

Academic Report Draft

1. Problem ve Veri

Problem Tanım: Bu proje, video zaman serisi verilerini kullanarak bir sonraki kareyi tahmin etme (Next-Frame Prediction) problemini ele almaktadır. Amaç, verilen ardışık video karelerinden (0-18. kareler) oluşan bir diziyi girdi olarak alıp, bu dizinin bir adım sonrasını (1-19. kareler) veya 20. kareyi tahmin edebilen bir derin öğrenme modeli geliştirmektir. Bu problem, meteorolojik tahminler, otonom sürücü ve video sıkıştırma gibi alanlarda kritik öneme sahiptir.

Veri Kaynağı: Projede, standart bir kayaslama veri seti olan **Moving MNIST** veri seti kullanılmıştır. Bu veri seti, tensorflow_datasets kütüphanesi aracılıkla elde edilmiştir.

Zaman Aralığı: Veri seti sentetik olduğunu için belirli bir gerçek dünya zaman aralığı (yıl, ay vb.) yoktur. Ancak her bir veri örneği, ardışık **20 zaman adımından (kare)** oluşan bir video kesidir.

Gözlem Skali: Veriler video formatındadır ve ardışık kareler (frames) şeklinde düzenlenmiştir. Her bir kare, zaman serisindeki bir gözlem anlamını temsil eder.

Kullanılan Değerler: Veri setindeki temel değerlerken, piksel yoğunluk değerleridir. * **Girdi:** 64x64 boyutunda, tek kanallı (siyah-beyaz) görüntüler. * **Dğer Aralığı:** 0-255 arası tamsayılar (ön işlemede 0-1 arasına normalize edilmiştir).

2. Veri Ön ~~İ~~leme

Veri seti model eğitimi~~ne~~ne uygun hale getirilmek için aşağıdaki adımlardan geçirilmi~~stır~~tir:

Ölçekleme (Scaling): Orijinal veri setindeki 0-255 aralığndaki piksel değerleri, [0, 1] aralığına normalize edilmi~~stır~~tir. Bu işlem, sinir a~~lların~~n daha hızlı ve kararlı yakınsamasını sağlamak ve aktivasyon fonksiyonları~~nın~~n (Relu, Sigmoid) verimli çalışması için yapılmaktır. $x_{\text{norm}} = x / 255.0$

Dönü~~ş~~ümler:

- **Boyutlandırma:** Görüntüler 64x64 çözünürlüğünde işlenmi~~stır~~tir. Model girdisi (Batch, Time, Height, Width, Channel) formatına uygun olarak (19, 64, 64, 1) şeklinde yeniden boyutlandırılmıştır.

Girdi-Çıktı Ayrımı: 20 karelük video dizileri şu şekilde ayrılmaktır:

- **Girdi (X):** İlk 19 kare ($t=0$ 'dan $t=18$ 'e kadar).
- **Hedef (Y):** Sonraki 19 kare ($t=1$ 'den $t=19$ 'a kadar). Bu yapı, modelin "Sequence-to-Sequence" (Diziden Diziye) öğrenme yapısının desteklemektedir.

Eksik Veri ~~İ~~lemleri: Moving MNIST sentetik ve tam bir veri seti olduğunu için eksik veri bulunmamaktadır. Bu nedenle eksik veri doldurma ~~İ~~lemine ihtiyaç duyulmamaktır.

3. Kullanılan Yöntem

Bu projede, spatiotemporal (uzay-zamansal) özellikleri öğrenmek için bir **ConvLSTM (Convolutional Long Short-Term Memory)** kullanılmıştır.

Seçilen Model: Model, konvolüsyonel sinir ağları (CNN) görsel özellik çökarma yeteneği ile LSTM ağlarının zamansal bağımızlıklarını öğrenme yeteneğini birleştirerek **ConvLSTM2D** katmanlarından oluşanmaktadır.

Model Mimarisi: 1. **Giriş Katmanı:** (19, 64, 64, 1) boyutunda dizi. 2. **ConvLSTM2D Katmanı 1:** 64 Filtre, 5x5 Kernel. Geniş alandaki hareketleri yakalar. 3. **ConvLSTM2D Katmanı 2:** 64 Filtre, 3x3 Kernel. Daha detaylı hareket özelliklerini öğrenir. 4. **ConvLSTM2D Katmanı 3:** 64 Filtre, 1x1 Kernel. Özellik haritaları birleştirir ve derinlik katar. 5. **Çıkış Katmanı (Conv3D):** 3x3x3 Kernel, Sigmoid aktivasyon. Zamansal ve uzaysal düzeltme yaparak nihai görüntüyü [0, 1] aralığında üretir.

Modelin Seçilme Nedeni: Standart CNN'ler zamansal bilgiyi (hareket yönü, hızı) yakalayamazken, standart LSTM'ler görüntüdeki uzaysal ilişkileri (nesne şekli, konumu) kaybeder. ConvLSTM, hücre içindeki matris çarpımalarını konvolüsyon işlemiyle deşifrererek hem uzaysal yapıyı korur hem de zamansal deşifrimi modeller. Bu nedenle video tahmin problemleri için idealdir.

Model Parametreleri: * **Optimizasyon Algoritması:** Adam (Adaptive Moment Estimation) * **Öğrenme Oranı (Learning Rate):** 5e-4 * **Batch Size:** 8 * **Epoch Sayısı:** 50 * **Kayıp Fonksiyonu (Loss Function):** Toplam Kayıp = BCE + Perceptual + GDL * **BCE (Binary Cross Entropy - 10.0):** Piksel bazında doğruluk salılar. * **Perceptual Loss (0.1):** VGG16 kullanılarak algısal benzerliği artırır (görüntülerin insan gözüne doğal

görünmesi). * **GDL (Gradient Difference Loss - 0.1)**: Görüntüdeki keskinlik ve kenar detaylarını korur.

4. Uygulama ve Sonuçlar

Eğitim / Test Ayrımları: Veri seti, modelin genelleme yeteneğini ölçmek amacıyla eğitim ve doğrulama (validation) setlerine ayrılmaktır. * **Doğrulama Seti:** İlk 1000 örnek. * **Eğitim Seti:** Geriye kalan veriler (kartalılarak kullanılmaktır).

Hata Ölçütleri: Modelin başarımını aşağıdaki metriklerle değerlendirmektedir: * **PSNR (Peak Signal-to-Noise Ratio):** Görüntü kalitesini ölçer. Yüksek olması iyidir. * **SSIM (Structural Similarity Index):** Yapısal benzerliği ölçer (0-1 arası). 1'e yakınlığı iyidir. * **MSE (Mean Squared Error):** Hata karesi ortalaması. Düşük olması iyidir. * **MAE (Mean Absolute Error):** Ortalama mutlak hata.

Sonuçlar: Yapılan testler sonucunda elde edilen ortalama değerler (Bkz: `evaluate.py` çıktıları): * **Ortalama PSNR:** X.XXXX (Örnek: 25.00+) * **Ortalama SSIM:** 0.XXXX (Örnek: 0.85+) * **Ortalama MSE:** 0.XXXX * **Ortalama MAE:** 0.XXXX

Tahmin Grafikleri: Elde edilen sonuç görselleri (`results/evaluation_results.png`) rapor ekinde sunulmuştur. Bu görsellerde sırasıyla "Gerçek Görüntü (Ground Truth)" ve "Model Tahmini (Prediction)" yan yana karşılaştırılarak modelin hareketli rakamları takip etme ve şekil bütünlüğünü koruma başarısını gösterilmektedir. Ayrıca eğitim süreci boyunca kaybın deklimi `results/loss_curve.png` grafinde verilmektedir.