

# Derin Öğrenme ile Kumaş Hatası Tespiti Fabric Defect Detection using Deep Learning

Abdulkadir Şeker<sup>1</sup>, Kadir Aşkın Peker<sup>2</sup>, Ahmet Gürkan Yüksek<sup>1</sup>, Emre Delibaş<sup>1</sup>

<sup>1</sup>Bilgisayar Mühendisliği Bölümü, Cumhuriyet Üniversitesi, Sivas, Türkiye  
{aseker, agyukse, edelibas}@cumhuriyet.edu.tr

<sup>2</sup>Bilgisayar Mühendisliği Bölümü, Melikşah Üniversitesi, Kayseri, Türkiye  
{kpeker}@meliksah.edu.tr

**Özetçe** —Kumaş hatası tespiti sektörel nitelik açısından önem arz etmektedir. Hata tespitinde otomatik sistemler geliştirilmiş, bu bağlamda görüntü işleme çalışmalarıyla yüksek hassasiyet elde etmek için birçok yöntem denenmiştir. Bu çalışmada da çok katmanlı mimarisi ile öne çıkan ve yüksek başarı ortaya koyan derin öğrenme, kumaş hatası tespitinde uygulanmıştır. Derin öğrenme algoritmalarında, giriş verisini sıkıştırarak ve genişleterek temsil etmeyi hedefleyen autoencoder, kumaş hatası tespitinde denenmiş, kabul edilebilir oranda başarı elde edilmiştir. Çalışmanın asıl hedefi autoencoder yönteminin giriş değerleri ve hiper parametreleri üzerinde ayarlamalar yaparak, öznelik çıkarım başarısını artırmaktır.

**Anahtar Kelimeler**—derin öğrenme, kumaş hatası tespiti, autoencoder, özellik çıkarımı.

**Abstract**—Fabric defect detection have importance in terms of sectoral quality. Automatic systems are developed on the defect detection, in this regard many methods are tried to obtain high precision with image processing studies. In this study, deep learning which distinguishes with multi-layer architectures and reveals high achievement is applied to fabric defect detection. Autoencoder -a deep learning algorithm- that aimed to represent input data via compression or decompression is tried to detect defect of fabrics and it gains acceptable success. The vital goal of this study is to increase achievement of feature extraction by tuning up the autoencoder's input value and hyper parameters.

**Keywords**—deep learning, fabric defect detection, autoencoder, feature extraction.

## I. GİRİŞ

Tekstil endüstrisinde kumaş hatalarının tespiti önemli bir kalite kontrol aşamasıdır. Bu hataların tespitinde, gelişen pazar hacmi ve üretim kapasitelerinin büyüklüğü sebebiyle insan görüşü ile tespit, büyük oranda zaman kaybına ve hata tespit oranının %70'lere düşmesine sebep olmaktadır [1]. Bundan dolayı görüntü işleme tabanlı bilgisayarlı hata tespit sistemleri sektörde kullanılmaya başlamıştır.

Kumaş hatası tespitinde daha yüksek başarımlar elde edebilmek için görüntü işleme alanında bir çok algoritma denenmiş veya uygulanmıştır.

Kumaşın kendine has bir dokusunun olması, diğer resim/görüntü türlerinden ayrı olarak incelenerek özneliklerinin çıkarılmasını gerektirmektedir. Öznelikler sınıflandırma problemlerinin ham maddesidir. Hata tespitinde

de en önemli aşama, doğru özneliklerin çıkarılmasıdır. Bu amaç doğrultusunda, kendi özneliklerini çıkaran, son yıllarda ses ve görüntü işleme alanında oldukça sık kullanılmaya başlayan derin öğrenme algoritmaları bu çalışmada öznelik çıkarımı ve hata tespiti için kullanılmıştır. Çalışmada derin ağına hiper parametrelerinin ayarlanması ile, optimum değerlerin tespiti üzerinde durulmuştur.

## II. DERİN ÖĞRENME ALGORİTMALARI

### A. Derin Öğrenme

2006 yılında makine öğrenmesi alanının öncülerinden Toronto Üniversitesi'nde Prof. Geoffrey Hinton derin öğrenme modelini ortaya atana kadar, yapay sinir ağlarında gizli katman sayısının artırılması, ağına derinleştirilmesi hesaplama işlemlerinde çıkmaza sebep olduğundan çok tercih edilmiyor, daha çok sığ ağlar kullanılıyordu [2]. Bu gelişmeden sonra ortaya çıkan derin öğrenme algoritmaları, yapay sinir ağları kökenli ve enerji tabanlı modellerinden oluşmaktadır [3]. Mimarisi birçok katman ve saklı değişkenden meydana gelir. En çok kullanılan algoritmalar Derin Yapay Sinir Ağları (Deep Neural Networks), Otomatik Kodlayıcılar (Autoencoders) ve Boltzmann Makinelerinin bir türevi olan Kısıtlı Boltzmann Makineleridir (Restricted Boltzmann Machines). Derin öğrenme verinin doğrusal olmayan dönüşümünü sağlamakta, sığ yapılar yerine çok katmanlı yapı ile karmaşık ilişkileri modelleyebilmektedir. Derin öğrenme modeli, özneliklerin öğrenilmesinde müthiş avantajlara sahiptir. Bu model tarafından öznelik öğrenmesiyle veri daha zengin bilgilerle temsil edilebilir, bu da sınıflandırma performansını artıracaktır [4].

Derin öğrenme yaklaşımları nesne tanıma, konuşma ve doğal dil işleme, sinyal işleme, hata tespiti gibi farklı alanlarda başarıyla kullanılmaktadır. MNIST el yazısı veri kümesi üzerinden yapılan rakam sınıflandırılmasında %1'lere kadar düşen hata oranı elde edilen başarılarından biridir. 2010 yılından beri her yıl düzenlenen IMAGENET yarışmasında geçen yıl birinci olan Microsoft ekibi ve büyük çoğunluktaki yarışma grupları derin öğrenme modellerini kullanmışlardır [5].

Bunun haricinde yine birçok çalışma yapılmıştır. Derin öğrenme kapsamında Stanford Üniversitesi'nde yapılan bir çalışmada resimden nesne tanıma ile doğal dil işleme sinir ağları birleştirilmiş ve bu sayede Şekil 1'deki nesneyi kuş olarak sınıflandırmayla kalmayıp resimdeki diğer nesnelerle ilişkisini ortaya koymuştur. Sonuç olarak "Bir ağaç dalına tünemiş kuş" yorumunu çıkarmıştır [6].

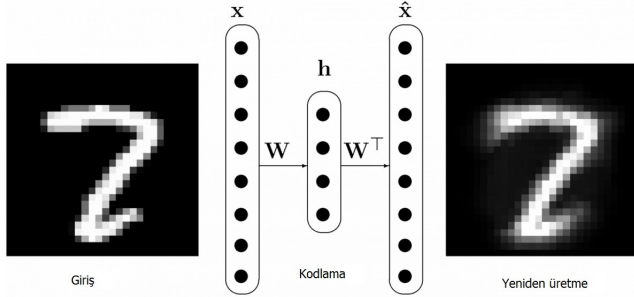


Şekil 1: Derin Ağ Yorumu: "Bir ağaç dalına tünemiş kuş"

Bu çalışmaların yanında piyasadaki büyük verilere sahip şirket ve kurumlar da derin öğrenme alanına yönelmişlerdir. DARPA, 2009 yılında derin öğrenme çalışmalarına destek vermeye başlamıştır. Google son dönemde bünyesine kattığı, Deep Mind firması ile yürüttüğü derin öğrenme çalışmaları kapsamında, sahip olduğu ve sürekli artan veri havuzunu zamanı geldiğinde geliştirdiği algoritmalar ile kullanarak akıllı sistemler ortaya çıkarabileceğini göstermiştir [7].

#### B. Autoencoder

Derin öğrenme tekniklerinden birisi de autoencoder'lardır. Autoencoder, kendi girişini çıkışında tekrar etmeye çalışan bir sinir ağıdır. Bu yüzden giriş ve çıkıştaki node sayıları aynı olacaktır. Gizli katmandaki nöron sayısı giriş node sayısından az olduğunda, Autoencoder sıkıştırılmış bir giriş temsili oluşturacaktır (Şekil 2). Autoencoder yönteminde, sisteme göre sıkıştırma veya genişletme işlemi yapılmaktadır.



Şekil 2: Autoencoder çalışma prensibi

Autoencoder iki aşama içermektedir. [8]

- Kodlayıcı: Lineer olmayan haritalama fonksiyonu  $f$ , gizli katmanda ( $h \in R^n$ ) temsil etmek için giriş verisini ( $x \in R^m$ ) haritalar.

$$h = f(x) = s_f(Wx + b) \quad (1)$$

Parametreler  $m \times n(W)$  boyutlu bir kodlayıcı ağırlık matrisini ve bir bias vektörü içerir ( $b \in R^m$ ).

- Kod-çözücü: Lineer olmayan haritalama fonksiyonu  $g$ , gizli katmandaki ( $h \in R^n$ ) temsilden ( $r \in R^m$ ) formundaki giriş verisini yeniden üretir.

$$r = g(h) = s_g(W'h + b') \quad (2)$$

Burada  $W$   $m \times n(W)$  boyutlu kod-çözücünün ağırlık matrisi,  $b \in R^n$  bias vektörüdür.

$s_g$  ve  $s_f$  genellikle  $\text{sigmoid}(z) = 1/(1 + e^{-z})$  formunda kullanılan lineer olmayan aktivasyon fonksiyonlarıdır.

### III. UYGULAMA

#### A. Veri Kümesi

Bu çalışmanın uygulama aşamasında, çok sayıda hatalı kumaş örnekleri bulunamadığından, 10 hatalı kumaştan elde edilen yüksek çözünürlüklü (4000x6000) fotoğraftan rastgele alınan pencerelerle veri kümesi oluşturulmuştur. İlk denemelerde 1000, sonrasında 5000 örnek ile eğitimler yapılmıştır.



Şekil 3: Yırtık hatası bulunan kaynak fotoğraf

#### B. Parametreler

Kaynak resimlerden alınan pencerelerin boyutu, kaynak resmin yeniden boyutlandırma parametresi, autoencoder için uygun giriş sayısı gibi faktörlerden dolayı, farklı veri kümeleri oluşturulmuştur.

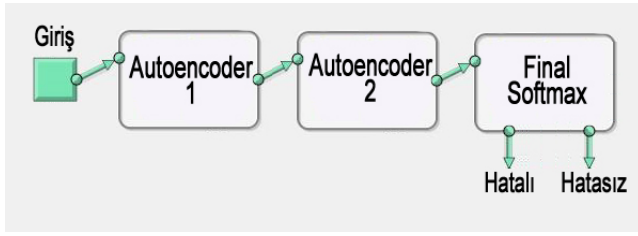
Autoencoder yapısı içindeki hiper parametre olarak da adlandırılan giriş düğüm sayısı, gizli katman düğüm sayısı, iterasyon sayısı, ağırlık katsayısı gibi parametreler de göz önüne alınarak uygulamalar yapılmıştır.

#### C. Derin Ağ Yapısı ve Eğitim

Kumaş hata tespiti bir derin öğrenme yapısı ile uygulanmıştır. Çoklu gizli katman ile kurulan bu sinir ağı, kumaş gibi karmaşık veriye sahip sınıflandırma problemlerinde iyi sonuç vereceği düşünülmüştür. Bu tip sinir ağları eğitmek için bir yöntem, aynı anda sadece bir katman eğitmektir. Bunu yukarıda da anlatılan autoencoder ile yapmak mümkündür.

Derin ağ oluşturulurken ilk olarak iki autoencoder ile denetimsiz (unsupervised) şekilde ağ eğitilmiştir. Bu ağlardan çıkarılan öznetelikler kullanılarak Autoencoder aksine etiketlenmiş veri üzerinden denetimli (supervised) olarak bir final-softmax katmanı ile sınıflandırma yapılmıştır (Şekil 4). Final-softmax katmanında çıkış olarak 2 sınıf için 0-1 arasında değerler üretilmekte, çıkan değerlerden yüksek olan ateşlenerek, sınıflandırmayı yapmaktadır.

Bu çalışmanın amacı kumaş hatasını tespit etmekten çok, kendine özgü bir dokusu olan kumaşların özneteliklerini doğru çıkarmaya çalışmak, autoencoder için optimum çalışma parametrelerini oluşturabilmektir. Bir başka ifade ile hata



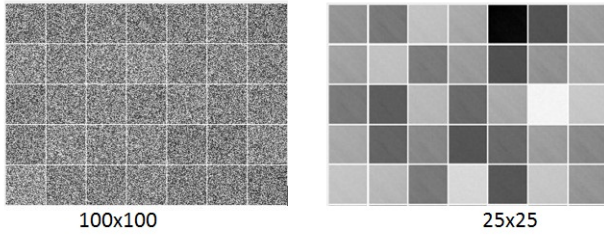
Şekil 4: Derin ağ mimarisi

tespitinden önce sistemin doğru parametrelerle çalıştırıldığından emin olmak hedeflenmiştir. Bu amaçlar doğrultusunda veri kümeleri ve autoencoder parametreleri ile oynamalar yapılarak aşağıdaki sonuçlar üretilmiştir.

#### IV. SONUÇLAR

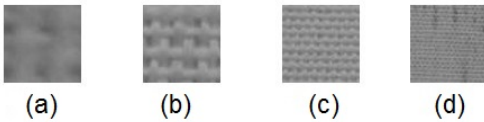
##### A. Autoencoder Eğitimi

Yapılan testler ilk olarak 100x100'lük pencerelerle başlamıştır. Birinci autoencoder çıktısı olan ağırlıklar görselleştirildiğinde ayırt edici özneliklerin çıkması beklenmektedir. Eğitimler gizli katman sayısı (1000-100 arası) ve iterasyon sayısı (100-800 arası) değiştirilerek yapılmıştır. Ağırlıklar her seferinde gürültü benzeri değerler olarak görülmüştür ve eğitim gerçekleşmemiştir.



Şekil 5: Ağırlıkların görselleştirilmesi

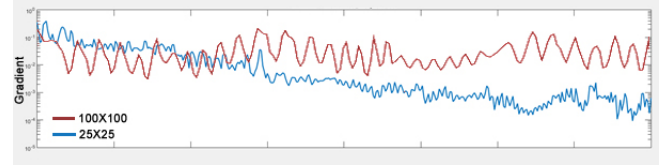
Bu gürültünün ve öğrenme gerçekleşmemesinin sebebi olarak 10000 değerli bir girişin 100 değere kadar sıkıştırılmasının mümkün olmadığı, giriş düğüm sayısının dolayısıyla resmin çözünürlüğünün düşürülmesi gerektiği kanısına varılmıştır.



Şekil 6: Yeniden boyutlandırılan kaynaktan alınan pencereler

Bu sonuçlardan yola çıkarak çözünürlük 25x25'e düşürülmüştür. Ancak kaynak resmin çözünürlüğü yüksek olduğundan, alınan pencereler kumaşın dokusunu ve hatayı ifade edememektedir (Şekil 6-(a)). Kaynak resim yeniden boyutlandırılarak yeni örnekler elde edilmiştir.

Yukarıdaki Şekil 6 (b) ve (c) örnekleri sırasıyla 0.5, 0.2 parametresi ile yeniden boyutlandırılmış kaynaktan alınmıştır. (c) türündeki 2000 örnek ile eğitim tekrarlanmış, ağırlıkların ayırt ediciliğinin arttığı (Şekil-5), türev grafiğinde azalma gerçekleştiği görülmüştür (Şekil-7). Yapılan eğitimler de gizli katman sayısı 100 olarak seçilmiştir.

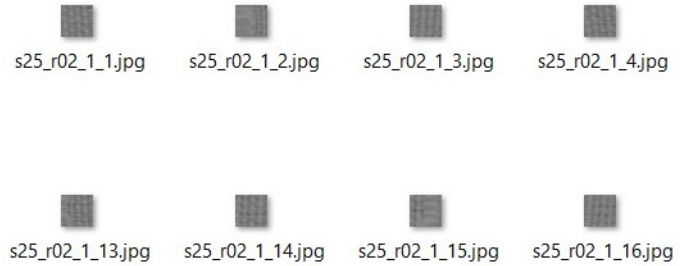


Şekil 7: İki farklı giriş değeri ile türev grafiği

Bu yapılan testler sonucunda autoencoder için kabul edilebilir giriş değerinin 25x25 çözünürlükte (625 giriş düğümü) olması öznelik çıkarımı açısından verimli olduğu anlaşılmaktadır.

##### B. Denetimli Öğrenme Aşaması

Autoencoder çalışmasında 25x25 çözünürlük verimli sonuçlar üretmesine karşın bir problemi de birlikte getirmiştir. Derin ağı bütün olarak çalışması için iki autoencoder arkasından, etiketlenmiş veriler üzerinde bir final-softmax katmanı çalışması gerekmektedir. Bu sebepten elde edilen pencerelerin, bir şekilde hatalı-hatasız olmak üzere iki sınıfa ayrılması gerekmektedir. Pencerelerin boyutunun küçülmesi, 5000 örneğin elle(manuel) sınıflandırılmasını imkansız hale getirmiştir.



Şekil 8: 25x25'lik pencerelerin görünümü

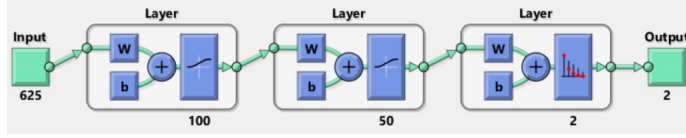
Bu durumu çözebilmek için, pencereler kaynak resimden alınırken 100x100, 200x200 gibi boyutlarda alınacak, sisteme girdi olarak verilmeden önce yeniden boyutlandırılarak 25x25 lik pencereler dönüştürülecektir. Bu aşamada pencerelerin kumaşın dokusunu kaybetmeyecek şekilde yeniden boyutlandırılması gerekmektedir.

Yazılan bir script ile kaynak resmin ve alınan pencerelerin yeniden boyutlandırılmasını parametre olarak, 100x100, 200x200, 300x300, 400x400 boyutlardaki pencereler; 1, 0.8, 0.6, 0.4, 0.2 değerleriyle yeniden boyutlandırılan kaynak resimlerden alınarak toplamda 20 farklı veri kümesi oluşturulmuştur. Bu veri kümelerinden test amaçlı alınan pencereler 25x25 çözünürlüğe getirilmiş, kumaş dokusunun ayırt ediciliğine göre seçilen veri kümesi ile denetimli (supervised) öğrenme aşamasına geçilmiştir. Yapılan testler 100x100 pencerelerin 0.2

parametre ile yeniden boyutlandırılmasından çıkarılan örneklerle yapılmıştır.

Eğitim aşamasında 153 hatalı olmak üzere 1000 örnek ile yapılmıştır. Testler için eğitimde kullanılmayan pencerelerden 40 hatalı, 60 hatasız 100 örnek kullanılmıştır.

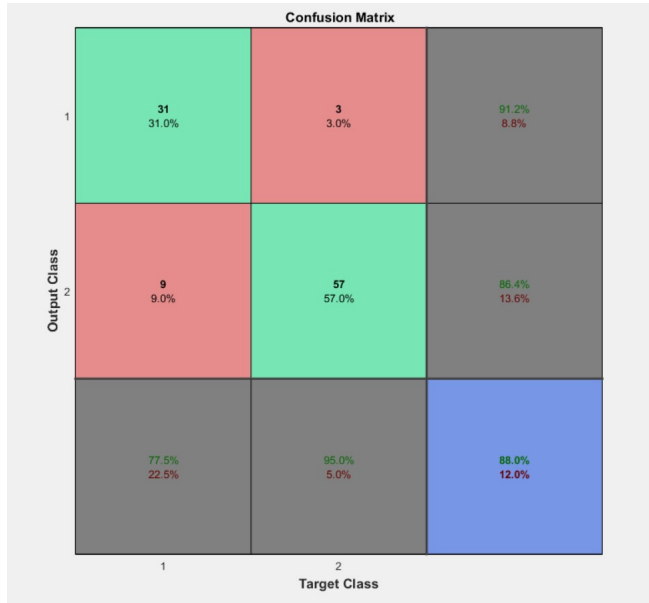
### C. Test Sonuçları



Şekil 9: Kumaş hatası tespit derin ağı

Derin ağ sırasıyla 100, 50 gizli katmandan oluşan iki autoencoder ile denetimli sınıflandırma yapacak final softmax katmanından oluşturulmuştur (Şekil-9).

Etiketlenmiş verilerin sınıflandırılmasını test etmek için bir hata matrisi oluşturulmuştur. Bu matrisin sağ alt köşesinde toplam doğruluk (overall accuracy) oranı verilmektedir.



Şekil 10: Test sonucunda çıkan hata (confusion) matrisi

Toplam doğrulukta %88 oranında başarı sağlanmıştır (Şekil-10). Matriste 1 numaralı sınıf hatalı, 2 numaralı sınıf hatasız örnekleri temsil etmektedir. Matristen çıkan sonuçlarda 40 hatalı örnekten, 9 tanesi hatasız; 60 hatasız örnekten ise 3 tanesi hatalı olarak işaretlenmiştir.

Yapılan çalışmanın bir başka sonucu olarak da kaynak fotoğraflardan ayrı ayrı oluşturulan veri setlerinde derin ağ çalıştırılmış, yırtık ve leke türünde hata tespiti daha yüksek doğrulukta gerçekleşirken, kaçık hatalarında bu oran düşük çıkmaktadır. Bunun sebebi ise autoencoder için giriş resmi sıkıştırıldığında, örneklerdeki kaçık hataları yok olduğu görülmüştür. (Şekil-11)



Şekil 11: kumaş hata örnekleri

Yapılan çalışmalarda hatalı örnek sayısı kısıtlı olduğundan, çalışmanın devamında daha çok sayıda ve farklı türde örneklerle testler yapılması planlanmaktadır. Bir başka planlanan çalışma ise örnekleri hatalı / hatasız olarak iki sınıfa ayırmak yerine; yırtık hatası, kaçık hatası, leke gibi hataları türlerine göre sınıflandırmayı sağlamaktır.

### KAYNAKÇA

- [1] C. Chan, G.K.H. Pang, "Fabric defect detection by Fourier analysis", IEEE Transactions on Industry Applications, 2000.
- [2] D. Hernandez, "Meet the Man Google Hired to Make AI a Reality", <http://www.wired.com/2014/01/geoffrey-hinton-deep-learning>, 2014.
- [3] J. Schmidhuber, "Deep learning in neural networks: An overview", TNeural Networks, 2015.
- [4] D. Hongli, Y. Tao, G. Jiangjin, "An Anomaly Detection Method Based On Deep Learning", The Open Automation and Control Systems Journal, 2015.
- [5] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition", Computer Vision and Pattern Recognition, CoRR, 2015.
- [6] A. Karpathy, L. Fei-Fei, "Deep Visual-Semantic Alignments for Generating Image Descriptions", Computer Vision and Pattern Recognition (CVPR), 2015.
- [7] F. Kurt "Derin Öğrenme (Deep Learning) Nedir?", <http://www.derinogrenme.com/2015/07/21/derin-ogrenme-deep-learning-nedir/>, 2015.
- [8] S. Rifai, P. Vincent, X. Muller, X. Glorot, Y. Bengio "Contracting auto-encoders: Explicit invariance during feature extraction", In Proceedings of the Twenty-eight International Conference on Machine Learning, 2011.