuç ve Gelecek Çalışmalar**

- **Osmanlıca OCR için derin öğrenme tabanlı en iyi model geliştirilmiştir.**
- **Osmanlica.com** üzerinden **kullanıcıların erişimine sunulmuştur**.
- **Gelecek çalışmalar:**
- **Daha büyük veri setleriyle modelin eğitilmesi**
- **Farklı Osmanlıca yazı türleri (divani, talik) için yeni modeller geliştirilmesi**
- **OCR sonrası Osmanlıcadan Türkçeye otomatik çeviri entegrasyonu**