

# Makine Öğrenmesi Kullanılarak Yapısal Analiz Sürelerinin İyileştirilmesi

## Improving Structural Analysis Duration by Using Machine Learning

### Öğrenci-1

Barış ÖZALP  
İnönü Üniversitesi  
Bilgisayar Mühendisliği  
Malatya, Türkiye  
barisozlp5@gmail.com

### Öğrenci-2

Elif KARTAL  
Eskişehir Teknik Üniversitesi  
İstatistik  
Eskişehir, Türkiye  
elifkartal025@gmail.com

### Öğrenci-3

Fatmanur CEYLAN  
Samsun Üniversitesi  
Yazılım Mühendisliği  
Samsun, Türkiye  
fatmanurceylan.89@gmail.com

### Öğrenci-4

Hilal Su KORKMAZ  
Turgut Özal Üniversitesi  
Bilgisayar Mühendisliği  
Malatya, Türkiye  
hilalsukorkmaz11@gmail.com

### Akademik Danışman

Ali ARI  
İnönü Üniversitesi  
Bilgisayar Mühendisliği  
Malatya, Türkiye  
ali.ari@inonu.edu.tr

### Sanayi Danışmanı-1

Alihan CAMBAZ  
TUSAŞ - Türk Havacılık  
ve Uzay Sanayi A.Ş.  
Ankara, Türkiye  
alihan.cambaz@tai.com.tr

### Sanayi Danışmanı-2

Osman Çağlar BAYSALLI  
TUSAŞ - Türk Havacılık  
ve Uzay Sanayi A.Ş.  
Ankara, Türkiye  
osmancaglar.baysalli@tai.com.tr

**Özetçe—** Bu çalışmada derin öğrenme yöntemleriyle mühendislik süreçlerinde kritik bir rolü olan yapısal analiz sürelerinin iyileştirilmesi amacıyla, geleneksel analiz yöntemlerinin yerini alabilecek hem zamandan hem de kaynaktan tasarruf sağlayacak nitelikte U-net tabanlı üreteç ve patchGAN ayrıştırıcı içeren bir Koşullu Üretken Karşıt Ağlar (cGAN) modeli önerilmiştir. Görsel verilere dayalı olması, farklı projelere kolayca entegre edilmesini sağlarken, simülasyon maliyetlerini düşürerek kaynak tasarrufu sunar. Model performans metrikleriyle değerlendirildiğinde yüksek doğruluk sağladığı gözlemlenmiştir. Algoritma Google Colab üzerinde A100 GPU ile yaklaşık 30 dakikada eğitilmiş, model kaydedildikten sonra her bir girdi için analiz süresini 2-3 saniyeye kadar indirmiştir.

**Anahtar Kelimeler—** Makine öğrenmesi; Derin Öğrenme; koşullu üretken karşıt sinir ağı; stres haritaları; yapısal analiz; görüntü tabanlı tahmin.

**Abstract—** In this study, a deep learning-based Conditional Generative Adversarial Network (cGAN) model, incorporating a U-net-based generator and a PatchGAN discriminator, is proposed to improve the duration of structural analysis, which plays a critical role in engineering processes. Designed as an alternative to traditional analysis methods, this model offers savings in both time and resources. Its reliance on visual data enables easy integration into various projects while reducing simulation costs. When evaluated with performance metrics, the model demonstrated high accuracy. The algorithm was trained on Google Colab using an A100 GPU in approximately 30 minutes, and after the model was saved, it reduced the analysis time for each input to as low as 2–3 seconds.

**Keywords —** Machine learning; Deep Learning; conditional generative adversarial network (cGAN); stress maps; structural analysis; image-based prediction.

## I. GİRİŞ

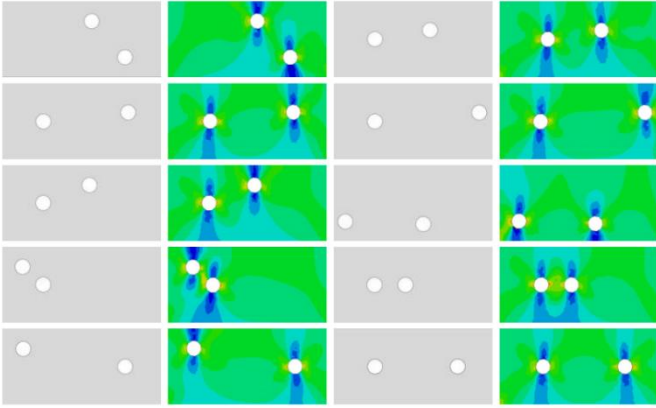
Yapısal analiz, bir yapı üzerindeki yüklerin ve iç kuvvetlerin etkilerinin hesaplanması ve belirlenmesi sürecidir.[1] Bu analiz sayesinde yapıların maruz kaldıkları koşullar altındaki davranışları incelenerek sürdürülebilir yapılar tasarlanmasına katkı sunar. Günümüzde kullanılan yapısal analiz araçları yüksek maliyet, teknik karmaşıklık, uyumluluk, simülasyon süresi uzunluğu başta olmak üzere bir çok dezavantajlı duruma sahiptir. ANSYS ve ABAQUS türevi yazılımların yüksek donanım maliyeti, işlem süresi uzunluğu, yoğun bir eğitim ve deneyim gerektirmesi bu dezavantajlı durumlara örnektir.[2][3]

Bu çalışmada geleneksel yapısal analiz araçlarına alternatif olabilecek derin öğrenmeye dayalı bir yöntem olan Koşullu Üretken Karşıt Ağlar (Conditional Generative Adversarial Network cGAN) modeli önerilmiştir. Model, geleneksel yöntemlere kıyasla daha az işlem ile hızlı ve yüksek doğrulukta tahminler üretebilmiştir.

## II. YÖNTEM

### A. Veri Tanımı

Çalışmada modelin eğitilmesi için boyutları 700mm x 300mm olan delikli demir levha görsellerinin iki boyutlu çekmeye karşı stres dağılımını gösteren 10 girdi-çıkış veri çifti kullanılmıştır. Girdi verisi geometrik yapılarını gösterirken çıktı verisi bu levhanın stres haritasını temsil etmektedir. Veriler ABAQUS programı üzerinden elde edilmiştir.



Şekil 1. Girdi-Çıkış Veri Çiftleri



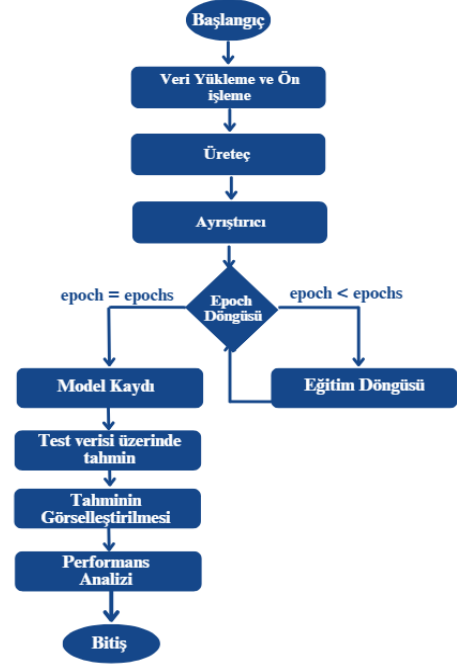
Şekil 2. Test Girdi Verisi

### B. Veri Ön İşleme

Modelin tüm görsel verilere uyumlu sonuç üretebilmesi amacıyla görsel veriler 512x512 piksel boyutlarında yeniden boyutlandırılmıştır (Padding ile görsel kalitesi korunmuştur.) ve piksel değerleri  $[-1,1]$  aralığında normalize edilmiştir. Bu normalizasyon işlemi eğitim kararlılığını artırmaya ve öğrenme performansının iyileşmesine olanak tanır. Veri seti,

TensorFlow'un `tf.data.Dataset` API'si kullanılarak hazırlanmış, giriş-çıkış görüntü çiftleri halinde organize edilmiştir.

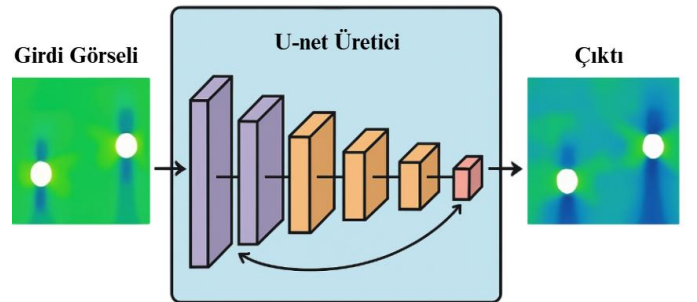
### C. Proje Akışı



Şekil 3. Proje Akış Şeması

### D. Üreteç(Generator) Mimarisi

Modelde generator U-net mimarisi esas alınarak tasarlanmıştır.



Şekil 4. U-net Tabanlı Üreteç Mimarisi

U-Net, kodlayıcı (encoder) ve kod çözücü (decoder) katmanlarının atlama bağlantılarıyla (skip connections) birleştirildiği bir yapıya sahiptir. Bu bağlantılar, ince ayrıntıların korunmasını sağlar ve yukarı örnekleme sırasında bilgi kaybını azaltır.[4]

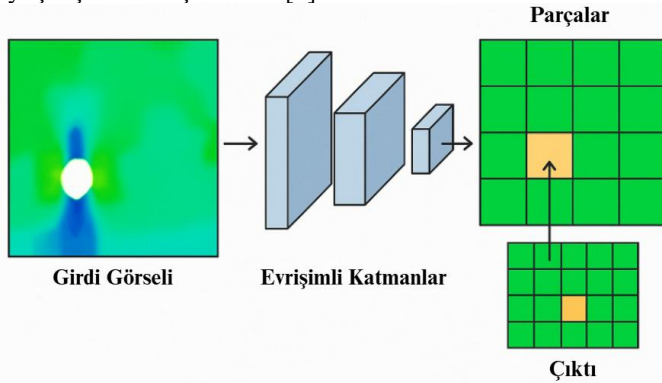
Kodlayıcı, her biri 4x4 filtrelili, 2 birim kaydırmalı ve "same" dolgu kullanan 8 evrişimsel katmandan oluşur. İlk iki blok hariç tutularak üçüncü bloktan itibaren batch normalization uygulanır. *LeakyReLU* aktivasyon fonksiyonu tüm kodlayıcı boyunca kullanılır. Kodlayıcı, uzaysal boyutları kademeli olarak küçültürken özellik haritalarının derinliğini artırır ve hiyerarşik temsiller çıkarır.

Kod çözümler görsel özelliklerinin tekrar orijinal çözünürlüğüne ölçeklendirmek için 7 transpoze evrişim katmanından oluşturulmuştur. Bu katmanlarda *ReLU* aktivasyon fonksiyonları kullanılmıştır ve aşırı öğrenmeyi önlemek için dropout uygulanmıştır. Kodlayıcı katmanlarından gelen atlama bağlantıları detayları korumak için kod çözümler katmanları ile birleştirilir.[5]

Son katman, 3 çıkış kanalına (RGB) sahip bir *Conv2DTranspose* katmanını kullanır ve piksel değerlerini  $[-1, 1]$  aralığında normalize etmek için "*tanh*" aktivasyon fonksiyonu uygular

#### E. Ayırıştırıcı (Discriminator) Mimarisi

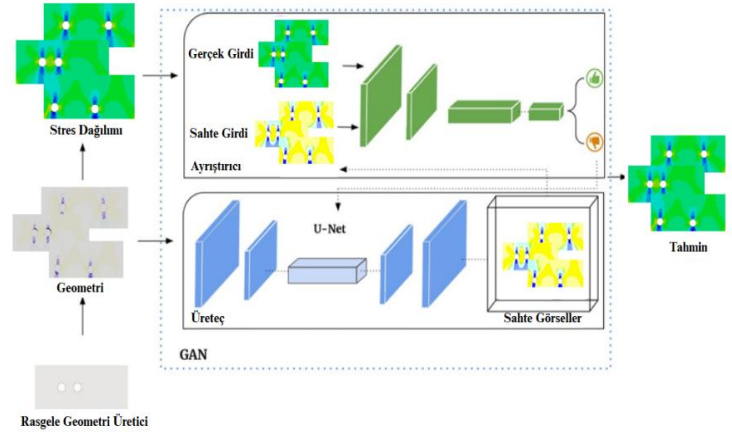
Ayırıştırıcıda görüntü parçalarının (patches) gerçek mi yoksa sahte mi olduğunu sınıflandıran PatchGAN mimarisi kullanılmıştır. Bu yaklaşım tüm görüntüyü bir kerede sınıflandırmaz, detaylara odaklanır. Bu sayede üreticinin daha iyi çalışmasını teşvik eder.[6]



Şekil 5. PatchGAN Tabanlı Ayırıştırıcı Mimarisi

#### F. Kayıp Fonksiyonları ve Optimizasyon

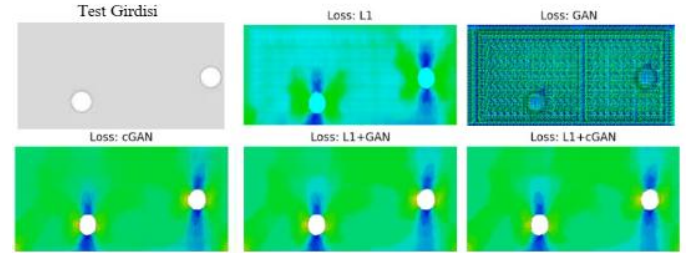
**Üreteç Kaybı** Adversarial kayıp, Üreteç'in Ayırıştırıcıyı kandırarak gerçekçi görüntüler üretmesini teşvik eder. **L1 kaybı**, üretilen ve hedef görüntüler arasındaki ortalama mutlak farkı ölçerek yapısal benzerliği ve piksel düzeyinde doğruluğu artırır. Bu kayıp, gerçekliğe olan bağlılığı vurgulamak için 50 ile ağırlıklandırılmıştır.



Şekil 6. Üreteç ve Ayırıştırıcının Birlikte Çalışma Prensipleri

Üreteç kaybı için farklı kayıp fonksiyonları denenmiş, en iyi performans veren fonksiyonun "*L1+cGAN*" olduğuna karar verilmiştir. Değerlendirme süreci aşağıdaki gibidir:

#### KAYIP FONKSİYONLARI



Şekil 7. Kayıp Fonksiyonlarının Değerlendirmesi

Eğitim ve değerlendirme sırasında, üretilen görüntülerin kalitesini sayısal olarak değerlendirmek için farklı metrikler kullanılmıştır. Detayları "*Performans Analizi*" başlığı altında incelenecektir.

Ayırıştırıcı Kaybı, gerçek görüntü sınıflandırma kaybı ile sahte görüntü sınıflandırma kaybının toplamıdır ve her iki kayıp da ikili çapraz entropi kullanılarak hesaplanır. Ayırıştırıcı, gerçek ve üretilmiş görüntü çiftlerini doğru bir şekilde ayırt etmeyi amaçlar.[6]

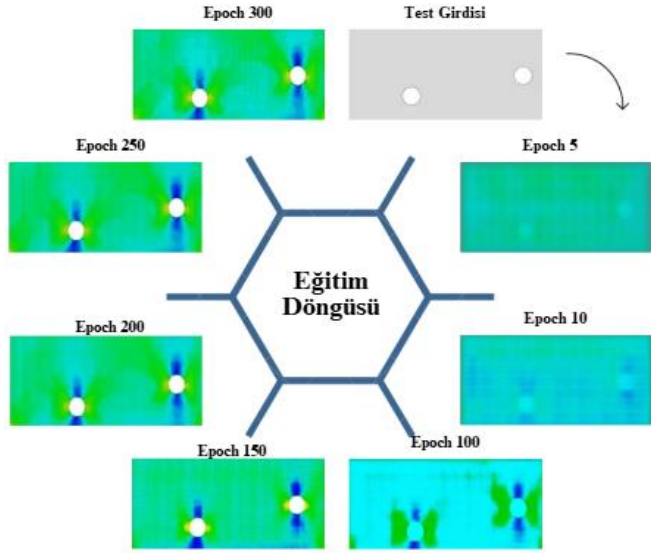
#### G. Eğitim Süreci

Model, toplam 300 epoch boyunca 1 batch size ile eğitilmiştir. Her epoch sırasında giriş görüntüleri, Üreteç (Generator) aracılığıyla sahte çıktılar oluşturmak için işlenmiş ve bu çıktılar, gerçek görüntülerle birlikte Ayırıştırıcı (Discriminator) tarafından değerlendirilmiştir. Adversarial eğitim düzeni, üreticinin çıktılarının ayırıştırıcıyı yanıltacak şekilde sürekli olarak iyileştirilmesini sağlamış ve giderek daha gerçekçi görüntülerin üretilmesini sağlamıştır.

Performans metriklerinin sonuçlarına bakılarak 300. epochtan sonra anlamlı bir artış gözlenmediğinden aşırı öğrenmeyi önleyebilmek amacıyla eğitim 300. epochta sonlandırılmıştır.

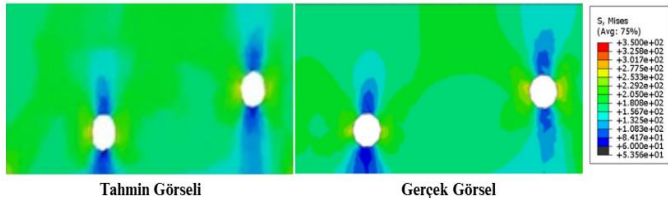
## H. Uygulama ve Sonuçlar

Model eğitim sırasında epoch bazında kaydedilen görüntüler aşağıdaki gibidir.



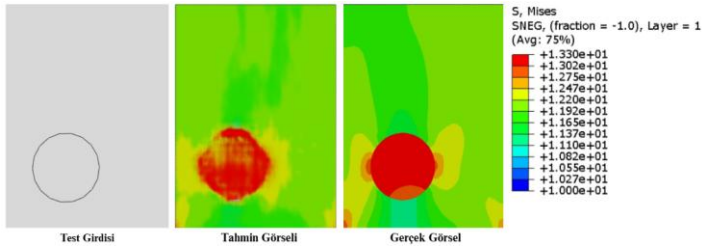
Şekil 8. Epoch Bazında Kaydedilen Görseller

Model eğitim döngüsünde epoch ilerledikçe modelin performansının arttığını ve anlamlı bir eğitim süreci geçirdiği gözlemlenmiştir. Eğitim sonucu elde edilen tahmin görüntüsü Şekil 9. da gösterilmiştir.



Şekil 9. Epoch 300 de Üretilmiş Tahmin Görseli ile Gerçek Görselin Karşılaştırılması

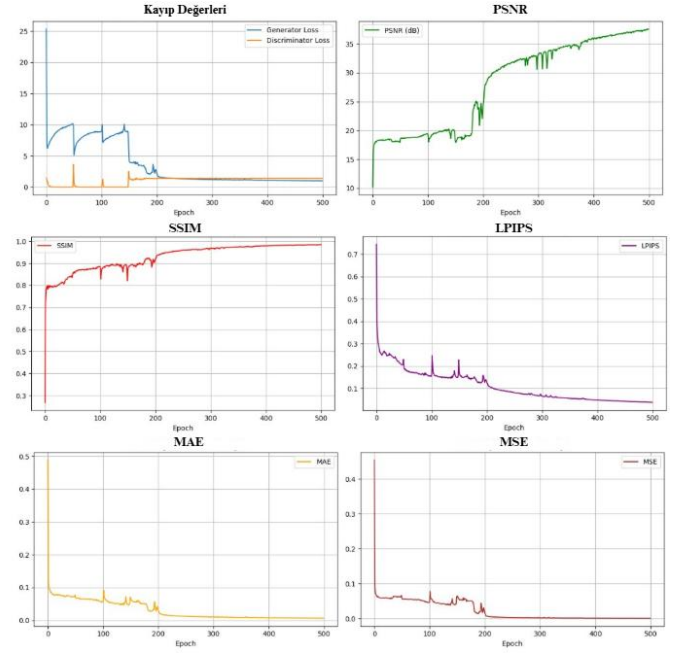
## I. Anizotropik Yapılarda Uygulama



Şekil 10. Anizotropik Yapılarda Uygulama

Model boyutları ve geometrik özellikleri farklı olan 10 girdi-çıkıttı veri çifti ile kompozit levhanın yapısal analizi için çalıştırıldığında sonuç Şekil 10. da gösterilmiştir.

## J. Performans Analizi



Şekil 11. Performans Grafikleri

**Kayıp Grafiği:** Bu grafik, eğitim sürecindeki üretic ve ayrıştırıcı kayıplarını göstermektedir. Başta üretic kaybı yüksektir, bu da modelin öğrenmenin ilk aşamalarında olduğunu gösterir. Yaklaşık 100. epoch civarında düşük seviyelere iner, bu da üretic ve ayrıştırıcının performansının iyileştiğini ve dengelendiğini gösterir.

**SSIM Grafiği:** Yapısal Benzerlik İndeksi (SSIM), model çıktıları ile orijinal görüntüler arasındaki yapısal benzerliği ölçer. Grafik, SSIM'in hızla arttığını ve eğitim süreci boyunca 0.9'un üzerinde sabitlendiğini görülmektedir.

**PSNR Grafiği:** En Yüksek Sinyal-Gürültü Oranı (PSNR), model tahminlerinin kalitesini ölçer. Eğitim süreci boyunca PSNR'de belirgin bir artış eğilimi gözlemlenmiştir, bu da model çıktıların giderek daha az hatalı ve daha yüksek kaliteli hale geldiğini göstermektedir.

**LPIPS Grafiği:** Öğrenilmiş Algısal Görüntü Parça Benzerliği (LPIPS), tahmin edilen görüntülerin algısal benzerliğini ölçer. Grafik, eğitim sırasında LPIPS değerlerinde kademeli bir azalma olduğunu gösterir; bu da model çıktıların orijinal görüntülere giderek daha fazla görsel olarak benzediğini ifade eder.

**MSE ve MAE Grafikleri:** Grafikler, modelin eğitim sürecindeki hata oranlarının nasıl değiştiğini göstermektedir. Mean Absolute Error (MAE) ve Mean Squared Error (MSE) başlangıçta yüksek değerlerle başlarken, eğitim ilerledikçe öğrenme sürecinin anlamlı ilerlediğini görülmektedir. MAE'nin azalması, modelin ortalama hata oranını düşürdüğünü, MSE'nin azalması ise büyük sapmaların minimize edildiğini ifade eder.

### III. SONUÇ

### KAYNAKLAR

Bu çalışmada, Python dilinde 2D levhaların stress haritalarını tahmin edebilen bir CGAN modeli geliştirilmiştir. Eğitim süreci, **Google Colab** üzerinde 10 veri çifti ile 300 epoch ve 1 batch boyutuyla gerçekleştirilmiştir. Eğitim süresi GPU türüne bağlı olarak değişmekle birlikte, **A100 GPU**'da yaklaşık 30 dakikada tamamlanmıştır. Model kaydedildikten sonra, her giriş için 2-3 saniye içinde stres dağılımı tahmini yapabilecek bir hızda çalışmaktadır. Eğitim sürecinin önemli bir yönü, modelin yalnızca görsel verilerden öğrenmesi ve görüntülerin yapısal detayları sağlanmadan çalışmasıdır. Ayrıca, modelin farklı veri kümeleriyle doğru tahminler yapabildiği gözlemlenmiştir. Bu özellikler, modelin çeşitli veri türleriyle verimli ve esnek bir şekilde çalışabilme potansiyelini ortaya koymaktadır.

#### A. Proje Ekibinin Kazanımları ve TUSAŞ'a Olası Katkısı

Proje yapısal analiz süreçlerinde hız, maliyet avantajı ve esneklik kazandırma potansiyeline sahiptir. Geleneksel yöntemlere kıyasla daha hızlı sonuç üretmesi, prototip geliştirme süreçlerini hızlandırırken tasarım optimizasyonuna katkı sağlar. Görsel verilere dayalı olması, farklı projelere kolayca entegre edilmesini sağlarken, simülasyon maliyetlerini düşürerek kaynak tasarrufu sunar.

Proje ekibi, yapay zeka, derin öğrenme ve görüntü tabanlı analiz yöntemleri konusunda uzmanlık kazanmış, U-Net, PatchGAN gibi mimarilerin mühendislik uygulamalarında entegrasyonu üzerine deneyim edinmiştir. Bu kazanımlar, hem TUSAŞ'ın teknolojik altyapısına katkı sağlamakta hem de yenilikçi çözümlerin geliştirilmesine zemin hazırlamaktadır.

#### B. Öneriler, Alınan Dersler ve Gelecek Çalışmalar

Başlangıçta, sabit 256x512 boyutlarında modeli eğitildi ve model tüm görsel boyutlarına uyum sağlayamadığı için düşük performans gösterdi. Modeli tüm boyutlara uyumlu hale getirmek amacıyla düzenlemeler yapıldı. Çalışmada 10 veri çiftiyle çalışıldı. Daha fazla veri ile sonuçların iyileşeceği düşünülmektedir. İlerleyen dönemlerde modelin bir arayüze entegre edilmesi planlanmaktadır. Model malzeme tanıma veya yapı optimizasyonu süreçleri için geliştirilmeye açıktır ve sadece yapısal analiz alanında değil diğer görüntü tahmin projeleri için kullanıma uygundur.

İlerleyen dönemlerde modelin 3D analizlerde de kullanılabilmesi amacıyla geliştirilmesi planlanmaktadır.

### TEŞEKKÜR

*Proje süreci boyunca bilgi, deneyim ve tecrübelerini bizden esirgemeyen sanayi danışmanlarımız Sayın Alihan Cambaz'a ve Sayın Osman Çağlar Baysallı'ya teşekkürlerimizi sunarız.*

- [1] TECHNIA. (n.d.). What is structural analysis? Retrieved from <https://www.technia.com/en/structural-analysis/>
- [2] Sherif M. Jou rna IP. Econ Lett [Internet]. 2020;109008. Available from: <https://doi.org/10.1016/j.econlet.2020.109008>
- [3] Wang X, Zhang B, Xu Z, Qin Y, Philip Chen CL. An exploration of deep learning for decision making: Methods, applications, and challenges. Expert Syst Appl [Internet]. 2025;289(May):128406. Available from: <https://doi.org/10.1016/j.eswa.2025.128406>
- [4] Yang Z. Application of Evolutionary Deep Learning Algorithm in Construction Engineering Management System. Syst Soft Comput [Internet]. 2025;200317. Available from: <https://doi.org/10.1016/j.sasc.2025.200317>
- [5] Yuan J, Wu Q, Zhou J, Yu S, Xin X, Liu J, et al. A Review of Deep Learning Models for Food Flavor Data Analysis. J Futur Foods [Internet]. 2025; Available from: <https://doi.org/10.1016/j.jfutfo.2025.01.001>
- [6] Yang F, Xu Q, Chen F. Active learning deep autoencoder model with importance sampling for reliability analysis. Appl Soft Comput [Internet]. 2025;180(June):113428. Available from: <https://doi.org/10.1016/j.asoc.2025.113428>