

BASKI DEVRE KARTLARINDA LEHİM HATASI DENETİMİ

**SOLDER JOINT INSPECTION ON PRINTED CIRCUIT
BOARDS**

FURKAN ÜLGER

DOÇ. DR. SENİHA ESEN YÜKSEL

Tez Danışmanı

Hacettepe Üniversitesi

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin

Elektrik ve Elektronik Mühendisliği Anabilim Dalı için Öngördüğü

YÜKSEK LİSANS TEZİ olarak hazırlanmıştır.

Mayıs 2022

ÖZET

BASKI DEVRE KARTLARINDA LEHİM HATASI DENETİMİ

Furkan Ülger

Yüksek Lisans Tezi, Elektrik ve Elektronik Mühendisliği

Danışman: Doç. Dr. Seniha Esen Yüksel

Mayıs 2022, 98 sayfa

Elektronik baskı devre kartı (BDK) üretiminde yüzey montaj elemanlarının (YME) BDK'ya yerleştirilmesi sırasında oluşan hataların büyük kısmı lehim hatalarından kaynaklanır. Lehimlerin boyutlarının küçük olması ve farklı şekiller alabilmeleri sebebiyle denetimleri zordur. Bu tez kapsamında lehim denetimini farklı koşullarda yapabilmek için iki farklı yöntem önerilir: Gözetimsiz derin üretken modeller ile anomali tespiti ve entropi düzenlileştirilmesi tabanlı ince taneli gözetimli sınıflandırma. İlk olarak, lehim denetimi sınırlandırılmamış bir bağlamda yani hatasız referans kart veya özel bir aydınlatma olmadan hem entegre hem de entegre olmayan devre elemanları denetimi için beta-Değişimsel Özkodlayıcı (beta-VAE) mimarisi tasarlanmıştır. Önerilen modelin, verinin daha ayrıştırılabilir çözülmüş bir temsilini öğrendiği, niteliklerin bağımsızlığını arttırdığı gösterilmiştir. Anomaliyi karakterize etmede kullanılan aktivasyon ve gradyan tabanlı temsiller karşılaştırılmıştır. Farklı beta değerlerinin daha ayrıştırılabilir temsil öğrenimine ve model başarımına olan etkisi araştırılmıştır. Beta-VAE ile anomali lehimlerin model eğitiminde yalnızca hatasız lehimler kullanarak yüksek başarı oranı ile sınıflandırılabilceği gösterilmiştir. Lehim denetimi için önerilen bir diğer yaklaşım ise ince taneli sınıflandırmadır. Hatalı ve hatasız lehim nitelikleri birbirine benzer olduğundan lehim denetiminin ince taneli sınıflandırma problemi olduğu gösterilmiştir. Lehim denetiminin

ince taneli sınıflandırmasında entropiyi düzenleyici yöntemler karşılaştırılmış ve önerilen çarpık Jensen-Shannon ıraksamasının (JSD), farklı mimarilerde diğer yöntemlerden daha başarılı olduğu gösterilmiştir. Ayrıca, çarpık JSD ile entropi düzenlenilmesinin lehimler üzerinde daha ayrıştırıcı bölgelerden çıkarım yapılmasını sağladığı gösterilmiştir. Gözetimsiz derin üretken modeller, model eğitimi sırasında nadir bulunan hatalı lehim örneklerine ihtiyaç duymazken gözetimli evrimsel ağlar kadar yüksek başarı oranı elde edemezler. Dolayısıyla önerilen gözetimli ince taneli sınıflandırma ile gözetimsiz derin üretken model tabanlı yöntemler arasında bir ödenleşim vardır.

Anahtar kelimeler: Lehim denetimi, derin üretken modeller, çözülmüş temsil öğrenimi, ince taneli sınıflandırma, entropi düzenlenilmesi.

ABSTRACT

SOLDER JOINT INSPECTION ON PRINTED CIRCUIT BOARDS

Furkan Ülger

Master of Science Thesis, Department of Electrical and Electronics

Engineering

Supervisor: Dr. Seniha Esen Yüksel

May 2022 , 98 pages

In the assembly process of printed circuit boards (PCBs), most of the errors are caused by solder joints in surface mount devices (SMDs). We propose using beta-Variational Autoencoder (beta-VAE) that learns from normal solder joints and disentangles factor of variations in data for anomaly detection of solder joints. Latterly, skewed Jensen-Shannon divergence (JSD) is proposed to maximize entropy of the probability distribution over data to classify fine-grained solder joints. In this thesis, two different methods are proposed to carry out solder inspection under different conditions: anomaly detection with an unsupervised deep generative model and fine-grained image classification with entropy regularization. At first, in order to solve the optical inspection problem in unrestricted environments with no special lighting and without the existence of error-free reference boards, we propose a new beta-VAE architecture for anomaly detection that can work on both integrated circuit (IC) and non-IC components. We show that the proposed model learns disentangled representation of data, leading to more independent features and improved latent space representations. We compare the activation and gradient-based representations that are used to characterize anomalies and observe the effect of different beta parameters on accuracy and untwining

the feature representations in beta-VAE. Finally, we show that anomalies on solder joints can be detected with high accuracy via a model trained directly on normal samples without designated hardware or feature engineering. The next proposed method for solder joint inspection is fine-grained image classification. Entropy-regularization based methods are compared in fine-grained classification of solder joints and it is shown that the proposed skewed JSD outperforms others on different model architectures. Besides, regularizing entropy with skewed JSD enables the model to focus on more class discriminative regions. While unsupervised deep generative models do not require defective solder joints that are rarely found, they can not achieve as high accuracy as the supervised fine-grained methods. Consequently, there is a trade-off between the proposed methods.

Keywords: Solder joint inspection, deep generative models, disentangled representation learning, fine-grained image classification, entropy regularization.

TEŐEKKÜR

Tüm hayatım boyunca sonsuz desteklerini esirgemeyip bugünlere gelmemi sađlayan annem Derya ÜLGER ve babam Mehmet ÜLGER'e,

Lisans ve lisansüstü eğitimim boyunca bilgisini benimle paylaşan, yönlendiren, sabır ve anlayış gösteren çok değerli tez danışmanım Doç. Dr. Seniha Esen YÜKSEL'e teşekkürü borç bilirim. Çalışmaların yayınlara dönüştürülmesindeki emekleri ve katkıları için ayrıca teşekkür ederim.

Çalışmalarımda kullanmak üzere veri kümelerini benimle paylaşan Prof. Dr. Atila Yılmaz, Doç. Dr. Dinçer Gökçen ve ekibine; sunduđu imkanlar ile çalışmalarımı destekleyen ASELSAN A.Ő. ailesine teşekkür ederim.

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	i
ABSTRACT	iii
TEŞEKKÜR	v
İÇİNDEKİLER	vi
TABLolar	ix
ŞEKİLLER	x
KISALTMALAR	xiii
1. GİRİŞ	1
1.1. Tezin Kapsamı	4
1.2. Literatüre Katkıları	4
1.3. Tezin Akışı	5
2. LİTERATÜRDEKİ LEHİM DENETİMİ YÖNTEMLERİ	6
2.1. Referans Bazlı Yöntemler	6
2.1.1. Görüntü Çıkarma	6
2.1.2. Nitelik Eşleştirme	8
2.1.3. İstatistiksel Modelleme	8
2.2. Nitelik Çıkarma Bazlı Yöntemler	9
2.2.1. Ön İşleme	9
2.2.2. Nitelikler	11
2.2.3. Sınıflandırıcı	15
2.3. Derin Öğrenme Bazlı Yöntemler	15
2.4. İnce Taneli Sınıflandırma	18
2.5. Açık Veri Kümeleri	19
3. TEMEL BİLGİLER	21
3.1. Değişimsel Çıkarsama ve Gizli Değişken Modeller	21
3.2. Entropi ve Çapraz-Entropi	24
3.3. Boyut İndirgeme Yöntemleri	24

3.3.1. Temel Bileşen Analizi (PCA)	25
3.3.2. t-Dağıtılmış Stokastik Komşu Gömme (t-SNE)	26
4. ÖNERİLEN YÖNTEMLER	27
4.1. Lehim Bölütlemesi	28
4.2. Derin Üretken Modeller	30
4.2.1. Değişimsel Özkodlayıcı (VAE)	30
4.2.2. beta-Değişimsel Özkodlayıcı (β -VAE)	31
4.2.3. Anomali Puanı	32
4.2.3.1. Geriçatma kaybı (reconstruction loss):	32
4.2.3.2. Değişimsel alt sınır (ELBO):	33
4.2.3.3. Gradyan kısıtı (GradCon):	33
4.2.3.4. Mahalanobis mesafesi:	34
4.2.4. Karar Mekanizması	34
4.3. İnce Taneli Sınıflandırma	35
4.3.1. Maksimum Entropi Düzenliştirmesi	36
4.3.1.1. Odak Kaybı (Focal loss)	36
4.3.1.2. Etiket Düzleştiren Düzenleştirme	36
4.3.1.3. Maksimum Entropi Öğrenimi	37
4.3.1.4. Çarpık Jensen-Shannon İraksaması ile Entropi Düzenliştirmesi	37
5. DENEYSEL ÇALIŞMALAR VE SONUÇLAR	38
5.1. Derin Üretken Modeller ile Deneysel Çalışmalar	38
5.1.1. Veri Kümesi ve Önişleme	39
5.1.2. Mimari ve Üst Parametreler	39
5.1.3. Deneysel Sonuçlar	42
5.2. Maksimum Entropi Öğrenimi ile Deneysel Çalışmalar	46
5.2.1. Veri Kümesi ve Üst Parametreler	46
5.2.2. Deneysel Sonuçlar	48
6. SONUÇ	50
EK 1 $D_{KL}(q_\phi(\mathbf{z} \mathbf{X}) p_\theta(\mathbf{z}))$ 'ın Çözümü	66
EK 2 KL İraksaması ve Çarpık JSD	66

EK 3Çarpık JSD'nin Dışbükeyliği ve Sürekliliği Üzerine	67
EK 4Etiket Düzleştiren Düzenleştirme	68
EK 5Aşırı Güveni Cezalandırma (Confidence Penalty)	69
EK 6Tezden Türetilmiş Yayınlar	71

TABLULAR

	<u>Sayfa</u>
Tablo 2.1 Denetlenen hata tipleri, aşamalı RGB aydınlatma kullanımı, işleme yöntemleri, nitelikler ve sınıflandırıcılara göre literatürdeki lehim denetimi yöntemleri.	17
Tablo 2.2 Denetlenen hata tipleri, aşamalı RGB aydınlatma kullanımı, işleme yöntemleri, derin öğrenme yöntemine göre literatürdeki lehim denetimi yaklaşımları.....	18
Tablo 2.3 Çalışma, görüntüleme yöntemi, çözünürlük, veri sayıları ve etiket bilgilerine göre literatürdeki açık veri kümeleri.....	21
Tablo 5.1 β -VAE için üst parametreler.	42
Tablo 5.2 Farklı anomali puanları için geriçatım bazlı ve üretken modellerin karşılaştırması (10 testin ortalaması) [1].	44
Tablo 5.3 β , üst parametrenin Gradyan kısıtı ile eğitilmiş β -VAE üzerindeki başarımlar etkisi [1].	44
Tablo 5.4 β -VAE modeli için hata matrisi.	45
Tablo 5.5 Farklı model mimarileri ve entropi-düzenleştirme yöntemlerinin lehim test kümesi üzerindeki F1-puanı (%). Her model için en yüksek başarı kalın yazılar ile verilir. Tüm sonuçlar 10 testin bir ortalamasıdır.	49
Tablo 5.6 Çarpık JSD ile eğitilmiş ResNet50 modeli için hata matrisi.	49

ŞEKİLLER

	<u>Sayfa</u>
Şekil 1.1	Kısa devre hataları içeren bir BDK görüntüsü. 2
Şekil 1.2	Elektronik kart üretiminde kullanılan temsili bir YME montaj hattı ([2, 3]'den esinlenerek çizilmiştir). 2
Şekil 2.1	XOR mantık operasyonu ile hatalı ve referans görüntü arasında fark işlemi [4]. 7
Şekil 2.2	(a) Entegre devre elemanı (IC) bölütlemesi (devre elemanı gövdesi, bacakları ve lehim), (b) entegre olmayan (non-IC) devre elemanı bölütlemesi ([5, 6]'dan esinlenerek çizilmiştir). 11
Şekil 2.3	YME bölütlemesi [1]. (a) Pasif devre elemanı, (b) entegre devre elemanı parselasyonunu göstermektedir. 12
Şekil 2.4	YME lehim yüzeylerini aydınlatmada kullanılan aşamalı RGB aydınlatma ([7]'den esinlenerek çizilmiştir). 13
Şekil 2.5	Dairesel RGB ışık altında çeşitli lehim hataları ([7, 8]'den esinlenilerek çizilmiştir). 14
Şekil 3.1	Gizli Değişken Modeli ([9]'den esinlenerek çizilmiştir). 22
Şekil 4.1	BDK üzerinde çokgen etiketleme örneği. 28
Şekil 4.2	YOLACT modelinin (a) eğitim ve (b) doğrulama kümesinde başarı oranı. 29
Şekil 4.3	YOLACT modelinin test kümesindeki lehim bölütlemeleri. 30
Şekil 4.4	VAE mimarisi ve gizli değişken vektörü üzerindeki önsel izotropik Gaussian dağılım [1]. 31
Şekil 4.5	(a) Lehim veri kümesinden örnekler: İlk satırda hatasız lehimler, ikinci satırda hatalı lehimler. (b) Mavi noktalar, VGG16 ağınnın son tam çizge katmanı (fully connected layer) üzerinde lehim verilerinin PCA izdüşümünü, kırmızı noktalar ImageNet örneklerini temsil eder. . 35

Şekil 5.1	İlk sütundaki bloklar, kodlayıcıyı temsi eder. Kodlayıcı, gizli değişken vektörü z 'yi elde edebilmek için gerekli parametreleri, beş evrişimsel (conv), üç maksimum ortalama ve evrişimsel darboğaz katmanları ile elde eder. İkinci sütundaki bloklar ise kodçözücüü ifade eder. Kodçözücü, z 'den girdi görüntüsünü geriçatmak için altı devrik evrişimsel (Tconv) ve üç örnekleme hızlandırma (upsampling) katmanı kullanır. bn ve LReLU, sırasıyla toplu normalleştirme (batch normalization) ve kaçak doğrultmaçlı aktivasyon biriminin kısaltmalarıdır [1].	41
Şekil 5.2	Geri çatılmış (reconstructed) normal ve abnormal örnekler. 1. ve 3. satır girdi görüntüleri ve gerçek etiketleri gösterirken 2. ve 4. satır geri çatımlar ile birlikte tahmin edilen etiketler ve anomali puanlarını gösterir [1].	43
Şekil 5.3	(a) $\beta = 0.1$, (b) $\beta = 1$, (c) $\beta = 3$ ve (d) $\beta = 10$ için t-SNE uygulaması sonrası GradCon ile eğitilen β -VAE'in iki boyutlu gizli uzay temsili. Mavi ve kırmızı daireler sırasıyla normal ve anomali test verisini temsil etmektedir [1].	46
Şekil 5.4	(a) $\beta = 0.1$, (b) $\beta = 1$, (c) $\beta = 3$ ve (d) $\beta = 10$ için t-SNE uygulaması sonrası ELBO ile eğitilen β -VAE'in iki boyutlu gizli uzay temsili. Mavi ve kırmızı daireler sırasıyla normal ve anomali test verisini temsil etmektedir [1].	47

- Şekil 5.5 ResNet50 modeli ortalama havuzlama (average pooling) katmanını test verisi çıktı niteliklerinin temel bileşenleri (principle components). Normal ve hatalı lehimler sırasıyla mavi ve kırmızı daireler ile gösterilmektedir. Yeşil daireler, düzenlenileştirmesiz eğitim ile sınıflandırma sonucu yanlış sınıflandırılırken (yanlış negatifler) çarpık JSD ile doğru sınıflandırılan (doğru pozitifler) hatasız lehimleri göstermektedir. Yeşil çarpılar ise çarpık JSD ile doğru sınıflandırılırken (doğru negatifler) düzenlenileştirme olmadan yanlış sınıflandırılan (yanlış pozitifler) hatalı lehim örneklerini temsil etmektedir. 50
- Şekil 5.6 Grad-CAM yaklaşımı sınıf ayırımında belirleyici bölgeleri gösterir. RGB test görüntüleri (a). Düzenlenileştirmesiz (b) ve çarpık JSD (c) ile eğitilen VGG16 modelinin aktivasyon haritaları. Isı haritası, sınıflandırma için modelin odaklandığı kısmı gösterir. Yoğunluk, maviden kırmızıya doğru gittikçe artar. En çok çarpık JSD ile düzenlenileştirilmiş modelin CAM sonuçları lehimlenmiş bölgeler ile örtüşür. 51

SEMBOLLER ve KISALTMALAR

Semboller

\mathbf{X}	: Veri noktası (görüntü)
$\boldsymbol{\mu}$: Ortalama değerler vektörü
\mathbf{z}	: Gizli değişkenler vektörü
$p_{\theta}(\mathbf{X})$: Marjinal olabilirlik
$p_{\theta}(\mathbf{X} \mathbf{z})$: Verinin olabilirliği
$p_{\phi}(\mathbf{z} \mathbf{X})$: Gizli değişkenlerin doğru sonsal dağılımı
$q_{\phi}(\mathbf{z} \mathbf{X})$: Gizli değişkenlerin yaklaşık sonsal dağılımı
α, γ	: Gradyan kaybı ağırlıkları
β	: Kullback-Leibler ıraksaması ağırlığı
θ, ϕ	: Kodlayıcı ve kodçözücü model parametreleri
ϵ	: Parametrize etmek için rastgele bir değişken
$H(p)$: p , olasılıksal dağılımının entropisi
$H(p, q)$: İki olasılıksal dağılım p, q arasındaki çapraz entropi
$D_{KL}(p q)$: İki olasılıksal dağılım p, q arası Kullback-Leibler (KL) ıraksaması
$J(p q)$: İki olasılıksal dağılım p, q arası Jensen-Shannon ıraksaması (JSD)
$J_{\alpha}(p q)$: İki olasılıksal dağılım p, q arası çarpık Jensen-Shannon ıraksaması

Kısaltmalar

GT	: Gerçek referans değer (ground-truth)
VAE	: Değişimsel Özkodlayıcı (Variational Autoencoder)
AAE	: Çekişmeli Özkodlayıcı (Adversarial Autoencoder)
BDK	: Baskı Devre Kartı
NCC	: Normalleştirilmiş Çapraz İlinti

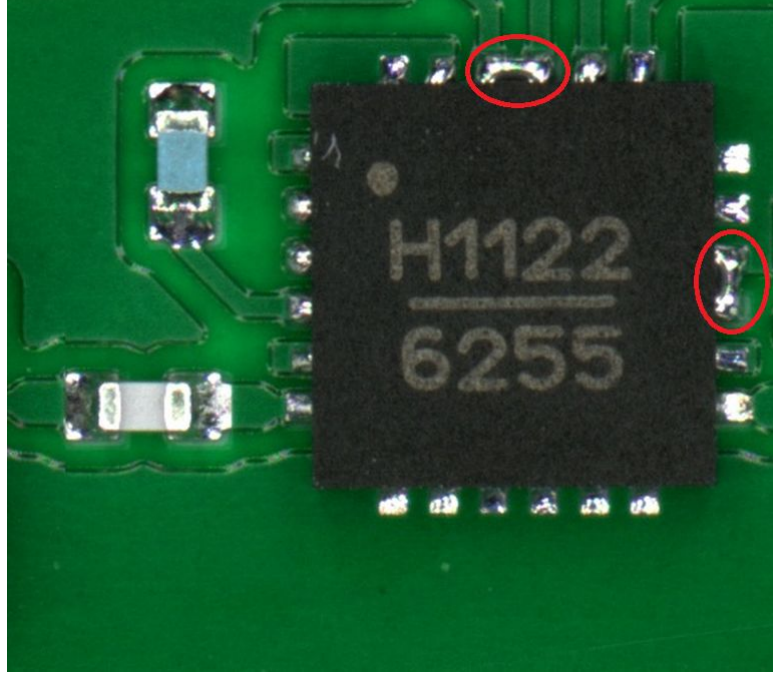
YME	:	Yüzey Montaj Elemanı
GradCon	:	Gradyan Kısıtı
ADAM	:	Stokastik Optimizasyon için Bir Yöntem
SGD	:	Stokastik Gradyan İnişi
PCA	:	Temel Bileşen Analizi
t-SNE	:	t-Dağıtılmış Stokastik Komşu Gömme
YMT	:	Yüzey Montaj Teknolojisi
YME	:	Yüzey Montaj Elemanı
KL	:	Kullback-Leibler İraksaması
JSD	:	Jensen-Shannon İraksaması
AOI	:	Otomatik Optik Denetim
SPI	:	Krem Lehim Denetimi
API	:	Otomatik Yerleşim Denetimi
IPC	:	Institute of Printed Circuit Boards
NS	:	Belirtilmemiş
YOLO	:	You Only Look Once
YOLACT	:	You Only Look at Coefficients
SIFT	:	Ölçekten Bağımsız Nitelik Dönüşümü (Scale Invariant Feature Transform)

1. GİRİŞ

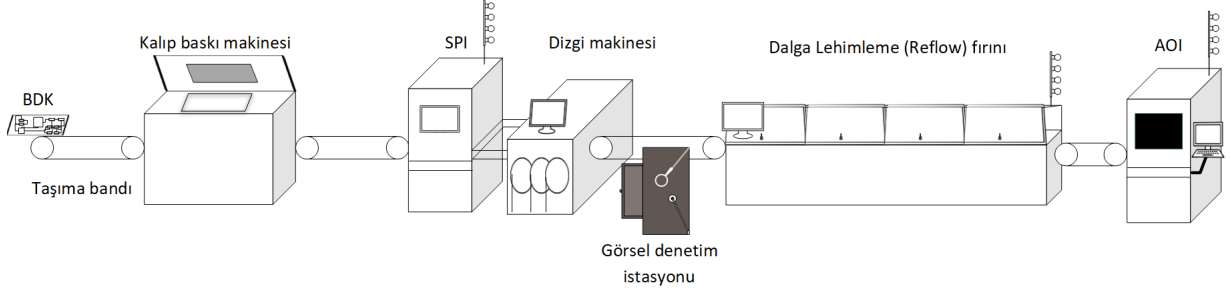
Yüzey Montaj Teknolojisi (YMT) ile elektronik üretim, elektronik devre elemanlarının baskı devre kartına yerleştirilmesi sürecine verilen isimdir. Son yıllarda üretim hatlarındaki yüksek yoğunluklu ve hızlı üretim, lehim hatalarının oluşmasına sebep olmaktadır. Hataların üretimin erken safhalarında tespiti; üretim güvenilirliğinin artırılması, yenileme (rework) maliyetleri ve hurda kartların azaltılması açısından önemlidir. Literatürde YMT süreci ve lehim denetiminin önemini anlatan çalışmalar vardır [2, 10].

YMT sürecinde BDK, çeşitli aşamalardan geçer: krem lehim uygulaması ve denetimi, yüzey montaj elemanı (YME) yerleştirilmesi ve denetimi, dalga lehimleme (reflow soldering) ve Otomatik Optik Denetim (AOI). İlk adımda kalıp baskı makinesi (stencil printer), krem lehim farklı çıplak BDK'lar için özel olarak oluşturulmuş kalıplar ile uygular. Bu uygulama sıcaklık, nem gibi çevresel etmenlere karşı oldukça hassastır. Bu sebepten krem lehim uygulaması aşaması ardından Krem Lehim Denetimi (SPI) cihazları ile denetim yapılır. SPI cihazı, krem lehimin doğru bir şekilde lehim pedlerine uygulanıp uygulanmadığını denetler. Daha sonra YME'ler, ilgili pedlere dizgi makineleri vasıtasıyla yerleştirilir. Görsel denetim istasyonunda ise YME yerleştirilen karttaki yerleşim hatalarının tespiti için operatör görsel denetleme yapar. Bu denetlemede amaç BDK üzerindeki kayma, dönme, polarite, boyut ve etiket hataları gibi hataları tespit etmektir. Dalga lehimleme (reflow) fırınında lehimlerin öncelikle kademeli olarak ısıtılıp daha sonra soğutulurak katı hale gelmesi ile YME'nin lehim pedine kalıcı olarak bağlanması sağlanır. Otomatik Optik Denetim (AOI) aşamasında ise hem YME'nin yerleşim hataları hem de lehim hataları denetlenir. Bu aşamadan itibaren YMT süreci tamamlanır [3]. Denetlenmesi amaçlanan bir BDK görüntüsü Şekil 1.1'de, temsili bir elektronik kart üretim hattı Şekil 1.2'de verilir.

Otomatik optik denetim cihazları öncesinde lehim kalitesi denetimi bir operatör tarafından manuel olarak yapılırdı. Ancak manuel denetim hem çok zaman alan hem de hataya yatkın bir denetim şeklidir. Elektronik üretim endüstrisindeki hızlı gelişim, BDK denetiminin hem yüksek kalitede hem de yüksek hızda otomasyonunu gerekli kılar.



Şekil 1.1. Kısa devre hataları içeren bir BDK görüntüsü.



Şekil 1.2. Elektronik kart üretiminde kullanılan temsili bir YME montaj hattı ([2, 3]'den esinlenerek çizilmiştir).

Yüksek kalitede üretim yapılabilmesi için Institute of Printed Circuit Boards (IPC) gibi kurumlar standartları belirler. BDK üretim hatlarında (YMT süreci) kullanılan denetim cihazları, bu standartları karşılayacak şekilde denetim yapar. Optik bilgiyi kullanarak lehim denetimi otomasyonu AOI cihazları ile sağlanır. AOI cihazları, yüksek çözünürlüklü kamera ve farklı aydınlatma teknikleri ile görüntüler alır. Alınan görüntüler işlenir ve belirleyici nitelikler çıkarılır. AOI cihazları güvenilir, hızlı ve isabetli denetim sunmanın yanı sıra manuel denetime göre daha az maliyetlidir. Ayrıca bu cihazlar üretimin farklı safhalarında da uygulanabilir.

Lehim denetimi yapılmasında birçok zorluk vardır. Denetlenmesi amaçlanan lehimler genellikle çok küçüktür ve BDK üzerinde YME türüne göre farklı şekiller alabilirler. Ayrıca lehim hataları birçok farklı türde oluşabilir: kısa devre, aşırı/eksik lehim, soğuk lehim, lehim pedinin ıslanmaması (non-wetting), dikey malzeme, kaymış malzeme, sünmüş lehim, lehim topçukları vb. Ayrıca hatalı ve hatasız lehimler arasında görsel olarak çok küçük farklılıklar bulunur [10] ve bu küçük farklılıklar BDK'da onulmaz hatalara neden olabilir. Bu sebepten lehim hatalarının üretimin erken aşamalarında tespiti büyük önem taşımaktadır.

Literatürde lehim denetimi problemi için sunulan yaklaşımlar temelde referans tabanlı yaklaşımlar, nitelik çıkarma ve derin öğrenme tabanlı yaklaşımlar olmak üzere 3 temel alanda incelenebilir. Referans tabanlı yaklaşımlar hatalı BDK'nın hatasızını içeren bir referansa göre karşılaştırma yapar. Nitelik çıkarma yaklaşımları ise önceden belirlenmiş ayrıştırıcı nitelikleri sınıflandırarak denetim sağlar. Son yıllarda yapılan çalışmalarda derin öğrenmenin geleneksel nitelik çıkarımı tabanlı yaklaşımlardan daha başarılı olması nedeniyle lehim denetimi için de birçok derin öğrenme yaklaşımları önerilir. Bu tezde de lehim denetimi problemi için derin öğrenme tabanlı yaklaşımlar önerilir.

Bu tezde lehim denetimi problemine iki ayrı yönden çözüm üretilmesi amaçlanmıştır. İlk yaklaşım, BDK üretim hatlarında lehim hataları nadir olduğundan hatalı lehim gerektirmeden normal lehim modelleyen gözetimsiz derin üretken model tabanlı yaklaşımdır. Bu yaklaşımda lehimlere anomali puanı atanıp daha önceden belirlenmiş bir eşik ile karşılaştırma yapılır ve lehimin hatalı/hatasız olduğuna karar verilir. Hatalı lehim örneklerinin yeterince bulunduğu durumlarda ise gözetimli evrimsel sinir ağları kullanılabilir. Görsel olarak birbirine benzeyen hatalı/hatasız lehim görüntülerini yüksek başarımla oranı ile sınıflandırabilmek için ise ince taneli lehim sınıflandırma problemine bir maksimum entropi düzenleme yaklaşımı önerilir. Maksimum entropi düzenlemesi için önerilen yaklaşımın literatürdeki mevcut yaklaşımlardan daha başarılı olduğu gösterilir. Sunulan yöntemler Bölüm 1.1.'de, literatüre katkılar Bölüm 1.2.'de verilmiştir.

1.1. Tezin Kapsamı

Bu tezde BDK üzerinde bulunan YME lehimlerinin denetlenmesi amaçlanmıştır. Literatürdeki mevcut yaklaşımlar çoğunlukla özel bir aydınlatma, nitelik mühendisliği (lehim hata tespiti için belirleyici niteliklerin bulunması) ve model eğitiminde hatalı lehim veya karşılaştırmak amacıyla hatasız bir referans BDK gerektirmektedir. Bu gereksinimlere ihtiyaç duymadan lehim denetimi yapabilmek amacıyla gözetimsiz derin üretken modeller ile hatasız lehimlerin modellenmesi önermekteyiz. Ek olarak çözülmüş temsiller öğrenilerek hatalı ve hatasız lehimler için daha ayrıştırıcı bir temsil öğrenilebileceği gösterilmiştir. Test sırasında farklı anomali puanlama yöntemleri kullanılarak hatalı/hatasız lehim ayrımı yapılmıştır.

Hatalı ve hatasız veri sayısının yeterli olduğu durumlarda ise gözetimli evrişimsel sinir ağları kullanılabilir. Hatalı/hatasız veri örnekleri birbirine benzediğinden ve aynı sınıftaki örnekler arası görsel farklılıklar fazla olabildiğinden lehim denetimi problemine ince taneli görüntü sınıflandırma problemi olarak yaklaşılmıştır. İnce taneli (fine-grained) sınıflandırmada entropi-düzenleştirme tabanlı yöntemlerin başarı oranını arttırdığı literatürde gösterilmektedir. Bu yöntemlere odak kaybı (focal loss), etiket düzleştiren düzenleştirme (label smoothing regularization), maksimum entropi öğrenimi örnek olarak verilebilir. Bu tezde ince taneli lehim sınıflandırması için çarpık Jensen-Shannon ıraksamasının (JSD) diğer entropi-düzenleştirme tabanlı yaklaşımlardan daha başarılı olduğu ve çarpık JSD ile daha ayrıştırıcı bölgelere odaklanılabildiği gösterilmiştir.

1.2. Literatüre Katkılar

Bu tezin literatürde katkıları aşağıda verildiği gibi özetlenmiştir:

- Lehim denetimi problemini sınırlandırılmamış bir bağlamda çözmek amaçlanmıştır: Özel bir aydınlatma, nitelik mühendisliği ve model eğitiminde hatalı lehim veya hatasız bir referans BDK gerektirmez.

- Lehim denetimi problemine özel bir β -VAE mimarisi tasarlanmış, farklı anomali puanlama teknikleri ile analiz edilmiş ve β terimindeki değişimin etkisi araştırılmıştır. Ayrıca, β -VAE kullanımının başarıma olan etkisi ve normal, anomali lehimlerin gizli uzay temsillerinin görsel olarak nasıl ayrıştırılabilir hale geldiği gösterilmiştir.
- Modelimiz, tek bir mimaride hem entegre hem de entegre olmayan (pasif) devre elemanlarının lehim denetiminde kullanılmıştır. Bildiğimiz kadarıyla anomali tespiti kapsamında daha önce literatürde böyle bir çalışma mevcut değildir.
- Lehim denetimi probleminin ince taneli bir görüntü sınıflandırma problemi olduğu gösterilmiştir.
- Lehimlerin ince taneli sınıflandırmasında entropi-düzenleştirme için çarpık JSD kullanımı önerilmiştir. Bu düzenleştirme yöntemi ile farklı modeller için başarı oranının artırılabilirdiği gösterilmiştir.
- Entropi-düzenleştirme tabanlı modellerin görüntüde daha ayırt edici bölgelere odaklanabilirdiği ve bunun model başarımını arttırdığı gösterilmiştir.

1.3. Tezin Akışı

Tezin akışı aşağıda verilmiştir:

- 1. Bölümde tezin motivasyonu, problem tanımı, literatüre olan katkılar ve tezin kapsamından bahsedilmiştir.
- 2. Bölümde lehim denetimi için sunulan mevcut çalışmalar, yardımcı materyallerden bahsedilmiştir.
- 3. Bölümde tezde kullanılan yaklaşımların temelleri verilmiştir.
- 4. Bölümde önerilen derin üretken modellerden ve maksimum entropi öğrenimi ile ince taneli sınıflandırmadan bahsedilmiştir.

- 5. Bölümde önerilen yöntemler ile yapılan deneysel çalışmalar ve sonuçları gösterilmiştir.
- 6. Bölümde tezin sonuçları ve yapılan çıkarımlara yer verilmiştir.

2. LİTERATÜRDEKİ LEHİM DENETİMİ YÖNTEMLERİ

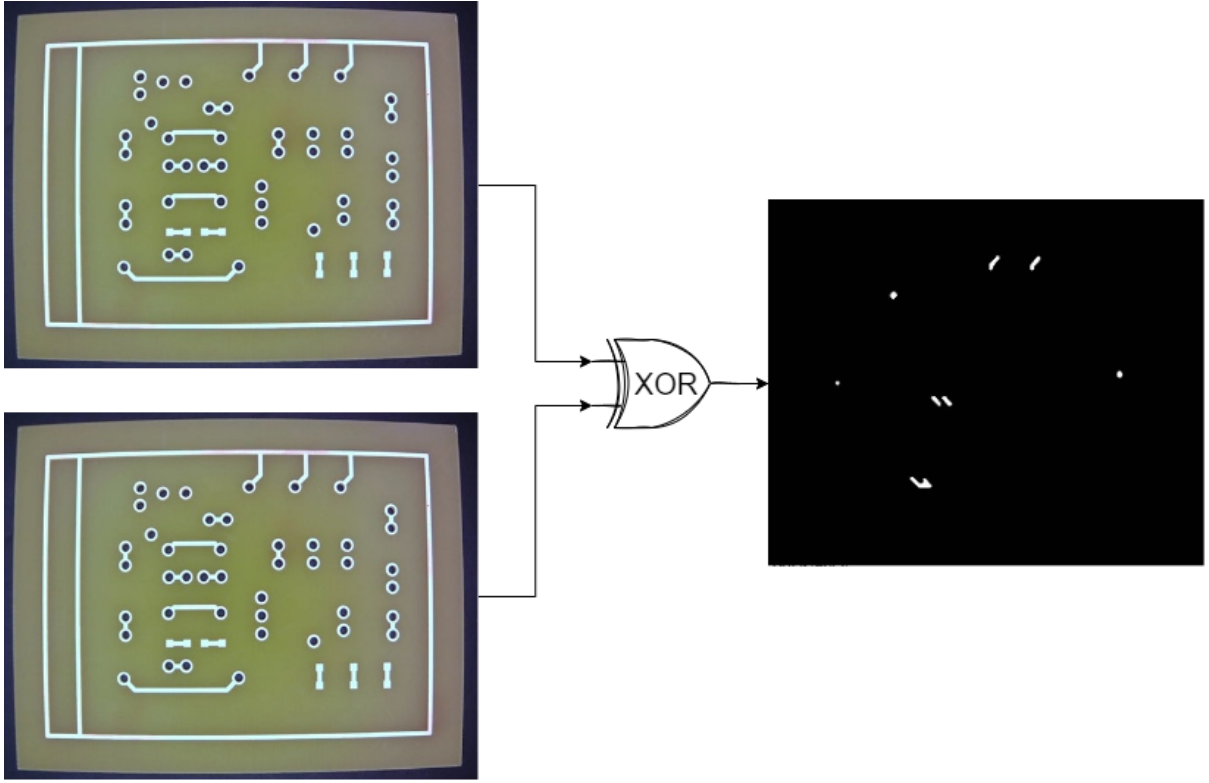
BDK lehimleri denetimi eskiden beri süregelen bir problemdir. Denetim bir veya birden fazla kamera ile dış etkilerden etkilenmeyecek bir düzenekte görüntüler alınarak yapılmaktadır. Lehim yüzeylerinin aydınlatılmasında da çeşitli aydınlatma teknikleri kullanılmaktadır. Lehimlerin otomatik optik denetimi için literatür temelde üçe ayrılmaktadır: Referans bazlı yöntemler, nitelik çıkarma bazlı yöntemler ve derin öğrenme bazlı yöntemler.

2.1. Referans Bazlı Yöntemler

Referans bazlı yöntemler test edilen BDK'nın referans (hatasız) hali ile piksel piksel veya nitelik karşılaştırması yapmaktadırlar [10]. Temelde görüntü çıkarma ve nitelik eşleştirme yaklaşımlarını kullanılmaktadır.

2.1.1. Görüntü Çıkarma

Görüntü çıkarma yaklaşımında bir referans kart görüntüsü ve test edilen kart görüntüsünün piksel seviyesinde karşılaştırılması amaçlanmaktadır. Referans görüntü, test edilen görüntünün hatasız bir versiyonudur. Çıkarılmış görüntü, referans ile hatalı karttaki farklılıkları (hatalı kısımları) gösterir. Görüntü çıkarmada kullanılan mantıksal XOR operasyonunun bir örneği Şekil 2.1'te verilmiştir. Referans görüntü bazlı yöntemler BDK su yolu denetiminde [4, 11, 12], BDK üzerindeki YME denetiminde kullanılmaktadır [4, 13–15]. Piksel seviyesinde karşılaştırma yapılabilmesi için bölütleme ve görüntü



Şekil 2.1. XOR mantık operasyonu ile hatalı ve referans görüntü arasında fark işlemi [4].

çakıştırma yaklaşımları kullanılarak, sonrasında eşleşen piksel farklarına bakılarak hata tespiti yapılmaktadır [4, 11, 13]. Daha hassas tespit için hatasız kartların Gauss karışımları ile modellenmesi yapılabilir. Böylece test görüntülerinin öğrenilen Gauss dağılıma olan uzaklıkları devre elemanlarının yerleşim hatalarını verir [15]. Ye vd. [13] ise hatasız entegre lehim ve hatalı entegre lehim görüntüleri arasındaki farkı renk özü (hue) kanalında analiz eder. Görüntü çıkarma yaklaşımı yalnızca uzamsal alanda değil frekans alanında da kullanılabilir. Tsai vd. [16] Fourier dönüşümünü kullanarak kart görüntülerini frekans alanına taşır, referans ve test görüntüleri arasındaki farklara bakarak hata tespiti amaçlar. Görüntü çıkarma bazlı yaklaşımları uygulaması kolaydır ancak renk değişimleri, görüntü çakıştırma, aydınlatma hassasiyeti ve yansıma değişimlerinden etkilendikleri için pratikte kısıtlı kullanıma sahiptirler.

2.1.2. Nitelik Eşleştirme

Nitelik eşleştirme yönteminde, BDK karşılaştırmak yerine test görüntüsünden çıkarılan nitelikler ve referans nitelikleri karşılaştırılmaktadır [10, 17]. Karşılaştırılan nitelikler aynı ise BDK hatasız kabul edilmektedir. Bu eşleştirme prosedürüne taslak eşleştirme adı verilir. Taslak eşleştirme yönteminde yalnızca nitelikler kullanıldığından depolama ve zaman gereksinimi azdır. Hatasız BDK'dan lokal ikili örüntüler [18], yön nitelikleri [19]; florasan aydınlatma yansımalarından su yolu örüntüleri [20] çıkartılarak test görüntüleriyle karşılaştırıp su yolu hatalarının tespiti amaçlanmaktadır. Taslak eşleştirme amacıyla normalleştirilmiş çarpaz ilinti (NCC), ilinti katsayısı gibi yöntemler, BDK üzerindeki yerleşim hataları (eksik, hizalanmamış, dönmüş devre elemanı) tespitinde kullanılmaktadır [21–24]. Görüntü çıkarma yaklaşımının aksine çevresel etmenlerden daha az etkilendikleri için daha gürbüz bir yaklaşımdır. Nitelik eşleştirmenin en büyük dezavantajı, gerçek senaryolarda uygulanabilirlik için çok sayıda taslak gerektirmeleridir.

2.1.3. İstatistiksel Modelleme

İstatistiksel modelleme bazlı yöntemler, referans aldıkları hatasız YME lehimlerini modelleyerek test edilen lehimin hatalı olup olmadığını tespit etmeyi amaçlamaktadırlar. Xie vd. [25] hatasız referans lehimin gri değerlerinin ortalaması ve varyansını kullanarak modellemektedirler. Daha sonra test görüntüsü ile farkı alınarak piksel farklılıklarını sınıflandırmada kullanılır. Bir başka çalışmada [26] ise entegre devre elemanı lehimleri modellemesinde, görsel arkaplan çıkarma (ViBe) yaklaşımı kullanılmaktadır. Sabit bir taslak kullanımı, yaklaşımın kullanılabilirliğini azaltacağından farklı lehimler için farklı taslaklar kullanılması da önerilmektedir [13]. Gürbüz Temel Bileşen Analizi (RPCA) yöntemi ise lehimlerin düşük boyutlu temsilinin öğrenilmesinde kullanılmaktadır [27]. Ancak büyük matrislerin özdeğerlerinin (eigenvalue) hesaplanması uzun zaman alabilir. Jiang vd. [28], krem lehim denetimi için birçok biyolojik renk nitelik haritalarından oluşan alt manifold öğrenirler. Bu sayede Bölüm 2.2.'de bahsedilen nitelik çıkarma bazlı yaklaşımlarda sıklıkla kullanılan aşamalı aydınlatmaya ihtiyaç duymadan denetim

yapılabilmektedir. İstatistiksel modelleme bazlı yaklaşımlar, model eğitiminde hatalı lehim verisi olmadan da denetim yapabilir ancak model öğrenimi için taslaklar sınırlı olduğundan hata tespitinde genelleşebilirliği tartışmalıdır.

Modelleme bazlı yaklaşımlarda yaygın ön işleme yöntemi, renk uzayı dönüşümleridir. Standart üç yük bağlaşımlı aygıt (CCD) kamera, kırmızı-yeşil-mavi (RGB) görüntüler üretir. Bazı çalışmalarda lehim bölgesinin arka plandan ayırt edilerek tespit edilmesinde farklı renk uzaylarının RGB'den daha etkili olduğu gözlemlenmektedir. Örneğin, lehim görüntülerini arka plandan ayırıp bölütleyebilmek için parlaklık-renklilik (YIQ) [29], hatasız lehimleri modellemede veya hata tespitinde renk özü-doygunluk-parlaklık (HSV) renk uzayları kullanılmaktadır [13, 26, 27].

2.2. Nitelik Çıkarma Bazlı Yöntemler

Nitelik çıkarma bazlı yaklaşımlar, temelde üç adımdan oluşmaktadır: Ön işleme, nitelik çıkarımı ve sınıflandırma. Ön işleme yöntemleri, lehim görüntülerinin çevresel etmenlerden mümkün olduğunca arındırılarak bölütlenmesi için kullanılmaktadır. İlgilenilen bölgeler elde edildikten sonra önceden belirlenen nitelikler çıkarılır veya nitelik çıkarma amaçlı çeşitli dönüşümler uygulanır. Son olarak çıkarılan nitelikler lehim hata karakteristiklerini belirlemek amacıyla bir sınıflandırıcıya sokulur. BDK denetiminde kullanılan nitelik çıkarımı tabanlı yaklaşımların özeti Tablo 2.1'de verilmiştir.

2.2.1. Ön İşleme

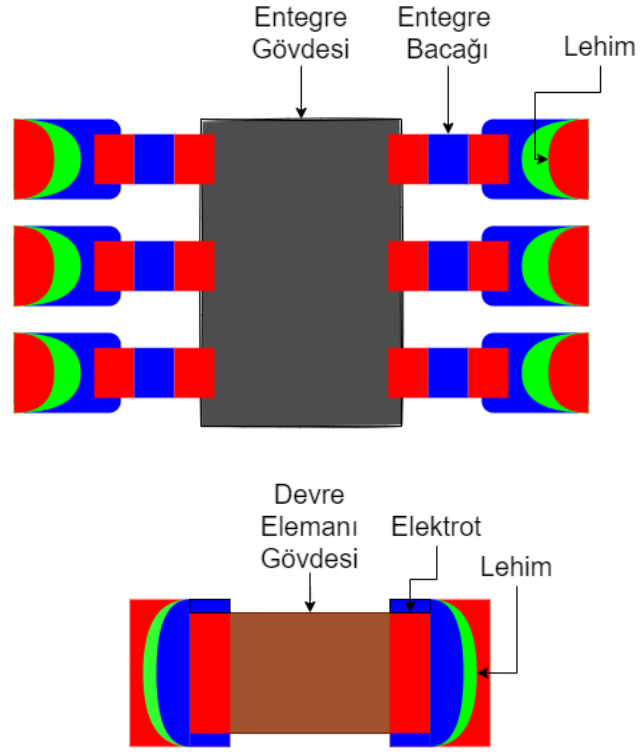
BDK üzerindeki lehimlerin bölütlenmesi, denetimin doğru yapılması için önemli bir adımdır. Bölütlemenin başarıyla yapılabilmesi için bölütleme öncesinde çeşitli yaklaşımlar uygulanmaktadır. Mar vd. [29], BDK'ların hizalanmasında Hough doğru dönüşümünü [30], aydınlatma tipindeki değişmelere karşın aydınlatma standartlaştırma (illumination normalization) yöntemini önermektedir. Hough dönüşümü ile BDK'daki hizalama problemleri tespit edilerek yatay ve dikeyde BDK'nın hizalanması sağlanmaktadır.

Aydınlatma standartlaştırması, düşük frekanslı ayırık konsinüs değişimi katsayılarını atarak aydınlatmanın BDK üzerinde düzgün dağılmasını ve bölütlemenin iyileştirilmesini sağlamaktadır. Daha sonra, lehim görüntülerini arkaplandan ayırıp bölütleyebilmek için öncelikle RGB renk uzayından YIQ uzayına dönüştürür. Gri piksel değerlerini eşikleyerek lehimleri bölütler. Jiang vd. [28] ise çıplak kart (bare board) lehimlerinin bölütlemesi için BDK üzerindeki su yolları ile lehimler arasındaki renk farklılıklarından yararlanmaktadırlar. İkili eşikleme kullanarak lehim görüntülerini bölütler, arkaplan gürültüsü filtrelenmesinde ise morfoloji kullanılmaktadır. Zeng vd. [31, 32] ise hem entegre hem de pasif devre elemanları bölütlenmesinde lehim yüzeyi üzerindeki renk dağılımları ve aydınlatmalardan yararlanmaktadır. Öncelikle BDK üzerindeki yansıtıcı tüm yüzeyler çoklu eşikleme yaklaşımı ile tespit edilir. Daha sonra, Gauss Karışım Dağılımı (GMM) ile modellenen renk dağılımı kullanılarak BDK üzerindeki delik ve işaretler filtrelenir [32].

Entegre veya entegre olmayan devre elemanlarından nitelik çıkarılacak bölütlenmiş bölgeler, ayrıca birçok alt bölümlere ayrılmaktadır [5, 6, 8, 33, 34]. Wu vd. [5], Şekil 2.2 (a)'de gösterildiği gibi entegre devre elemanını, Hongwei vd. [6] ise Şekil 2.2 (b)'deki gibi entegre olmayan devre elemanlarını birçok alt bölüme ayırır. Daha sonraki çalışmalarda, nitelikler yalnızca entegre gövdesi ve lehim pedlerinden çıkarılmaktadır [8, 33]. Song vd. [34] ise genetik algoritmayı kullanarak niteliklerin nereden çıkarılmasını gerektiğini belirlemektedirler.

Sınıflandırıcı başarı oranını arttırmak amacıyla bölütlenmiş bölgelerden çıkarılacak belirleyici nitelikleri seçebilmek için entropi [33], Adaboost [6], kalımsal (genetic) algoritma [8] yaklaşımları ile nitelikler belirlenip başarı oranının artırılması da amaçlanmaktadır.

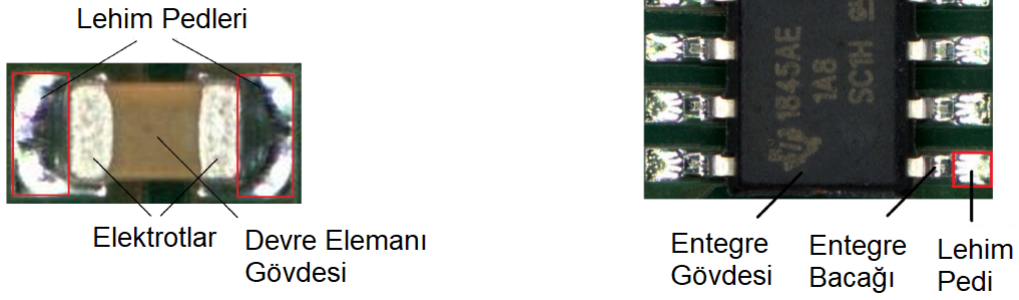
Ong vd. [35], dikey görüntülemeye ek olarak bir ayna yardımıyla eğimli görüntüleme yaparak lehimlerle ilgili daha çok geometrik bilgi almayı amaçlamaktadır. Eğimli görüntüleme ile alınan verilerde sınıflandırma başarısının dikey görüntülemeye göre yüksek olduğu gözlemlenmektedir.



Şekil 2.2. (a) Entegre devre elemanı (IC) bölütlemesi (devre elemanı gövdesi, bacakları ve lehim), (b) entegre olmayan (non-IC) devre elemanı bölütlemesi ([5, 6]'dan esinlenerek çizilmiştir).

2.2.2. Nitelikler

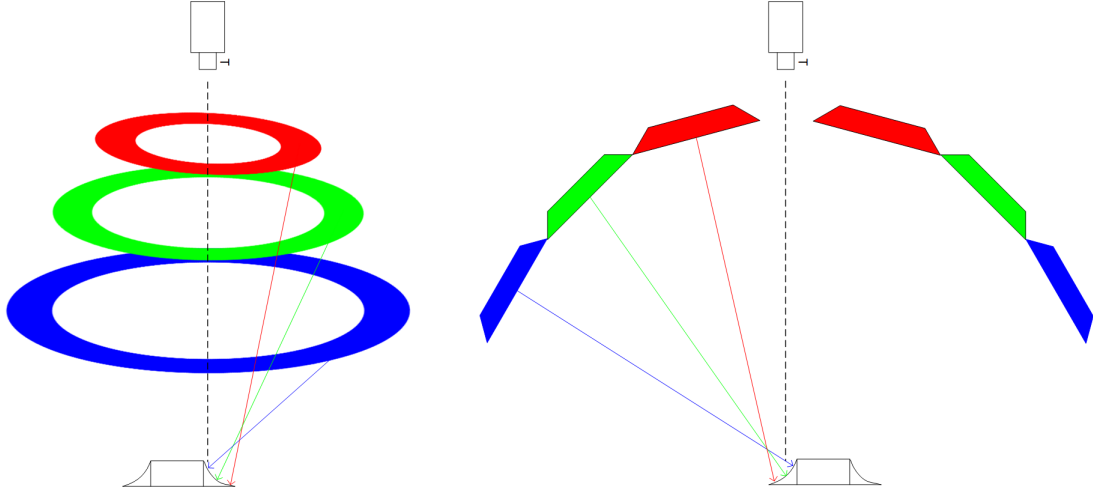
Nitelik çıkarma bazlı yöntemler, hatalı ve hatasız lehim, su yolu veya YME hataları tespiti için belirleyici nitelikler çıkarmayı amaçlamaktadırlar. Lehim durumunu belirlemede, büyük oranda lehimin geometrik ve optik kaynaklı nitelikleri kullanılmaktadır. Bu nitelikler başlıca geometrik nitelikler, renk nitelikleri ve diğer nitelikler olarak sınıflandırılmaktadır. Nitelikler, genellikle YME lehim yüzeylerinden ve gövdelerinden çıkarılmaktadır. YME ve lehim pedi bölümleri Şekil 2.3'te verilmektedir. Çıkarılan nitelikler daha sonra bir sınıflandırıcıya sokularak lehimin hatalı olup olmadığı ve ne tip bir hata olduğu tespit edilmektedir.



Şekil 2.3. YME bölütlemesi [1]. (a) Pasif devre elemanı, (b) entegre devre elemanı parselasyonunu göstermektedir.

Capson vd. [36] lehim hataları tespitinde aşamalı aydınlatma kullanarak yansıtıcı lehim yüzeylerinde renk konturları oluşturmaktadır. Bu sayede üç boyutlu lehim yüzeyi bilgisi, iki boyutta kullanılabilir. Lehim hata denetiminde kullanılan aşamalı RGB aydınlatmanın bir örneği Şekil 2.4'te verilmektedir. Kırmızı, yeşil ve mavi aydınlatma farklı açılardan lehim yüzeylerine geldiği için farklı lehim hatalarında, lehim yüzeylerinde farklı renk konturları oluşmaktadır. Bu bilgi, lehim hata tipini karakterize etmede kullanılır. Lehim yüzeyinde, farklı lehim hataları için oluşan renk konturları Şekil 2.5'te gösterilmektedir. Aydınlatma sonucunda oluşturulan renk konturlarından çıkarılan renk nitelikleri başlıca: Ortalama gri değeri, beyaz piksellerin oranı (percentage of highlight) [8, 33, 34, 37], renk değişimleri [7] olarak sıralanabilir. Kim vd. [37] hatasız, eksik ve aşırı lehim tespiti için her bir RGB kanalın ortalama gri değerleri ve eşikleme sonrası beyaz piksellerin oranını kullanmaktadır. Renk niteliklerinin belirleyici olmadığı durumlarda ek belirleyici niteliklere ihtiyaç vardır. Lehim yüzeyi yansıtıcı olduğundan farklı açılardan uygulanan aydınlatma, lehim yüzeylerinden dağılır ve yansır. Yansıma ve dağılımlar, lehimin yönelimini hesaplamada kullanılır. Wu vd. [7] ise literatürde kullanılan yaygın renk niteliklerine ek olarak renk değişimleri kullanımını önermektedir.

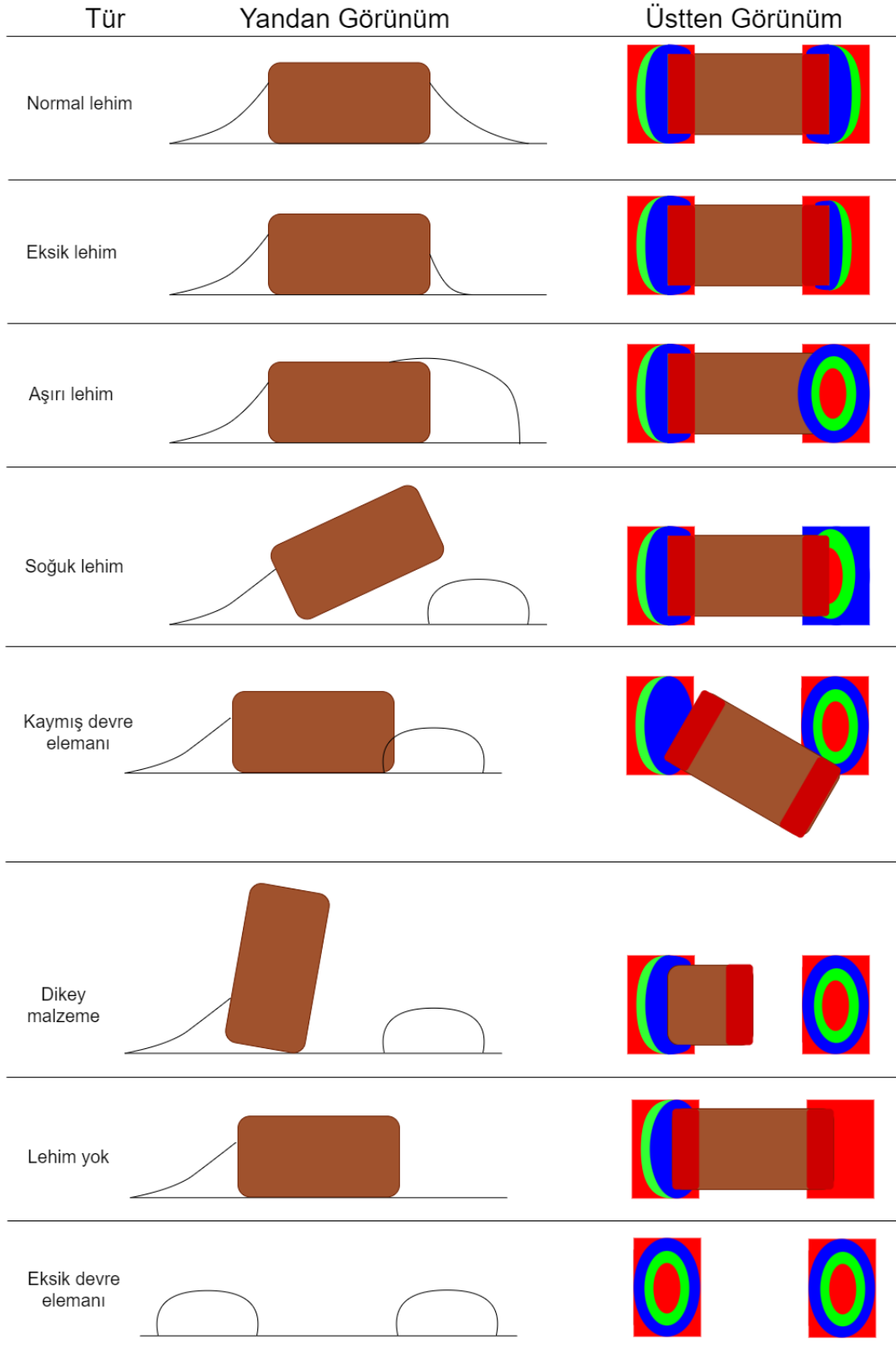
RGB aydınlatma ve renk konturlarından nitelikler çıkarma yaklaşımı endüstriyel cihazlarda da yaygın olarak kullanılır. Lehim denetimi için AOI cihazı üreten bazı şirketler: Omron, Orbotech, Viscom AG, Koh Young Technology, GOPEL electronic'dir. İlk olarak Omron şirketi, lehim denetimi için endüstriyel bir cihaz geliştirmiştir [38].



Şekil 2.4. YME lehim yüzeylerini aydınlatmada kullanılan aşamalı RGB aydınlatma ([7]'dan esinlenerek çizilmiştir).

Literatürde renk nitelikleri ile birlikte geometrik niteliklerde sıklıkla kullanılmaktadır. Geometrik nitelik olarak lehim alanını [7, 39], çevresi, uzunluğu, yüksekliği, genişliği [36, 40], kutup işaret alanı [39] belirlenir. Aynı zamanda lehim eğriliği ve durağanlığı belirleyici nitelik olarak kullanılmaktadır [41]. Entegre bacak ve lehim pedlerinden çıkarılan renk dağılımları ve lehim alanları, ağırlık merkezi, piksel süreklilikleri lehim hatalarını sınıflandırmada kullanılır [5, 7, 39]. Hongwei vd. [6], her bir renk kanalının ağırlığı, alanı, sürekliliği ve zıtlık oranını kullanır. Bir başka çalışmada ise geometrik niteliklere ek olarak dalgacık nitelikleri kullanımının başarı oranını arttırdığı gözlemlenmiştir [22].

Literatürde sıklıkla kullanılan aşamalı RGB aydınlatmanın yanı sıra aşamalı aydınlatmaya ihtiyaç duymadan denetim amaçlayan çalışmalar da vardır [22, 29, 42–44]. Chiu vd. [42], tek kademeli dairesel aydınlatma ile pasif devre elemanı lehimlerini aydınlatır. Lehim yüzeyindeki parlak bölgelerin dağılımı [42, 43], renk bölütlemesi, maskeleye [44] lehim hata tipini belirlemede kullanılmaktadır. Çeşitli dönüşümler kullanarak farklı nitelikler de elde edilebilmektedir. Acciani vd. [22], referans görüntü ile test görüntüsünün dalgacık nitelikleri arasındaki ilintiyi kullanır. Mar vd. [29] ise lehimleri sınıflandırmada Gabor niteliklerinin dalgacık niteliklerinden daha başarılı sonuç verdiğini gösterir. Bu yaklaşımlar, özel bir aydınlatma gerektirmeden lehim denetimini sağlayabilir, ancak önceden belirlenmiş niteliklere kıyasla daha çok zaman ve hesaplama kaynağına ihtiyaç duymaktadırlar.



Şekil 2.5. Dairesel RGB ışık altında çeşitli lehim hataları ([7, 8]'den esinlenilerek çizilmiştir).

2.2.3. Sınıflandırıcı

Yüzey montaj elemanlarının ilgili yerlerinden nitelik çıkarımı sonrasında çıkarılan nitelikler, lehimin hata tipini belirlemek için bir sınıflandırıcıya sokulur. Literatürdeki çalışmalar birçok farklı sınıflandırıcı kullanmaktadır: Çok katmanlı perceptron (sinir ağı) [8, 14, 22, 37, 43], naif Bayes sınıflandırıcı [33, 37], Destek Vektör Makinesi (SVM) [33, 34, 45], karar ağacı (decision tree) [6], vektör nicemleme (learning vector quantization) [7, 35, 46, 47] gibi. Bazı çalışmalarda [33, 37] ise tek sınıflandırıcının hata tipini belirlemede yeterli olmadığı durumlarda başarı oranını arttırmak için iki aşamalı sınıflandırıcı önerilmektedir.

2.3. Derin Öğrenme Bazlı Yöntemler

Son yıllardaki çalışmalarda derin sinir ağları bazlı yaklaşımların, geleneksel nitelik çıkarımı bazlı yöntemlere göre daha başarılı olduğu gösterilmektedir [48]. Bu sebepten literatürde, lehim denetimi probleminde de birçok derin öğrenme bazlı yaklaşımlar uygulanmaktadır. BDK denetiminde kullanılan derin öğrenme tabanlı yaklaşımların özeti Tablo 2.2'de verilmektedir.

Cai vd. [49], entegre lehiminde denetim yapılacak bölgeyi tespit etmek ve lehimin hatalı olup olmadığına karar vermek için evrimsel sinir ağları (CNN) kullanmaktadır. CNN bazlı yaklaşımın, modelleme bazlı yaklaşımlar ve geleneksel nitelik çıkarımı bazlı yaklaşımlardan daha başarılı olduğunu gösterir. Dai vd. [50] ise lehim bölgesi tespitinde You Only Look Once (YOLO) algoritması ve etiketli veri sayısını arttırmak için aktif öğrenme önermektedir. Wu vd. [51], Mask R-CNN algoritması kullanarak ortak bir mimaride hem lehimin tespitini hem de sınıflandırmasını amaçlar. Daha sonra Faster RCNN ve YOLO modellerini birlikte kullanarak hata tespit başarısının artırılması amaçlanmaktadır [52].

Gözetimli sinir ağların eğitiminde hem hatalı hem hatasız etiketli lehim görüntüleri kullanılmaktadır. Ancak lehim denetimi probleminde hatalı lehim görüntüleri elde etmek zordur. Ayrıca lehim hataları çok farklı şekillerde oluşabildiğinden tüm hata tiplerini kapsayıcı bir veri kümesine ihtiyaç vardır. Bu sebepten bazı çalışmalar, probleme anomali

tespiti olarak yaklaşmaktadır [1, 53–55]. Mujeeb vd. [53] hatasız lehim görüntülerini öğrenmede Lineer Özkodlayıcı kullanmaktadır. Test görüntülerinin anomali derecesi, test ve referans görüntülerin nitelik vektörleri arasındaki Euclidean mesafesi ile hesaplanmaktadır. Referans görüntüye yakın olan test görüntüleri hatasız lehim, uzak olan görüntüler ise hatalı olarak değerlendirilir. Bu çalışmada, önerilen yaklaşımın scale invariant feature transform (SIFT) gibi geleneksel nitelik çıkarma bazlı yaklaşımlardan daha başarılı olduğu raporlanır. Bir başka çalışmada [55], hatasız lehim görüntüleri modellemesi için Çekişmeli Özkodlayıcı, test görüntülerinin anomali derecesini belirlemede Mahalanobis mesafesi kullanılmaktadır. Zheng vd. [54], krem lehim uygulamasındaki farklılıklar sebebiyle çoklu-görev öğrenimi (multitask learning) yaklaşımını önermektedir. Bu yaklaşım, mevcut problemin çözümü için farklı problemlerden elde edilen çıkarımları kullanır. Bu sayede lehim denetimi, daha genel bağlamlarda yapılabilir. Derin üretken modeller de hatasız lehimleri modelleme ve anomali tespitinde kullanılabilir. Li vd. [56], Least-squares GAN (LS-GAN) yaklaşımını kullanarak hatasız entegre lehimlerini modellemektedir. Hatalı ve hatasız görüntü arasındaki piksel farkları anomali puanı olarak kullanılır. Ulger vd. [1] ise hem entegre hem entegre olmayan lehimleri modellemede β -VAE kullanmaktadır. Test görüntülerine anomali puanı ataması için ise hem aktivasyon hem de gradyan bazlı yöntemler önermektedir.

Literatürdeki lehim denetimi yöntemlerinin birbirine göre avantaj ve dezavantajları vardır. Nesne tespit algoritmalarını kullanan yöntemler, hem hatalı lehim/devre elemanını tespit edip [50] hem de sınıflandırabilmektedirler [51, 52]. Diğer yöntemler gibi ön bir bölütleme ihtiyacı duymadan BDK üzerinde hatayı tespit edebilmeleri uçtan uca denetim imkanı sağlamaktadırlar. Yalnızca hatasız lehim kullanarak gözetimsiz öğrenim yapan yöntemler [1, 13, 25–28, 53, 57], elde edilmesi zor olan hatalı lehim verisine ihtiyaç duymadan denetim sağlarken gözetimli modeller kadar yüksek başarı oranına erişemezler. RGB aydınlatma kullanarak yansıtıcı lehim yüzeylerinden nitelikler çıkaran [5, 5–8, 13, 26, 27, 33, 34, 36, 37, 39, 41, 45, 47, 58, 58, 59, 59, 60] veya derin modeller [49–51, 53, 56] kullanan yaklaşımlar literatürdeki çalışmaların büyük çoğunluğunu oluşturmaktadır. Günümüzdeki endüstriyel AOI cihazlarında da sıklıkla kullanılan küresel/dairesel RGB aydınlatma özel bir donanım gerektirmektedir.

Tablo 2.1. Denetlenen hata tipleri, aşamalı RGB aydınlatma kullanımı, işleme yöntemleri, nitelikler ve sınıflandırıcılara göre literatürdeki lehim denetimi yöntemleri.

Çalışma	Lehim hataları	RGB	İşleme yöntemleri	Nitelikler	Sınıflandırıcılar
[36]	Hatasız, yetersiz, aşırı, eksik lehim, kayıp bacak, ısınmamış ped ve bacak	✓	Pencereleme, eşikleme, Katar uzunluğu kodlaması	Alan, çevre, dairesellik, ağırlık merkezi	Karar ağacı
[8, 33]	Hatasız, soğuk, yetersiz lehim, kaymış ve yönlü malzeme, dikey malzeme	✓	Entropi[24], genetik algoritma[23]	Ortalama gri değeri, aydınlatma yüzdesi, taslak eşleştirme	SVM ve Bayes [24], MLP[23]
[45]	Aşırı,eksik, yetersiz, hatasız lehim	✓	NS	Ortalama gri değeri, aydınlatma yüzdesi	SVM
[6]	Hatasız, eksik, kaymış, soğuk ve aşırı lehim	✓	Alt bölgelere ayırma, Adaboost	Renk alanı, ağırlığı, sürekliliği ve zıtlık oranı	İkili karar ve bağlanım (regression) ağacı
[58]	Hatasız, eksik, ve aşırı lehim	✓	Çapraz ve oto ilinti	Aydınlatılmış örüntü alanları ve konumları	MLP
[39]	Kaymış, eksik, yetersiz, soğuk lehim, dikey, eksik, polarite hatalı YME	✓	Alt bölgelere ayırma	Yerleşim, mantıksal ve şekil nitelikleri	NS
[7]	Hatasız, soğuk, eksik, aşırı lehim, eksik, kaymış, dikey malzeme	✓	Alt bölgelere ayırma	Lehim alanı ve dağılımı, ağırlık merkezi, renk değişimi	LVQ
[41]	Yetersiz, aşırı, eksik ve hatasız lehim	✓	Alt bölgelere ayırma	Ortalama gri değeri, aydınlatma yüzdesi, yüzey alanı ve eğrisi	k-Nearest Neighbour (kNN)
[34]	Hatasız, yanlış, eksik, dönmüş, dikey malzeme, kaymış, aşırı, soğuk lehim	✓	Genetik algoritma	Ortalama gri değeri, aydınlatma yüzdesi	SVM
[43]	Hatalı / hatasız	-	Bölütleme, eşikleme	Lehim alanı, çevresi, varyans	MLP
[23]	Eksik, hizalanmamış, yönlü, yanlış devre elemanı, kötü lehim	-	Parlaklık ve boyut normalleştirme, pencereleme	Normalleştirilmiş standart sapma ve hacim, gri değerlerin piksel sıklığı	NS
[21, 24]	Eksik, hizalanmamış, yönlü malzeme	-	Köşe tespiti, genetik algoritma [21], Ayrık Kosinüs Dönüşümü (DCT) [24]	Taslak eşleştirme	-
[4, 14, 15]	Yerleşim hataları	-	Bölütleme, görüntü çakıştırma[4]	Görüntü çıkarma	MLP [14]
[29, 61]	Hatasız, yetersiz, aşırı lehim	-	Aydınlatma normalleştirme, bölütleme	DCT, Gabor, Dalgacık nitelikleri	Mahalanobis mesafesi
[35]	Hatasız, yetersiz, aşırı	-	Çoklu açıdan görüntüleme	-	LVQ
[5]	Soğuk, eksik, yetersiz, kaymış lehim, kısa devre, kalkmış, bükülmüş bacak	✓	Bölütleme	Şekil, dijital, mantıksal nitelikler	NS
[22]	Hatasız, yetersiz, aşırı	-	Bölütleme	Geometrik, dalgacık, ilinti	MLP
[46, 47]	Hatasız, yetersiz, aşırı lehim	-	-	-	LVQ [46, 47], Bulanık kural-bazlı sınıflandırıcı[46]
[37, 59]	Hatasız, eksik, yetersiz, aşırı lehim	✓	Eşikleme, bölütleme [37]	Ortalama gri değeri [37], aydınlatma yüzdesi, eğim açıları[37, 59]	MLP, Bayes sınıflandırıcısı [37]
[28]	Hatasız / hatalı lehim	-	Boyut indirgeme, YIQ renk uzayı	Renk nitelikleri	k-NN
[26, 60]	Hatasız / hatalı lehim	✓	HSV renk uzayı	Gauss Karışım Dağılımı [60], arkaplan çıkarma [26]	Sıklık haritası
[27]	Hatasız / hatalı lehim	✓	HSV renk uzayı	Ön konum bilgisi	Eşikleme
[13]	Hatasız / hatalı lehim	✓	Adaptif taslak üretimi, HSV renk uzayı	Görüntü çıkarma	Ağırlıklandırılmış pikseller

Tablo 2.2. Denetlenen hata tipleri, aşamalı RGB aydınlatma kullanımı, işleme yöntemleri, derin öğrenme yöntemine göre literatürdeki lehim denetimi yaklaşımları.

Çalışma	Lehim hataları	RGB	İşleme yöntemleri	Derin model
[49]	Hatalı / hatasız	✓	Bölütleme	CNN
[51]	Hatalı, hatasız, kaymış, dikey, eksik devre elemanı	✓	NS	Mask-RCNN
[52]	Kısa devre, aşırı/eksik lehim, delik lehim, lehimde kirlilik	-	BDK çıkarımı ve parsellemesi	Faster RCNN, YOLO, FPN
[50]	Hatalı / hatasız	✓	Aktif öğrenme	YOLO, CNN
[53]	Hatalı / hatasız	✓	Bölütleme	Lineer Özkodlayıcı
[55]	Hatalı / hatasız	-	NS	Çekişmeli Özkodlayıcı
[56]	Hatalı / hatasız	✓	İkili görüntü çıkarma, eşikleme	GAN, kodlayıcı

2.4. İnce Taneli Sınıflandırma

İnce taneli sınıflandırmada temel zorluk, farklı sınıfların birbirine görsel olarak oldukça benzer olması ve genel sınıflandırmanın aksine sınıf içi görsel varyasyonun fazla olmasıdır. Nitelikleri uçtan uca kodlayarak sınıflandırmak, literatürde geçerli bir yaklaşımdır. Uçtan uca nitelik kodlama yöntemi, sınıflar arası göze çarpmayan farklılıkları yakalayabilecek bir temsil öğrenmeyi amaçlamaktadır. Bunun için aşırı güvenli tahminleri cezalandırıp entropi artırılarak model geliştirilmesi artırılabilir [62]. Literatürde model tahminlerindeki aşırı güveni azaltıcı özgül kayıp fonksiyonları ile bu yaklaşım gerçekleştirilmektedir [63, 64]. Model eğitimi sırasında amaç, etiketlerin olabilirliğini maksimize etmektir. Bu işlem, model ile etiketlerin olasılıksal dağılımları arasındaki çapraz-entropiyi minimize ederek gerçekleştirilir ancak aşırı uyumlamaya (overfitting) sebep olabilir. Literatürde model gelişmesini arttırmak için çeşitli yöntemler önerilmektedir. Etiket düzleştirici düzenleme (label smoothing regularization), tahmin edilen etiket dağılımlarının tekdüze

dağılımdan sapmasını cezalandırmaktadır [63]. Dubey vd. [65], sınıfsal dağılımlar üzerindeki aşırı güveni cezalandırmanın (confidence penalty) ince taneli sınıflandırma başarısını arttırdığını göstermektedir. Mukhoti vd. [66] ise odak kaybının (focal loss) aşırı güveni engellemek için model tahminleri entropisinin arttırabildiğini gözlemler. Bu sayede aşırı uyumlama azaltılıp model genelleşme başarımı arttırılarak ince taneli sınıflandırmada fayda sağlamaktadır. Meister vd. [67], genelleştirilmiş entropi-düzenleştirilmesi ismiyle çarpık Jensen Shannon ıraksamasından (JSD) alternatif bir entropi-düzenleştirme yöntemi türetmektedir. çarpık Jensen Shannon ıraksaması (JSD) bazlı entropi-düzenleştirme yönteminin doğal dil üretiminde, α parametresini ayarlayarak etiket düzleştirme ve aşırı güveni cezalandırma yöntemlerinden daha başarılı olduğunu göstermektedir.

2.5. Açık Veri Kümeleri

BDK üzerindeki devre elemanlarının türüne göre lehimlerin şekil ve boyutları oldukça değişkendir. Bu sebepten nitelik çıkarımı veya model eğitimi sonucu genelleşebilen bir yaklaşım üretebilmek için birçok hatalı veya hatasız lehim görüntüsüne ihtiyaç vardır. Tespit edilecek veya denetlenecek devre elemanına göre ihtiyaç duyulan veriler değişkenlik gösterebilse de, bu bölümde yalnızca BDK görüntülerini tümüyle paylaşan çalışmalara yer verilmektedir. Literatürde hatasız lehimleri içeren BDK görüntülerine sahip veri kümeleri olmasına karşın hatalı lehim içeren BDK görüntülerine sahip kapsamlı, açık bir veri kümesi yoktur.

Literatürde BDK görüntüleri içeren ilk açık veri kümelerinden biri Pramerdorfer vd. [68] tarafından paylaşılmıştır. Bu veri kümesi, DSLR kamera ile çekilmiş 4928×3280 çözünürlükte 165 adet BDK görüntüsü içermektedir. Bu çalışmada, BDK üzerindeki YME'ler arasından yalnızca entegreler etiketlenir. Bu etiketlemeler, BDK üzerinde entegrelerin tespitinde kullanılmaktadır. Benzer bir çalışmada da [69] yalnızca entegreler etiketlenerek entegrelerin tespiti amaçlanmaktadır. Bir diğer çalışma da [70] ise BDK üzerindeki devre elemanları tespiti ve sınıflandırması için 48 adet BDK üzerindeki direnç, sığaç, led, anahtar, bağlantı elemanı, bobin, entegre devre elemanları etiketlenmiştir. Lu vd.

[71], yüksek çözünürlüklü (8256×5504) BDK görüntüleri üzerinde entegre, sığaç, direnç, bobin, transistör ve diyot devre elemanları ve üzerlerindeki metni etiketleyerek BDK devre elemanı sınıflandırmasında kullanılmaktadır. Benzer şekilde [72] çalışmasında sığaç, direnç, entegre, diyot, transistör gibi devre elemanları ve metinleri etiketlenmiştir. Etiketlenmiş görüntüler, devre elemanlarının yerleşim hatalarını belirlemede kullanılır. Ek olarak, tespit edilen devre elemanları üzerindeki yazılardan yazı hatalarını bulmak amaçlanır. Jesserun vd. [73] ise 73 adet BDK görüntüsü üzerindeki birçok farklı türde devre elemanı ve bu devre elemanları üzerindeki metinleri etiketler. Diğer çalışmaların aksine daha kesin ve hassas bölütleme için sınırlayıcı kutular ile dikdörtgen etiketleme yerine çokgen etiketleme de yapmaktadır.

Fridman vd. [74] ise diğer çalışmalardan farklı olarak BDK'ları referans ve referans kartın hatalı versiyonu olarak yayınlamıştır. Referans ve hatalı BDK çiftlerini karşılaştırarak BDK üzerindeki farklılar üzerinden hata tespiti yapmaktadır.

YME'leri yerleştirilmiş BDK'ların yanı sıra bazı çalışmalar [75, 76], çıplak kartlar üzerinde yalnızca su yollarını içeren BDK görüntülerini paylaşmaktadır. Bu veri kümelerinde farklı su yolu hata tipleri (delinmemiş bölge (missing hole), su yolunda küçük eksiklikler (mouse bite), açık devre, kısa devre, sahte bakır, pin delikleri (pinholes)) etiketlenmiştir. Bu sayede su yolu hataları tespit edilip sınıflandırılabilir.

Bahsedilen açık veri kümeleri, BDK devre elemanları tespitinde, bölütlemesinde kullanılabilir. Ancak üretim hatlarında, BDK devre elemanları üzerinde lehim hataları oldukça nadir olduğundan veya manüel olarak üretilmesi zaman alıcı olduğundan açık veri kümelerindeki hatalı lehim sayısı çok azdır. Bu sebepten literatürdeki çoğu açık veri kümesi, lehim hata tespiti için çalışma yapmaya uygun değildir. Literatürde yayınlanan açık veri kümeleri özetle Tablo 2.3'te verilmektedir.

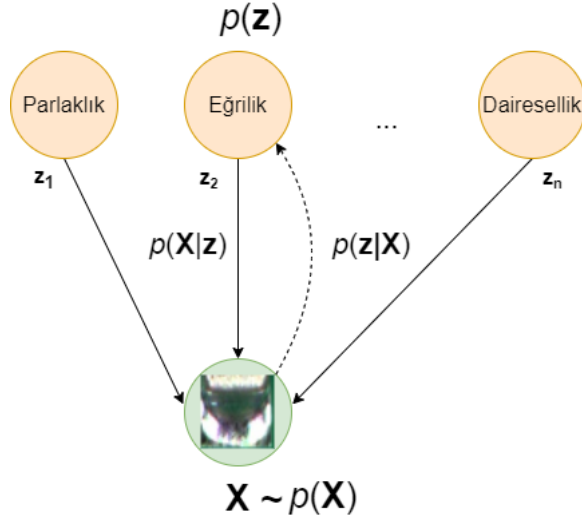
Tablo 2.3. Çalışma, görüntüleme yöntemi, çözünürlük, veri sayıları ve etiket bilgilerine göre literatürdeki açık veri kümeleri.

Çalışma	Görüntüleme yöntemi	Çözünürlük	BDK sayısı	Etiket bilgisi
[68]	DSLR	4928 × 3280	165	Entegre
[70]	DSLR/ Endüstriyel Kamera	NS	48	Direnç, sığaç, led, anahtar, bağlantı elemanı, bobin, entegre
[71]	Dijital mikroskop, DSLR	8256 × 5504	31	Sığaç, direnç, bobin, transistör, diyot, entegre, devre elemanı metinleri
[69]	Dijital mikroskop, DSLR	NS	165	Entegre
[72]	CCD Kamera	2048 × 2048	NS	Sığaç, entegre, direnç, diyot, transistör ve metinleri
[74]	Kamera	NS	20	NS
[73]	DSLR	NS	73	Çeşitli devre elemanları ve metinleri
[76]	NS	16000 × 16000	11	Açık devre, kısa devre, su yolunda eksiklik (mouse bite), sahte bakır (spurious copper), pin deliği (pinhole)
[75]	16 megapiksel HD Kamera	4608 × 3456	10	Açık devre, kısa devre, çıkıntı, küçük eksiklik (mouse bite), bakır çıkıntısı, eksik delik (missing hole)

3. TEMEL BİLGİLER

3.1. Değişimsel Çıkarsama ve Gizli Değişken Modeller

Değişimsel çıkarsama (variational inference), olasılıksal modellerin sonsal dağılımlarına optimizasyon vasıtasıyla yakınsamayı amaçlamaktadır. Değişimsel çıkarsama için gizli değişken modeller kullanılmaktadır. Gizli değişken modeller, gizli değişkenler $z =$



Şekil 3.1. Gizli Değişken Modeli ([9]'den esinlenerek çizilmiştir).

$\{z_1, \dots, z_n\}$ vasıtasıyla gözlemlenen değişkenler $\mathbf{X} = \{X_1, \dots, X_n\}$ üzerinde bir dağılım tanımlamaktadır [9]. Burada n , veri sayısıdır. Şekil 3.1'de verilen gizli değişken modelinde $p(\mathbf{z})$, gizli değişkenler üzerindeki önsel dağılımı, $p(\mathbf{X}|\mathbf{z})$, \mathbf{z} verildiğinde \mathbf{X} 'i gözleme ihtimalimizi (likelihood), $p(\mathbf{z}|\mathbf{X})$ ise \mathbf{X} verildiğinde \mathbf{z} 'nin gözlemlenme ihtimalini ifade etmektedir. Her bir veri \mathbf{X} 'e karşılık gelen bir gizli değişken (vektör) \mathbf{z} vardır. Gizli değişkenler, modelin bir parçasıdır ancak doğrudan gözlemlenemezler. Bu sebepten veri kümesinin bir parçası değildir [77, 78].

$p(\mathbf{X}, \mathbf{z})$, modelin birleşik olasılıksal dağılımını belirtir. Değişimsel çıkarsamada amaç, gözlemlenen değişkenler \mathbf{X} verildiğinde gizli değişkenlerin sonsal dağılımını $p(\mathbf{z}|\mathbf{X})$ yaklaşık olarak hesaplamaktır [79]. Sonsal dağılım, birleşik dağılım ve gözlemlerin marjinal olabilirliği (marginal likelihood) cinsinden aşağıdaki gibi ifade edilebilir:

$$p(\mathbf{z}|\mathbf{X}) = \frac{p(\mathbf{z}, \mathbf{X})}{p(\mathbf{X})}$$

Sonsal dağılımın hesaplanabilmesi için önce marjinal olabilirliğin bulunması gerekmektedir. Marjinal olabilirlik ise birleşik dağılımın, gizli değişkenler üzerinden integralinin alınması ile hesaplanır.

$$p(\mathbf{X}) = \int p(\mathbf{z}, \mathbf{X}) d\mathbf{z}$$

Ancak birçok model için bu ifade kapalı bir matematiksel formda hesaplanamaz. Bu sebepten bazı modellerde çıkarsama yapmak oldukça zordur [80]. Böyle bir modeli kullanmak/eğitmek için doğrudan marjinal olabilirliği $p(\mathbf{X})$ hesaplamak yerine değişimsel çıkarım (variational inference) sayesinde marjinal olabilirlik yaklaşık olarak hesaplanmaktadır [9]. Marjinal olabilirliğin alt sınırı Eşitlik 1'de verilmektedir. Alt sınırı hesaplayabilmek için Jensen eşitsizliğinden yararlanılır [81]. Bu ifade, evidence lower bound (ELBO) veya değişimsel alt sınır olarak da isimlendirilir. İlk terim, olabilirliğin logaritmasının (log-likelihood) ortalama değerini; ikinci terim, modelin gizli değişkenler üzerindeki dağılımı ile gizli değişkenler üzerindeki önsel dağılımı arasındaki Kullback-Leibler (KL) ıraksamasını ifade eder. ELBO, aynı zamanda 4.2.1. bölümünde bahsedilen VAE algoritmasının görev (objective) fonksiyonudur. VAE algoritması, marjinal olabilirliğin alt sınırını maksimize ederek model eğitimi yapmaktadır.

$$\begin{aligned}
p(\mathbf{X}) &= \int p(\mathbf{X}|\mathbf{z})p(\mathbf{z})d\mathbf{z} \\
&= \int p(\mathbf{X}|\mathbf{z})p(\mathbf{z})\frac{q(\mathbf{z})}{q(\mathbf{z})}d\mathbf{z} \\
&= \int p(\mathbf{X}|\mathbf{z})\frac{p(\mathbf{z})}{q(\mathbf{z})}q(\mathbf{z})d\mathbf{z} \\
\log p(\mathbf{X}) &\geq \int q(\mathbf{z})\left(\log p(\mathbf{X}|\mathbf{z})\frac{p(\mathbf{z})}{q(\mathbf{z})}\right)d\mathbf{z} \\
&= \int q(\mathbf{z})\log p(\mathbf{X}|\mathbf{z})d\mathbf{z} - \int q(\mathbf{z})\log\frac{p(\mathbf{z})}{q(\mathbf{z})}d\mathbf{z} \\
\log p(\mathbf{X}) &\geq \mathbb{E}_{q(\mathbf{z})}[\log p(\mathbf{X}|\mathbf{z})] - D_{KL}[q(\mathbf{z})||p(\mathbf{z})]
\end{aligned} \tag{1}$$

3.2. Entropi ve Çapraz-Entropi

Entropi, bir rastgele değişkenin kararsızlık ölçütüdür. Ayrık rastgele değişkenin, \mathbf{X} , entropisi Eşitlik 2’de verilmiştir. x_i , \mathbf{X} ’in durumlarını gösterir, şöyle ki, $p(\mathbf{X} = x_i) = p_i$.

$$H(p) = - \sum_i p(\mathbf{x}_i) \ln(p(\mathbf{x}_i)) \quad (2)$$

Gözetimli sınıflandırmada model dağılımı p_θ ve etiketlerin dağılımı y arasındaki çapraz-entropi $H(y, p_\theta)$, sinir ağını eğitmek için minimize edilmektedir. İkili sınıflandırma için çapraz-entropi Eşitlik 3’te verilir:

$$H(y, p_\theta) = -\ln(p_t) \quad (3)$$

$$p_t = \begin{cases} p_\theta, & \text{if } y_i = 1 \\ 1 - p_\theta, & \text{otherwise} \end{cases}$$

$y_i \in \{0, 1\}$, gerçek sınıflar etiketlerini temsil etmektedir. p_θ , örneğin, $y_i = 1$ sınıfına ait olma ihtimalidir. θ , model parametrelerini temsil etmektedir.

3.3. Boyut İndirgeme Yöntemleri

Gerçek dünya verileri birçok farklı boyuttan, niteliklerden oluşmaktadır. Boyut indirgeme yöntemleri, yüksek boyutlu gerçek dünya verilerini (görüntüler) analiz etmek amacıyla verileri genellikle 2 veya 3 boyuta indirgemede kullanılmaktadır. Böylece indirgenen veri, saçılım grafiklerinde görselleştirilebilmektedir. Boyut indirgeme sebebiyle bilgi kaybı olsa da boyut indirgeme yöntemleri, orjinal verinin bazı karakteristiklerini muhafaza etmektedirler. Bu sayede veri, görselleştirilerek anlamlandırılmış olur [82].

$\mathbf{X} = \{\mathbf{X}_1, \dots, \mathbf{X}_N\}$, yüksek boyutlu veriler, N , veri sayısı, $\mathbf{X}_i \in \mathbb{R}^h$ ise h boyutlu uzayda yer almaktadır. Boyut indirgeme yöntemleri, yüksek boyutlu h uzayından düşük boyutlu l uzayına indirgemek $DR : \mathbb{R}^h \Rightarrow \mathbb{R}^l$ için bir fonksiyon bulmayı amaçlar. İndirgenen uzay genelde 2 veya 3 boyutludur. Bölüm 3.3.1.'de literatürde sıklıkla kullanılan Temel Bileşen Analizi (PCA) ve Bölüm 3.3.2.'de t-Dağıtılmış Stokastik Komşu Gömme (t-SNE) yöntemlerinden bahsedilmektedir.

3.3.1. Temel Bileşen Analizi (PCA)

PCA, verinin daha düşük boyutlu bir uzaya lineer dikey izdüşümü olarak tanımlanabilir [83]. İzdüşüm sonrasında elde edilen alt uzaydaki verinin varyansını maksimize edecek dikey lineer dönüşümü bulmayı amaçlar. Bu dönüşüm matrisi, özvektörlerden (eigenvectors) $\mathbf{W} = [\mathbf{e}_1 \dots \mathbf{e}_k]$ oluşur. \mathbf{e}_k , k . sıradaki en büyük varyansa sahip özvektörü temsil etmektedir. İndirgenecek boyutun büyüklüğüne göre ilk l tane özvektör seçilerek dönüşüm matrisi \mathbf{W}' oluşturulmaktadır. En yüksek varyansa sahip özvektör, en yüksek özdeğere sahip olandır. Bu özvektörler temel bileşenler olarak da isimlendirilmektedir.

PCA algoritması adımları şu aşamalar ile özetlenebilir: Yüksek boyutlu orjinal verinin ortalama değerleri $\bar{\mathbf{x}}$ ve eşdeğişinti (covariance) matrisi $\mathbf{S} = \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T$ hesaplanmaktadır. Özdeğerlerine göre büyükten küçüğe sıralanmış özvektörler \mathbf{W} arasından indirgenecek uzayın boyutuna göre dönüşüm matrisi oluşturulmaktadır. Eşitlik 4'te verildiği gibi dönüşüm matrisi \mathbf{W}' ile orjinal veri \mathbf{X} eleman eleman çarpılarak indirgenmiş veri \mathbf{Y} elde edilmektedir.

$$\begin{aligned} \mathbf{Y} &= (\mathbf{W}')^T \mathbf{X} \\ &= \begin{bmatrix} e_{11} & \dots & e_{1n} \\ \vdots & \vdots & \vdots \\ e_{l1} & \dots & e_{ln} \end{bmatrix}_{l \times n} \begin{bmatrix} x_{11} & \dots & x_{1h} \\ \vdots & \vdots & \vdots \\ x_{n1} & \dots & x_{nh} \end{bmatrix}_{n \times h} \end{aligned} \quad (4)$$

PCA, böylece orjinal niteliklerin lineer kombinasyonlarını kullanarak yeni nitelikler çıkarır. Normal ve anomali lehim görüntülerinin PCA ile 2 boyuta indirgenmiş bir örneği Şekil 5.5'te verilmiştir. Ancak PCA, orjinal niteliklerin lineer bir kombinasyonunu bulduğundan daha doğru bir temsil, lineer olmayan bir kombinasyon ile bulunabilmektedir.

3.3.2. t-Dağıtılmış Stokastik Komşu Gömme (t-SNE)

t-SNE [84], PCA'ın aksine lineer olmayan bir boyut indirgeme yaklaşımıdır. t-SNE, veriler arasındaki yerel benzerlikleri boyut indirgeme sırasında korumayı amaçlar ve rastgelelik kuramına göre indirgeme yapmaktadır. Düşük boyuttaki temsil, yüksek boyutlu orjinal veri ile düşük boyuttaki temsilin olasılıksal mesafesi arasındaki Kullback-Leibler (KL) iraksaması, gradyan inişi algoritması ile minimize edilerek elde edilmektedir. Yüksek boyuttaki birleşik olasılıksal dağılım P , ve düşük boyutlu uzaydaki dağılım, Q , arasındaki KL iraksaması Eşitlik 5'te verilmiştir. Böylece veriler arasındaki yerel yapı korunurken yüksek boyuttaki dağılım, P , ile düşük boyuttaki temsil, Q , birbirine olabildiğince benzer hale gelmektedir.

$$D_{KL}(P||Q) = \sum_{i=1}^n \sum_{j=1}^n p_{ij} \ln\left(\frac{p_{ij}}{q_{ij}}\right) \quad (5)$$

p_{ij} , yüksek boyutlu uzayda i ve j veri çiftleri arasındaki benzerliği ifade eden olasılıksal dağılımı temsil etmektedir. \mathbf{X}_i ve \mathbf{X}_j , düşük boyutlu uzaydaki veri noktalarıdır.

$$p_{ij} = \frac{\exp(-\|\mathbf{X}_i - \mathbf{X}_j\|^2/2\sigma^2)}{\sum_{k \neq l} \exp(-\|\mathbf{X}_k - \mathbf{X}_l\|^2/2\sigma^2)}$$

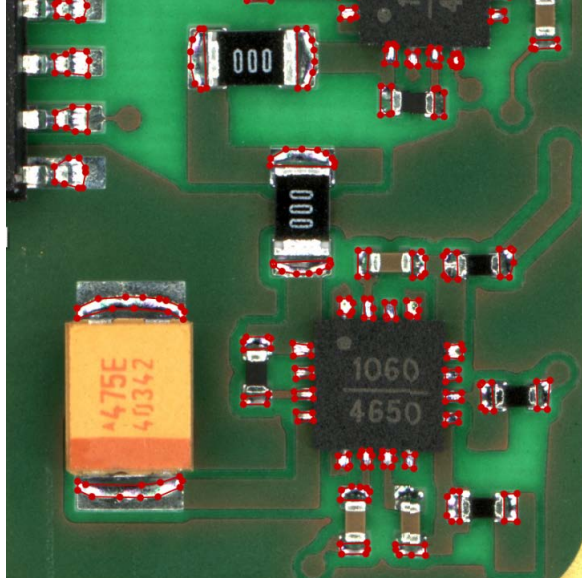
Benzer şekilde q_{ij} düşük boyutlu uzaydaki (2 veya 3 boyutlu), veri çiftleri i, j arasındaki benzerlik aşağıdaki ifade ile tanımlanmaktadır. \mathbf{Y}_i ve \mathbf{Y}_j , düşük boyutlu uzaydaki veri noktalarıdır.

$$q_{ij} = \frac{(1 + \|\mathbf{Y}_i - \mathbf{Y}_j\|^2)^{-1}}{\sum_{k \neq l} (1 + \|\mathbf{Y}_k - \mathbf{Y}_l\|^2)^{-1}}$$

Boyut indirgemedede önemli bir parametre karışıklıktır (perplexity). Karışıklık, genel ve yerel karakteristikler arasındaki dengenin nasıl ayarlanacağını belirleyen bir parametredir. Bu parametre, her bir verinin yakın komşu sayısını gösteren bir parametredir. Pratikte düşük sayıda veriler için karışıklık parametresi düşük, yüksek sayıda veriye sahip kümeler için ise parametrenin yüksek seçilmesinin daha uygun olduğu gözlemlenmektedir [85]. t-SNE ile 2 boyuta indirgeme örneği Şekil 5.3 ve 5.4'te verilmiştir.

4. ÖNERİLEN YÖNTEMLER

BDK üzerindeki lehimlerin denetlenmesi temelde 2 aşamadan oluşur: lehimlerin bölütlenmesi ve sınıflandırılması. BDK üretim hatlarında denetim yapan AOI cihazları bir tepe kamerasından görüntüler alır. Ancak ham BDK görüntüleri doğrudan lehimlerin denetlenmesi için yeterli değildir. Bu sebepten öncelikle BDK üzerindeki her bir lehimini bölütleriz. Lehim bölütleme yaklaşımından Bölüm 4.1.'de bahsedilmiştir. Bölütlenen lehimler üzerinde denetim yapmak amacıyla iki ayrı yaklaşım önermekteyiz. İlk yaklaşım, model eğitiminde hatalı lehime ihtiyaç duymadan derin üretken modeller ile hatasız lehimini modellemektir. Model başarımını artırılması için ise lehim verisinin gizli değişken faktörleri için çözülmüş temsiller öğrenmeyi amaçlarız. Daha sonra test edilen lehimlere anomali puanı atanarak anomali derecesini belirler ve sınıflandırma yaparız. Derin üretken model ile önerilen anomali tespiti yaklaşımından Bölüm 4.2.'de bahsedilmiştir. Önerilen bir diğer yaklaşım ise hatalı ve hatasız lehim görüntüleri birbirine oldukça benzer olduğundan ince taneli sınıflandırmadır. İnce taneli sınıflandırma için çarpık JSD ile entropi düzenlileştirilmesi yaklaşımını önermekteyiz. Önerdiğimiz Çarpık JSD ile düzenlileştirme yaklaşımı, lehimleri sınıflandırmada literatürdeki diğer yaklaşımlardan daha başarılı olduğunu göstermektedir. İnce taneli sınıflandırma için entropi düzenlileştirme tabanlı yaklaşımlar Bölüm 4.3.'te verilmiştir.

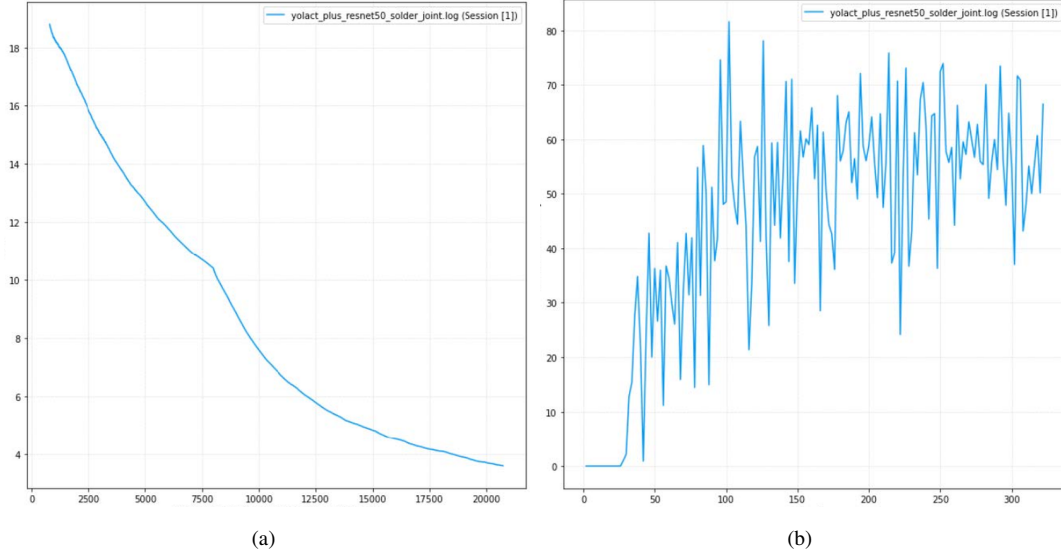


Şekil 4.1. BDK üzerinde çokgen etiketleme örneği.

4.1. Lehim Bölütlemesi

Lehim bölütlemesi için kullanılan veri kümesinde toplam 3000'den fazla lehim içeren 7 adet BDK kullanılmıştır. BDK'lar 480×640 'lık görüntü parçalarına bölünmüştür. Bu görüntüler model eğitimi, doğrulaması ve testi için kullanılmıştır. Bu bölünmüş görüntüler üzerinde Labelme [86] kullanılarak yaklaşık 1000 lehime çokgen etiketleme yapılmış ve model eğitiminde kullanılmıştır. Yaklaşık 120 adet lehim ise doğrulama kümesinde, kalan BDK görüntü parçaları test veri kümesinde kullanılmıştır. Bir BDK görüntü bölütü üzerindeki etiketleme örneği Şekil 4.1'de verilmiştir.

BDK üzerindeki lehimlerin bölütlenebilmesi amacıyla semantik bir öge bölütleme yaklaşımı olan You Only Look at Coefficients (YOLACT) [87] modeli kullanılmıştır. Diğer semantik bölütleme yaklaşımlarındaki gibi öncelikle girdi görüntüleri üzerinde Öznitelik Piramit Ağı (Feature Pyramid Network) [88] kullanarak nitelikler çıkarılmıştır. Sonrasında çıkarılan nitelikler, bölütleme amacıyla iki dalda kullanılır. İlk dal, Tam Evrişimsel Ağ (Fully Convolutional Network) [89] kullanarak tüm görüntü üzerinde, lehimleri gösteren prototip maskeler üretmeyi amaçlar. İkinci dal ise üretilmiş her bir maskenin sınırlayan kutularını ve sınıf tahminlerini çıkarır. Düşük tahminler içeren sınırlayıcı kutular maksimum olmayan

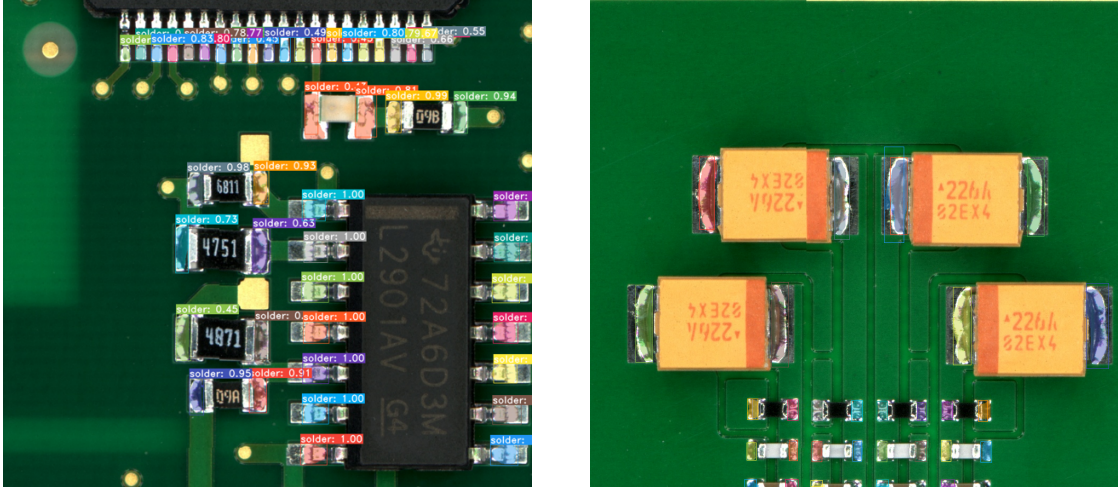


Şekil 4.2. YOLACT modelinin (a) eğitim ve (b) doğrulama kümesinde başarıım oranı.

bastırma (non-maximum suppression) ile elenir. Son olarak dallar lineer olarak birleştirilerek girdi görüntüsü üzerindeki son maske çıktıları elde edilmiştir.

Etiketleme sonrası etiketlenen lehim görüntüleri kullanılarak YOLACT modeli eğitilmiştir. Şekil 4.2 (a)'da gösterildiği gibi eğitim sırasında model, lehimleri temsil eden maskeler üretmeyi öğrenir ve gerçek referans değerlere daha yakın tahminler yaptıkça eğitim kaybı azalır. Eğitim sırasında modelin gerçek referans değerlere aşırı uyumlamadan (overfitting) optimum hale gelebilmesi için doğrulama kümesindeki model performansı gözlemlenmiştir. Doğrulama kümesinde başarıım oranının azaldığı veya dengeye ulaştığı durumda eğitim durdurulmuştur. Modelin doğrulama kümesi için elde edilen başarıım grafiği Şekil 4.2 (b)'de verilmiştir.

Son olarak eğitilen model, test görüntülerinde kullanılarak lehimlerin bölütlenmesi sağlanmış ve Bölüm 4.2. ve 4.3.'te kullanılan veri kümesi elde edilmiştir. Bölütlenmiş lehimlerden oluşan veri kümelerinden Bölüm 5.1.1. ve Bölüm 5.2.1.'de detaylıca bahsedilmiştir. Veri kümesinde bulunan BDK görüntülerinde yeterli sayıda hatalı lehim mevcut olmadığından, YOLACT modeli yalnızca hatasız lehimlerin bölütlenmesinde kullanılmıştır. Hatalı lehimler ise manuel olarak bölütlenmiştir. YOLACT modelinin test veri kümesi üzerinde bir bölütleme örneği Şekil 4.3'te verilmiştir.

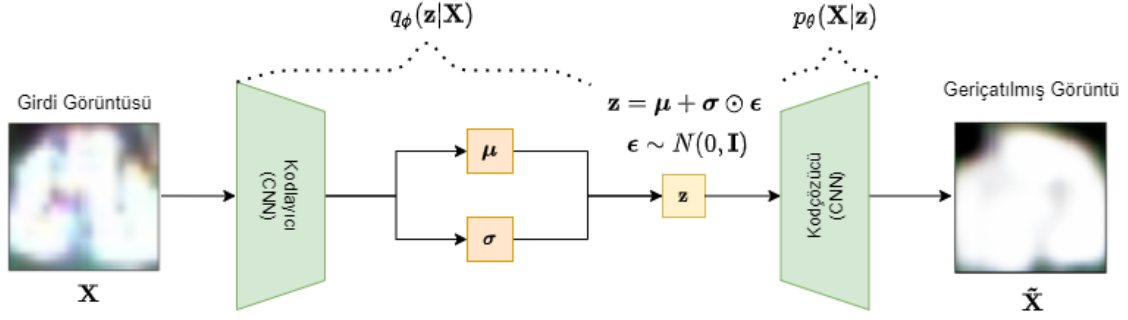


Şekil 4.3. YOLACT modelinin test kümesindeki lehim bölütlemelemleri.

4.2. Derin Üretken Modeller

4.2.1. Değişimsel Özkodlayıcı (VAE)

VAE, \mathbf{X} 'in yüksek boyutlu uzayda bir veri noktası (görüntü) olduğu yerde tümleştirilmiş olabilirlik (marginal likelihood) $p_{\theta}(\mathbf{X})$ 'i maksimize eden üretken bir modeldir. Şekil 4.4'te gösterildiği gibi olasılıksal bir kodlayıcı (encoder) ve kodçözücünden (decoder) oluşmaktadır. Ağın ağırlıkları (weight) ve varsayımlarını (bias) içeren ϕ ve θ , bu kodlayıcı ve kodçözücünün parametreleridir. Kodlayıcı, dağılımı Gaussian olduğu varsayılan gerçek sonsal (posterior) dağılımı $p_{\phi}(\mathbf{z}|\mathbf{X})$ tahmin etmeye çalışır. \mathbf{z} , gizli (latent) değişkenler vektörüdür. Tahmin edilen sonsal dağılımın $q_{\phi}(\mathbf{z}|\mathbf{X})$ ortalama değeri ve standart sapması, kodlayıcının çıktısıdır. Ayrıca, gizli değişkenler \mathbf{z} üzerindeki önsel (prior) dağılımın standart Gaussian dağılıma $p(\mathbf{z}) = N(0, \mathbf{I})$ sahip olduğu varsayılır. Daha sonra \mathbf{z} , sonsal dağılımdan örneklenir $\mathbf{z} \sim q_{\phi}(\mathbf{z}|\mathbf{X}) = N(\mu, \sigma)$. Kodçözücü, \mathbf{z} 'den $\tilde{\mathbf{X}}$ 'i geriçatar (reconstruction). Ancak, örnekleme işlemi türevlenebilir olmadığından modeli eğitmek için kayıp (loss) fonksiyonu geri yayılamaz (backpropagation). Bu sebepten bir parametrelendirme yöntemi kullanılmıştır. Bu yöntem, standart Gaussian dağılımdan $N(0, \mathbf{I})$ örneklenmiş bir rastgele değişken ϵ kullanır ve bu değişkeni ortalama değer kadar kaydırıp standart sapma ile



Şekil 4.4. VAE mimarisi ve gizli değişken vektörü üzerindeki önsel izotropik Gaussian dağılım [1].

ölçeklendirerek ($\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}$) \mathbf{z} değişkenine çevirir. VAE mimarisi Şekil 4.4 ve Şekil 5.1’de verilmiştir. VAE’yi eğitirken amaç, bir verinin tümleştirilmiş olabilirliğinin alt sınırını (değişimsel alt sınır (ELBO)) maksimize eden θ ve ϕ parametrelerini bulmaktır. Bu ifade Eşitlik 6’da verilmiştir.

$$\max_{\theta, \phi} \log p_{\theta}(\mathbf{X}) \geq \max_{\theta, \phi} \mathbb{E}_{\mathbf{z} \sim q_{\phi}(\mathbf{z}|\mathbf{X})} (\log p_{\theta}(\mathbf{X}|\mathbf{z})) - \beta D_{KL}(q_{\phi}(\mathbf{z}|\mathbf{X}) || N(\mathbf{z}; 0, \mathbf{I})) \quad (6)$$

Eşitliğin sağ tarafındaki ilk terim, geriçatma kaybı (reconstruction loss) olarak ifade edilir ve \mathbf{X} girdi görüntülerini geriçatma ihtimalini maksimize etmektedir. İkinci terim ise yaklaşılansonsal dağılım ve standart Gaussian dağılım arası mesafeyi temsil eden KL iraksamasını (divergence) minimize etmektedir.

4.2.2. beta-Değişimsel Özkodlayıcı (β -VAE)

β -VAE, verinin varyasyon faktörlerini çözerek daha çözülmüş temsiller öğrenmeyi amaçlar [90]. Varyasyon faktörlerinin çözülmesiyle veriyi temsil eden her bir nitelik birbirinden bağımsız hale gelir ve model öğrenimi kolaylaşır [91, 92]. β -VAE, Eşitlik 6’da verildiği gibi β adında geriçatma ve gizli kanal kapasitesini dengeleyici bir parametre önermektedir. $\beta > 1$ iken sonsal dağılımın $q_{\phi}(\mathbf{z}|\mathbf{X})$, önsel standart Gauss dağılımına $p(\mathbf{z})$ daha yakın olması teşvik edilir [93]. Bu durum, daha çözülmüş bir temsil (disentangled representation) öğrenmek için gizli değişkenin \mathbf{z} kapasitesini sınırlandırmaktadır. β -VAE için $\beta = 1$

olduğu durum, orjinal VAE'ye eşittir. β -VAE için $\beta > 1$ olduğu durumlarda, verinin gizli uzayında daha verimli bir temsil öğrenebildiği için çeşitli veri kümelerinde VAE'dan daha başarılı olduğunu göstermektedir. Ancak, geriçatma doğruluğu ile çözülmüş temsil öğrenme arasında bir ödünleşim (trade-off) vardır. Yüksek β değeri, sınırlandırılmış gizli (latent) kanaldan geçerken bilgi kaybına neden olabileceği için daha kötü geriçatmaya neden olabilir. Bu sebepten çözülmüş temsil öğrenirken aynı zamanda girdi görüntülerini, \mathbf{X} , geriçatabilen bir denge bulmak önemlidir [90].

4.2.3. Anomali Puanı

Model eğitimleri sonrası test kümesindeki her bir örneğin, anomali derecesini gösteren anomali puanı ataması yapılmaktadır. Literatürde önerilen geriçatım tabanlı ve gradyan tabanlı anomali puanlama yöntemleri, lehim sınıflandırmasında kullanılmaktadır.

4.2.3.1. Geriçatma kaybı (reconstruction loss): Model testi sırasında test kümesindeki örnekler \mathbf{X} , eğitilmiş VAE'ın kodlayıcısına girdi olarak verilmiştir. Kodlayıcı, her bir görüntünün ortalama değer ve standart sapma vektörlerini çıktı olarak verir. Bu ortalama değer ve standart sapmaya sahip Gaussian dağılımdan örneklenmiş z gizli vektörünü elde edebilmek için parametrelendirme yöntemi uygulanmıştır. Daha sonra kodçözücü, bu dağılımdan girdi görüntülerini geriçatar. Geriçatma kaybı (reconstruction loss), anomali puanı olarak her bir test görüntüsü için girdi görüntüsü ile geriçatılmış görüntü arasında ortalama karesel hata (mean squared error) ile Eşitlik 7'de verildiği gibi hesaplanmıştır. x_i ve \tilde{x}_i , test görüntüsü \mathbf{X} 'in i numaralı piksel değerini ifade eder. N ise görüntünün toplam piksel sayısını ifade eder. Yalnızca hatasız (normal) veriler ile eğitilmiş VAE'nin, yalnızca normal görüntüleri düşük hata ile geriçatması beklenir ve düşük anomali puanı çıkarır. Ancak normal olmayan (anomali) görüntüler, yüksek hata ile geriçatılır ve yüksek anomali puanı vermektedir.

$$\mathcal{A}(\mathbf{X})_{Recon} = \frac{1}{N} \sum_{i=1}^N (x_i - \tilde{x}_i)^2 \quad (7)$$

4.2.3.2. Değişimsel alt sınır (ELBO): Görev fonksiyonu, \mathbf{X} 'in çıktıda da gözlemlenme ihtimali olduğu için anomali puanı $\mathcal{A}(\mathbf{X})_{ELBO} = -\log p_{\theta}(\mathbf{X})$ şeklinde hesaplanabilir. Verinin sürekli ve kodlayıcının Gaussian dağılıma sahip olduğu varsayıldığından KL ıraksaması, kapalı formda Eşitlik 8'in sağ tarafındaki ikinci terimde verildiği gibi hesaplanmıştır [94]. Bu ifadenin türetimi, Ekler EK 1 -'de verilmiştir.

$$\mathcal{A}(\mathbf{X})_{ELBO} = -\frac{1}{N} \sum_{i=1}^N (x_i - \tilde{x}_i)^2 - \frac{1}{2} \sum_{j=1}^J (1 + \log((\sigma_j)^2) - (\mu_j)^2 - (\sigma_j)^2) \quad (8)$$

4.2.3.3. Gradyan kısıtı (GradCon): Kwon vd. [95], normal veri dağılımlarını modelleyebilmek ve aktivasyon-bazlı temsillerde mevcut olmayan bilgiyi de anomali tespitinde kullanabilmek için özkodlayıcıları (autoencoder), gradyan kısıtı (gradient constraint) ile eğitmeyi önermektedir. Normal veriler ile eğitilmiş bir modelin, anomali verileri temsil edebilmesi için normal verilerden daha fazla parametre güncellemesi gerekmektedir. Bu sebepten geri yayılmış (backpropagated) gradyanlar hesaplanır ve girdi verisi için ne kadar parametre güncellemesi gerektirdiğine bakılarak anomali tespiti yapılabilir. Her bir kodçözücü katmanının model parametrelerine göre gradyanları $\frac{\partial \mathcal{L}}{\partial \theta_i}$, geriçatma kaybını geri yayarak hesaplanmıştır. Algoritma, geriçatma ve gradyan kayıplarının birleşimini Eşitlik 9'daki gibi anomali puanı hesabında kullanır:

$$\mathcal{A}(\mathbf{X})_{GradCon} = \mathcal{A}(\mathbf{X})_{Recons} + \gamma \mathcal{L}_{grad} \quad (9)$$

Burada γ skalerdir, \mathcal{L}_{grad} ise kosinüs benzerliği yani eğitim gradyanlarının ortalaması ile mevcut yinelemedeki (iteration) (k_{th}) gradyan kaybı arasındaki açının kosinüsüdür ve Eşitlik 10'da verilmektedir:

$$\mathcal{L}_{grad} = -E_i \left[\cos SIM \left(\frac{\partial J^{k-1}}{\partial \theta_{i_{avg}}}, \frac{\partial \mathcal{L}^k}{\partial \theta_i} \right) \right] \quad (10)$$

$$\frac{\partial J^{k-1}}{\partial \theta_{i_{avg}}} = \frac{1}{k-1} \sum_{t=1}^{k-1} \frac{\partial J^t}{\partial \theta_i} \quad (11)$$

J , model eğitimi sırasında hesaplanan kayıp fonksiyonudur, \mathcal{L} , ELBO'ya eşittir ve α ise gradyan kaybı için bir ağırlıktır, aşağıdaki gibi tanımlanır:

$$J = \mathcal{L} + \alpha \mathcal{L}_{grad} \quad (12)$$

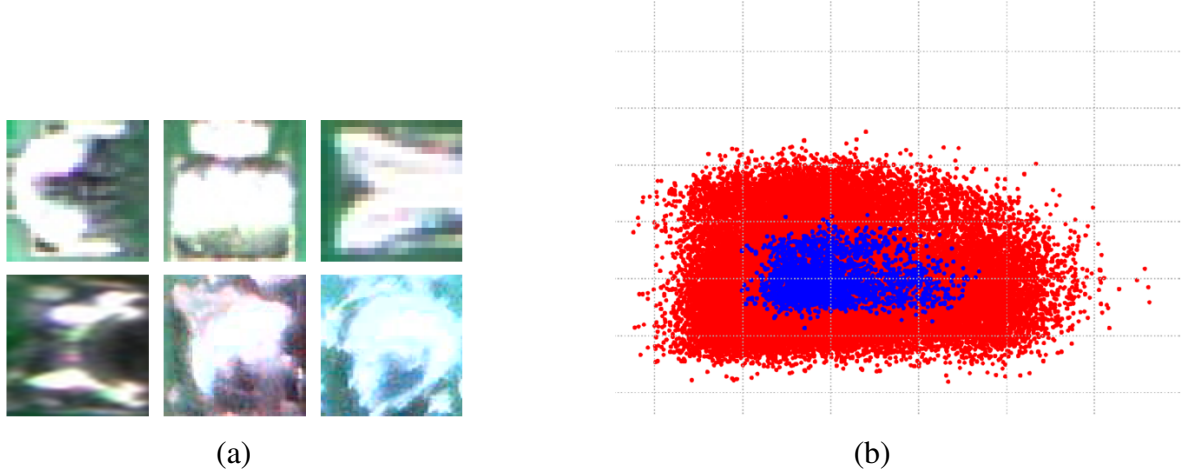
Özkodlayıcıların eğitimi için gradyan kısıtı kullanan ve geriçatma hatası ile gradyan kaybının birleşimini anomali puanı olarak kullanan anomali tespit algoritmasına GradCon adı verilmiştir.

4.2.3.4. Mahalanobis mesafesi: Çekişmeli Özkodlayıcı (AAE) için anomali puanı, gizli değişkenler üzerinden normal veri dağılımı ile her bir test örneği arasındaki Mahalanobis mesafesi ile Eşitlik 13'te verildiği gibi hesaplanmıştır. z , her bir test örneğinin gizli vektörüdür. μ ise normal verilerin gizli vektörlerinin ortalama değerler vektörüdür. Σ^{-1} , normal verilerin kovaryans matrisinin tersine eşittir.

$$T^2 = (\mathbf{z} - \mu)^T \Sigma^{-1} (\mathbf{z} - \mu) \quad (13)$$

4.2.4. Karar Mekanizması

Anomali puanı hesaplandıktan sonra bir test örneğinin anomali olup olmadığına karar vermek için örneğin anomali puanını, bir eşığe göre karşılaştıran bir karar mekanizması kullanılmıştır [96]. Normal örnekler için anomali puanının düşük, anomali örnekler için ise yüksek olması beklenir. Karar mekanizması için kullanılan eşik, bir parametre olarak görülür, doğrulama kümesinde normal verilerin ortalama anomali puanı ve puanların standart sapmalarının iki katının toplamı olarak hesaplanmıştır.



Şekil 4.5. (a) Lehim veri kümesinden örnekler: İlk satırda hatasız lehimler, ikinci satırda hatalı lehimler. (b) Mavi noktalar, VGG16 ağının son tam çizge katmanı (fully connected layer) üzerinde lehim verilerinin PCA izdüşümünü, kırmızı noktalar ImageNet örneklerini temsil eder.

4.3. İnce Taneli Sınıflandırma

Lehimlerin görsel denetimindeki temel zorluk, hatalı ve hatasız lehimler arasında görsel olarak genelde küçük farklılıklar vardır. Aynı sınıftan (hatasız/hatalı) lehim görüntüleri birbirinden oldukça farklı, farklı sınıftaki lehimler ise birbirine benzer olabildiğinden lehim denetimi problemine ince taneli sınıflandırma yaklaşımları uygulanabilir. Bazı hatasız ve hatalı lehim örnekleri Şekil 4.5 (a)'da verilmiştir.

Lehimlerin ince taneli olduğunu göstermek için jenerik görüntüler içeren ImageNet veri kümesi ile lehim verilerinin nitelik çeşitlilikleri karşılaştırılmaktadır. VGG16 ağının son tam çizge katmanı çıktısına uygulanan PCA izdüşümü ile 2 boyuta indirgenen lehim ve ImageNet örnekleri Şekil 4.5 (b)'de gösterildiği gibi çizdirilmiştir. Lehimlerin nitelik çeşitliliğinin ImageNet'e kıyasla çok daha düşük olduğu gözlemlenmiştir. Bu sebepten lehim veri kümesi ince tanelidir.

Modelin eğitim kümesinde tüm olasılığı tek bir sınıfa ataması aşırı güvenli olmasına neden olmaktadır. Aşırı güvenli model, çoğunlukla aşırı uyumlamanın (overfitting) belirtisidir [97]. Modelde aşırı güveni cezalandırıcı düzenleme terimleri eklenerek tek bir sınıfa yüksek olasılık atanması engellenir ve model genelleşme başarımı artar. Aşırı güveni

cezalandırmanın bir yöntemi entropiyi maksimize etmektir [64]. Belirli sınıflara daha yüksek olasılığa sahip dağılımların entropisi düşük iken en yüksek entropi tekdüze dağılımlar ile elde edilir. Bu sebepten ince taneli sınıflandırmada model entropisi artırılarak tekdüze dağılım teşvik edilmektedir [65]. İnce taneli sınıflandırmada model genelleştirmesini arttırmak, özel niteliklere bağlılığı azaltmak için model entropisi artırılarak aşırı güvenli model tahminlerin azaltılması teşvik edilmektedir. Bu bölümde öncelikle literatürde kullanılan entropiyi maksimize eden yöntemlerden daha sonra ince taneli sınıflandırma için önerdiğimiz çarpık Jensen-Shannon iraksamasından (skew JSD) bahsedilmektedir.

4.3.1. Maksimum Entropi Düzenliştirmesi

Maksimum entropi düzenlileştirmede amaç, maksimum entropiye sahip olasılıksal dağılım ile elde edilen en tarafsız model temsilini elde etmektir [98]. Literatürde sınıflandırıcının entropisini maksimize ederek model başarımını arttırmayı amaçlayan birçok yaklaşım mevcuttur. Model tahminleri üzerindeki olasılıksal dağılım tekdüze (uniform) olduğunda sınıflandırıcı en kararsız ve entropisi en yüksek durumundadır. Bu durumda sınıflandırıcı, sınıflar arası tahminde sınıflara aynı olasılığı atar.

4.3.1.1. Odak Kaybı (Focal loss) Yanlış sınıflandırılmış örneklere daha çok ağırlık verilmesi için çapraz-entropi kaybına bir terim ekler. Focal kayıp, Eşitlik 14’de verilmiştir:

$$\mathcal{L}_f(\theta) = -\alpha_t(1 - p_t)^\gamma \ln(p_t) \quad (14)$$

Ek olarak tahmin edilen olasılıksal dağılımının entropisinin büyük olmasını sağlar. Bu sayede daha kararsız tahminler ve daha iyi model genelleştirmesi sağlanır. $\alpha \in [0, 1]$ ve γ , skaler bir büyüklüktür.

4.3.1.2. Etiket Düzleştiren Düzenlileştirme Hedeflenen etiket dağılımını, düzleştirilmiş bir dağılım ile değiştirerek model tahminlerini daha kararsız bir hale getirmektedir. Ayrıca,

tahmin edilen dağılımın p_θ , tekdüze dağılıma u yakın olmasını teşvik ederek entropiyi maksimize eder. Görev fonksiyonu Eşitlik 15'te verilmiştir:

$$\mathcal{L}_s(\theta) = (1 - \varepsilon)H(\mathbf{y}, p_\theta) + \varepsilon H(u, p_\theta) \quad (15)$$

ε , düzleştirme terimidir, $\varepsilon \in [0, 1]$.

4.3.1.3. Maksimum Entropi Öğrenimi Model dağılımı p_θ ile doğru etiket dağılımı \mathbf{y} arasındaki KL ıraksamasını minimize eder. Aşırı emin sınıflandırmayı cezalandırıcı bir terim ekleyerek model tahminlerinin entropisini maksimize ederek daha tekdüze hale getirmektedir. Maksimum entropi öğrenimi görev fonksiyonu, Eşitlik 16'da verilmiştir:

$$\mathcal{L}_m(\theta) = D_{KL}(\mathbf{y}||p_\theta) - \beta H(p_\theta) \quad (16)$$

β , cezalandırma miktarını belirleyici bir terimdir.

4.3.1.4. Çarpık Jensen-Shannon ıraksaması ile Entropi Düzenlemesi JSD, iki olasılıksal dağılım arası benzerlik metriğidir. Bu tezde, lehimlerin ince-taneli sınıflandırması için entropi düzenlemesi olarak çarpık JSD önerilmektedir. Eşitlik 17'deki terim eklenerek model ile tekdüze dağılım arası mesafe minimize edilir ve yüksek entropi teşvik edilmektedir.

$$J_\alpha(u||p_\theta) = \frac{1}{\alpha(1 - \alpha)} [-\alpha H(p_\theta) - (1 - \alpha)H(u) + H((1 - \alpha)u + \alpha p_\theta)] \quad (17)$$

Sonuç olarak model öğreniminde minimize edilen görev fonksiyonu Eşitlik 18'de verilmiştir:

$$\mathcal{L}_{J_\alpha}(\theta) = H(\mathbf{y}, p_\theta) + \beta J_\alpha(u||p_\theta) \quad (18)$$

Jensen-Shannon ıraksamasının KL ıraksaması cinsinden ifadesi ve Eşitlik 17’de verilen çarpık JSD ifadesinin türetimi Ekler EK 2 –’de verilmiştir. $\alpha \rightarrow 0$ için çarpık JSD’yi düzenleştireci olarak kullanmak **maksimum entropi öğrenimine**, $\alpha \rightarrow 1$ ise **etiket düzleştiren düzenleştirmeye** eş değerdir. Etiket düzleştiren düzenleştirme ve aşırı özgüveni cezalandırma yaklaşımlarının çarpık JSD’den türetimi Ekler EK 4 – ve EK 5 – bölümlerinde verilmiştir.

Derin sinir ağlarında minimize edilen görev fonksiyonları çoğunlukla dışbükey olmadığından yerel minimum noktalarına sahiptir. Bu durum çarpık JSD fonksiyonu için de geçerlidir. Ekler EK 3 –’te gösterildiği gibi çarpık JSD fonksiyonu dışbükey veya içbükey bir fonksiyon değildir. Ayrıca $\alpha \in [0, 1]$ için çarpık JSD fonksiyonunun sürekli olduğu Ekler EK 3 –’te verilmiştir.

5. DENEYSEL ÇALIŞMALAR VE SONUÇLAR

Tüm deneyler, bir GPU erişim hizmeti olan Google Colab Pro [99] kullanılarak eğitilmiştir. Model eğitim süreleri modelin karmaşıklığı, kullanılan veri kümesi boyutu, verilerin çözünürlüğü ve epok sayısına göre değişmektedir. 5.1. adımında eğitilen her bir derin üretken model yaklaşık 50 dakika, 5.2. adımında eğitilen entropi düzenleştirmesi tabanlı modeller ise yaklaşık 25 dakikada eğitilmiştir. Her bir modelin en az 10 kez test edilmesi yaklaşık 20 dakika sürmektedir. Tüm model eğitimleri için PyTorch kütüphanesi [100], çeşitli yardımcı makine öğrenimi algoritmaları için ise scikit-learn [101] kütüphanesi kullanılmıştır.

5.1. Derin Üretken Modeller ile Deneysel Çalışmalar

İlk olarak kullanılan veri kümesinden ve ön işleme adımlarından bahsedilmiştir. Daha sonra VAE, β -VAE, AAE ve CAE modellerinin karşılaştırması yapılmaktadır. Evrişimsel Özkodlayıcı (CAE), girdi görüntülerinin düşük boyutlu bir temsiliyi öğrenen gözetimsiz bir modeldir [102]. CAE, düşük boyutlu ayrıştırıcı bir temsil öğrenerek normal ve anomalik

verilerin ayrıştırılmasında etkilidir. AAE ise VAE algoritması gibi gizli değişkenlere bir önsel dağılım varsayımı yapar.

5.1.1. Veri Kümesi ve Önışleme

2619 normal eğitim verisinin yüzde 10'u çapraz doğrulamada kullanılmak üzere doğrulama kümesi olarak ayrılmıştır. Test verisi ise 66 anomali, 117 hatasız veriden oluşmaktadır. Bazı anomali verileri açık kaynaklı veri kümesinden elde edilmiştir [71]. Veri kümesindeki bazı anomali (hatalı lehim) örnekleri: kısa devre, yetersiz lehim, aşırı lehim, pin delikleri (pinhole), lehimin ıslatmaması (solder dewetting), sünmüş lehim (solder flags), reçine kalıntıları, kayıp ve kaymış devre elemanı. Anomali tespiti başarısı değerlendirilirken referans değer etiketleri yalnızca test esnasında kullanılmıştır. Veri kümesi, birçok BDK'nın hem entegre hem de entegre olmayan (pasif devre elemanı) devre elemanlarının lehim pedleri bölütlenerek elde edilmiştir. Lehimler, manuel olarak LabelMe yazılımı ile etiketlenmiştir [86]. Son olarak veri kümesi, bölütleme hataları, hatalı veya net olmayan görüntülere karşı kontrol edilmiştir. Önışleme kapsamında veri, lehimlerin bütünlüğünün bozulmaması için kısa kenardan kesilmiş ve 64×64 'lük boyutlara indirgenmiştir. Sonrasında veri kümesi, $[-1, 1]$ değerleri arasında değerler alabilmek için sıfır ortalama değer ve bir varyans değerine normalize edilmiştir. Aşırı uyumlamayı (overfitting) azaltmak, veri çeşitliliğini arttırmak amacıyla bazı görüntüler, yatay eksene göre yansıtılmıştır.

5.1.2. Mimari ve Üst Parametreler

VAE ve β -VAE, aynı mimariyi paylaşmaktadır. İlgili mimariler Şekil 4.4'te ve tüm parametreleriyle detaylı mimari Şekil 5.1'te verilmiştir. Karşılaştırma amacıyla AAE, hem [55]'te önerilen mimari hem de katman ve filtre sayıları, ara katman boyutu değiştirilerek test edilmiş, en yüksek başarılı model raporlanmıştır. CAE'de ise VAE'nin aksine ara katmanda tek bir evrişimsel katman kullanılmaktadır. Görüntülerin ortalama değer ve standart sapmalarına yakınsayan kodlayıcı, filtre sayısı 3'ten 256'ya çıkan 5 adet evrişimsel katmandan ve 2 ayrı evrişimsel ara katmandan oluşmaktadır. Kodçözücü,

kodlayıcıya karşın devrik evrişimsel katmanlardan oluşmaktadır. Evrişimsel katmanların filtre boyutları 3×3 iken adım sayıları 1'dir. Evrişimsel katmanları, nitelik haritalarını (feature map) normalleştiren toplu normalleştirme (batch normalization) katmanları takip etmiştir [103]. Kaçak doğrultmaçlı aktivasyon birimi (Leaky rectified linear unit) (LReLU), kodçözücünün son katmanı hariç tüm katmanlara doğrusalsızlık (nonlinearity) sağlanması için kullanılmaktadır. Kodçözücünün son katmanında Tanh fonksiyonu kullanılmıştır. Nitelik haritalarının uzamsal boyutunu indirgemek için maksimum havuzlama (max pooling), yükseltmek için ise örnekleme hızlandırma katmanları kullanılmıştır. Tüm parametreleri ile birlikte VAE mimarisi Şekil 5.1'te verilmiştir. $\text{Conv}_{a,b,c}$ and $\text{Tconv}_{a,b,c}$, sırasıyla evrişimsel ve devrik evrişimsel katmanları temsil etmektedir. a , girdi görüntüsünün kanal sayısı, b , evrişim sonucu çıkan kanal sayısı, c ise evrişimsel filtrenin boyutudur. $\text{maks.havuzlama}_{2,2}$, filtre boyutu ve adımı temsil etmektedir. ör.hızlandırma_2 ise 2 kat yükseltme yapar. Gizli değişken vektörü z , girdi görüntülerini üretebilmek için $10 \times 8 \times 8$ 'lık boyuta getirilmiştir.

Modeller, $1e - 2$ başlangıç öğrenme oranı ile "A Method for Stochastic Optimization" (ADAM) [104] eniyileyicisi (optimizer) kullanılarak eğitilmiştir. Doğrulama kaybı azalması durdukça öğrenme oranı düşürülmektedir. Eğitim kümesini aşırı uyumlandırmayı engellemek amacıyla spesifik nitelikler üzerine yüksek ağırlık verilmesini engelleyen L_2 düzenleştirmesi kullanılmıştır. Eğitimde kullanılan toplu veri boyutu (batch size) 64'tür. Etkisi 5.1.3. bölümünde tartışılacak olan KL terimi ağırlığı β , verinin daha iyi bir çözülmüş temsilini elde edebilmek için birden büyük seçilmiştir. z için çok düşük boyutlu vektör, verinin marjinal olabilirliğini modelleyemezken çok büyük boyutlu vektör, anomali ve normal veri arasında ayrıştırıcı bir temsil öğrenemediğinden başarıyı düşürmektedir. Bu sebepten gizli değişken vektörü z boyutu, deneysel olarak 640 seçilmiştir. Doğrulama kümesinde belirlenen bu üst parametreler (hyperparameters) Tablo 5.1'de verilmiştir.



Şekil 5.1. İlk sütundaki bloklar, kodlayıcıyı temsil eder. Kodlayıcı, gizli değişken vektörü \mathbf{z} 'yi elde edebilmek için gerekli parametreleri, beş evrişimsel (conv), üç maksimum ortalama ve evrişimsel darboğaz katmanları ile elde eder. İkinci sütundaki bloklar ise kodçözücüü ifade eder. Kodçözücü, \mathbf{z} 'den girdi görüntüsünü geriçatmak için altı devrik evrişimsel (Tconv) ve üç örnekleme hızlandırma (upsampling) katmanı kullanır. bn ve LReLU, sırasıyla toplu normalleştirme (batch normalization) ve kaçak doğrultmaçlı aktivasyon biriminin kısaltmalarıdır [1].

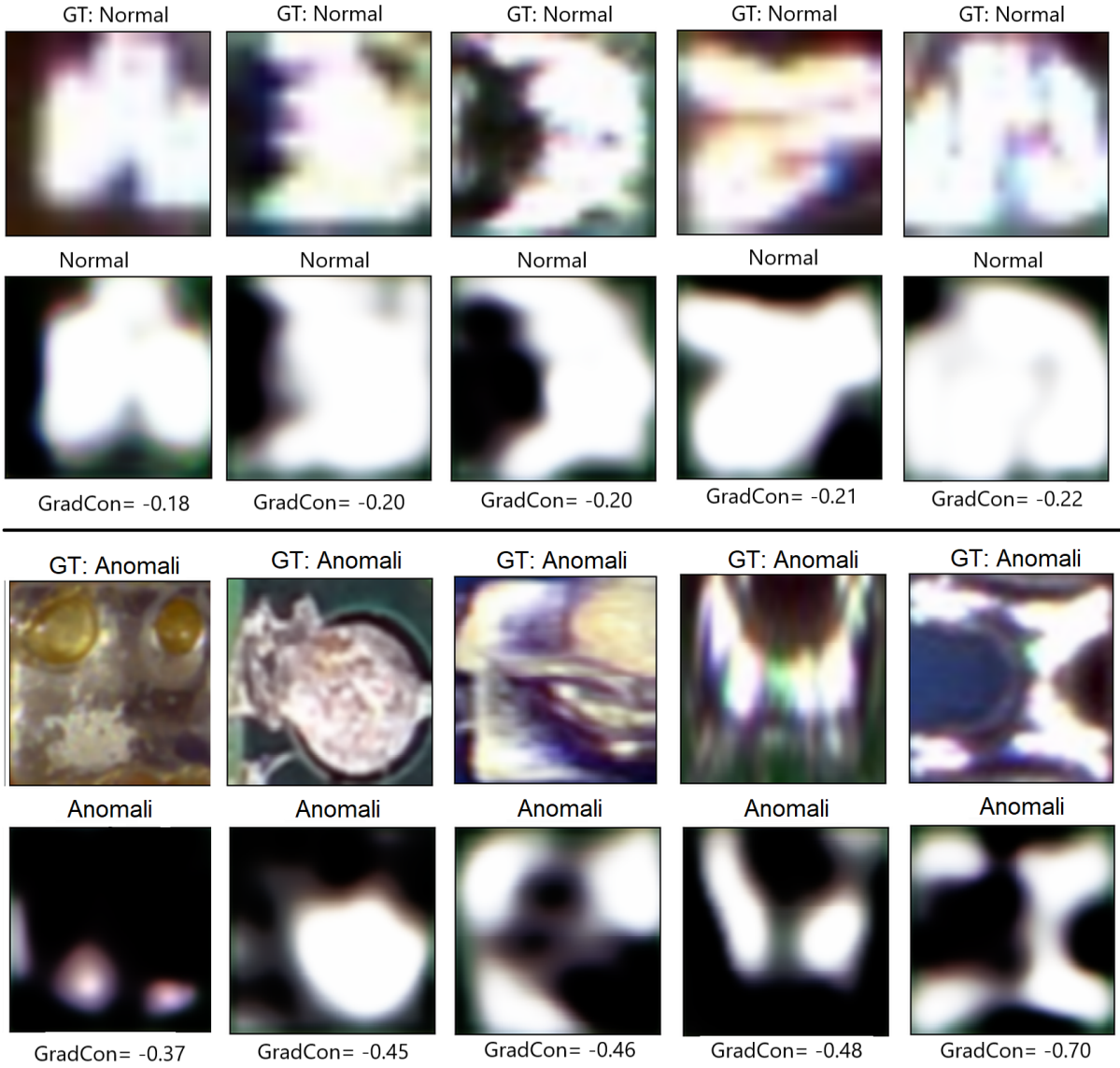
Tablo 5.1. β -VAE için üst parametreler.

Üst parametreler	Değerler
Başlangıçta öğrenme oranı	$1e - 2$
Kayıp değişmediğinde öğrenmeyi düşürme oranı	$\gamma = 0.1$
Epok sayısı	100
Eniyileyici (optimizer)	ADAM
Ağırlık düşürme oranı (weight decay)	$\lambda = 1e - 4$
Toplu veri boyutu (batch size) (N)	64
KL ağırlığı (β)	3
Gizli uzay boyutu	640

5.1.3. Deneysel Sonuçlar

Normal ve anomali girdi görüntüleri ile geri çatılan görüntüler Şekil 5.2’de verilmiştir. Normal örnekler düşük anomali puanı üretirken anomali örnekler, yüksek anomali puanı üretir. Kesinlik, duyarlılık, F1-puanı değerlendirme metriği olarak kullanılmıştır. Duyarlılık, anomali örneklerin ne kadarının doğru tespit edildiğini, kesinlik ise bu tahminlerin doğruluğunu göstermektedir. F1-puanı, bu metriklerden hesaplanarak modelin genel başarısını göstermektedir. Tüm sonuçlar 10 testin ortalamasıdır. Kısıtlı veri kümesinden dolayı 10 aşamalı çapraz-doğrulamaya başvurulmuştur. Bu sayede veri kümesindeki her örnek test edilmiştir. Her bir aşamada, veri kümesinin farklı bir kısmı doğrulama için ayrılmıştır. Geri kalanı ise model eğitimi için kullanılmıştır. Doğrulama kümesinde en düşük hatayı alan model son olarak test veri kümesinde test edilmiştir.

VAE ve β -VAE derin üretken modelleri geriçatım hatası (reconstruction loss), ELBO ve GradCon anomali puanı (geriçatım hatası ve gradyan kaybının bileşimi) metriklerine göre karşılaştırılmıştır. Bu karşılaştırma Tablo 5.2’de verilmiştir.



Şekil 5.2. Geri çatılmış (reconstructed) normal ve abnormal örnekler. 1. ve 3. satır girdi görüntüleri ve gerçek etiketleri gösterirken 2. ve 4. satır geri çatımlar ile birlikte tahmin edilen etiketler ve anomali puanlarını gösterir [1].

Tablo 5.2. Farklı anomali puanları için geriçatım bazlı ve üretken modellerin karşılaştırması (10 testin ortalaması) [1].

Yöntem / Değerlendirme	CAE		VAE			β -VAE			AAE [55]
	Recon	GradCon	Recon	ELBO	GradCon	Recon	ELBO	GradCon	T^2
Kesinlik	0.768 \pm 0.008	0.73 \pm 0.003	0.759 \pm 0.02	0.737 \pm 0.02	0.753 \pm 0.02	0.719 \pm 0.01	0.728 \pm 0.02	0.767 \pm 0.03	0.648 \pm 0.02
Duyarlılık	0.672 \pm 0.009	0.708	0.775 \pm 0.02	0.799 \pm 0.02	0.789 \pm 0.02	0.817 \pm 0.02	0.801 \pm 0.02	0.78 \pm 0.03	0.737 \pm 0.03
F1-puanı	0.717 \pm 0.007	0.719 \pm 0.002	0.766 \pm 0.02	0.766 \pm 0.01	0.770 \pm 0.01	0.764 \pm 0.01	0.762 \pm 0.01	0.774 \pm 0.02	0.689 \pm 0.02

CAE, geriçatım hatası ve GradCon ile değerlendirilmiştir. En yüksek başarı oranı CAE, VAE ve AAE'yi geride bırakarak β -VAE, $\beta = 3$ ve GradCon için 0.774 başarı oranı ile elde edilmiştir. ELBO kaybı, VAE ve β -VAE için geriçatım hatası ile aynı başarı oranına sahiptir. GradCon ise aktivasyon-bazlı temsil öğrenen derin üretken modeller için başarı oranını arttırmaktadır. AAE, anomali puanı olarak Mahalanobis mesafesi ile değerlendirilmiştir. AAE'nin çekişmeli öğrenme sayesinde VAE'ye göre daha keskin geriçatımlar yapmasına rağmen daha düşük başarı oranı elde ettiği gözlemlenmiştir.

Çözülmüş temsiller, geriçatım başarısı ile gizli kanal kapasitesi arasında denge sağlandığında ortaya çıkmaktadır [90]. Bu sebepten GradCon ile eğitilen β -VAE üzerindeki farklı β değerlerinin etkileri araştırılmış ve Tablo 5.3'te verilmiştir.

Tablo 5.3. β , üst parametrenin Gradyan kısıtı ile eğitilmiş β -VAE üzerindeki başarı etkisi [1].

Değerlendirme	Beta (β)				
	0.01	0.1	1 (VAE)	3	10
Kesinlik	0.705 \pm 0.01	0.709 \pm 0.01	0.775 \pm 0.01	0.767 \pm 0.03	0.647 \pm 0.04
Duyarlılık	0.712 \pm 0.01	0.752 \pm 0.02	0.752 \pm 0.02	0.78 \pm 0.03	0.686 \pm 0.05
F1-puanı	0.708 \pm 0.01	0.73 \pm 0.01	0.763 \pm 0.01	0.774 \pm 0.02	0.665 \pm 0.04

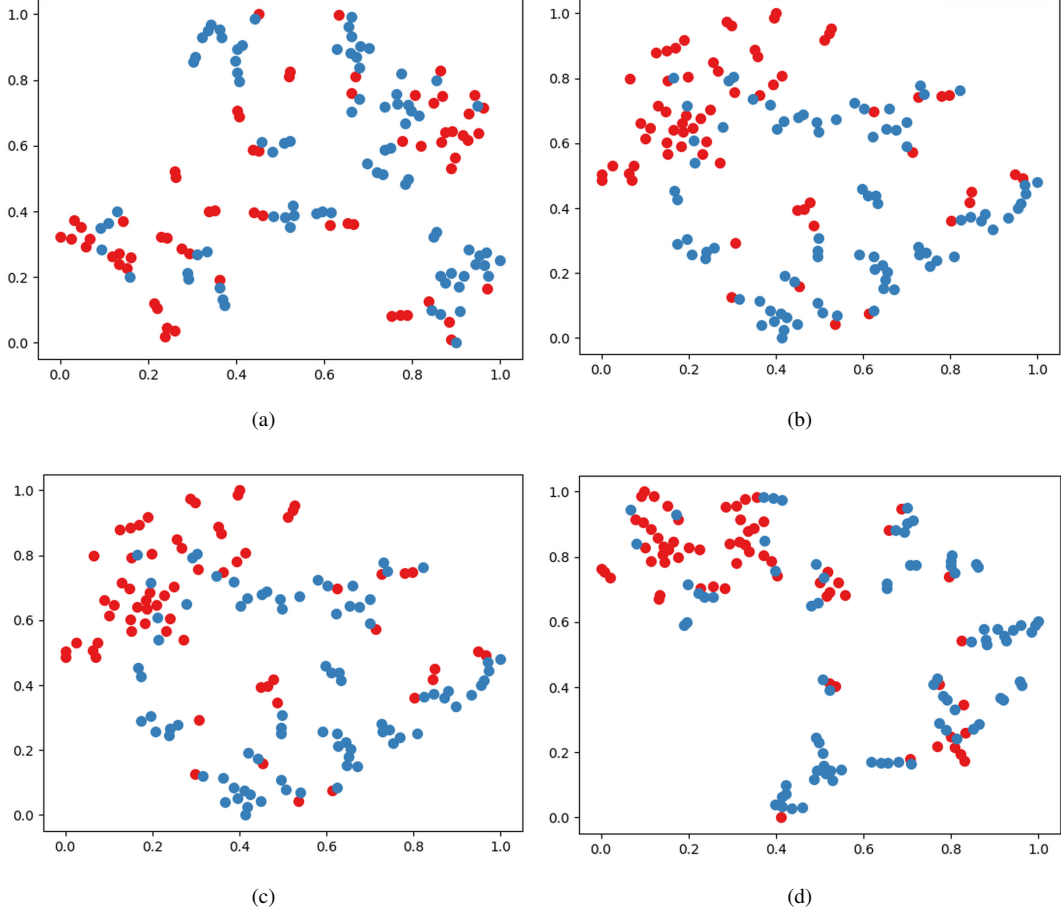
β -VAE için en yüksek başarı oranı $\beta = 3$ için elde edilmiştir. Daha büyük veya küçük beta değerleri, daha düşük başarı oranı verirken büyük beta değerlerinin daha kötü geriçatımlar yaptığı gözlemlenmiştir.

Farklı β değerlerinin gizli uzay temsilleri üzerindeki etkisini görmek amacıyla yüksek boyutlu veriyi görselleştirmede etkili lineer olmayan boyut indirgeme algoritması olan t-distributed Stochastic Neighbor Embedding (t-SNE) [84] kullanılmıştır. Farklı beta değerleri için GradCon ve ELBO görev fonksiyonları ile eğitilen β -VAE'ye test görüntüleri

		Tahmin çıktısı		Toplam
		Normal	Anomali	
Gerçek sınıf	Normal'	TP = 90	FN = 18	P' = 108
	Anomali'	FP = 11	TN = 54	N' = 65
Toplam		P=101	N=72	

Tablo 5.4. β -VAE modeli için hata matrisi.

verilmiştir. Çıktı olarak alınan yaklaşık sonsal dağılımın ortalama değer vektörüne, veri kümesi küçük olduğundan karışıklığı (perplexity) 5 seviyesindeki t-SNE indirgemesi uygulanmıştır. Daha sonra 640 boyutlu ortalama değerler vektörü 2 boyutlu uzaya indirgenmektedir. İndirgenen değerler, $[0, 1]$ değerlerine ölçeklendirilmiştir. GradCon ve ELBO görev fonksiyonları ile eğitilen beta-VAE Fig. 5.3 ve 5.4'te verilmiştir. t-SNE, 100 kez uygulanmış ve en düşük KL ıraksamasına sahip temsiller her bir model için seçilmiştir. Fig. 5.3 ve 5.4'te gösterildiği gibi aynı sınıftan olan örnekler birbirine yakın ve kümeler halindedir. Beta'nın 1 ve 10 değerleri için GradCon ile eğitilmiş beta-VAE'de sınıflar arası örnekler birbiri içine geçer, kolaylıkla ayırt edilemezler (Fig. 5.3 (b) ve (d)). Bu durum ELBO ile eğitilen beta-VAE için beta'nın 0.1 ve 10 değerleri için gözlemlenmiştir (Fig. 5.4 (a) ve (d)). Farklı parçalara bölünmüş olsalar iki model için de $\beta = 3$ için sınıflar daha ayrıştırılabilir (Fig. 5.3 ve 5.4 (c)). Gerçek karmaşık veriler için çözülmüş bir temsil elde etmek veya iki boyutta gözlemleyebilmek zordur ancak $\beta = 3$ için daha iyi bir çözülmüş temsil öğrenildiği sonucuna varabiliriz. Gradyan kısıtı (GradCon) ile eğitilen, $\beta = 3$ için β -VAE modelinin 173 örnekleli test kümesinde elde edilen hata matrisi Tablo 5.4'te verilmiştir. Gerçekte hatasız lehim (normal) olup normal olarak sınıflandırılan 90, anomali olarak sınıflandırılan 18 örnek vardır. Gerçekte hatalı lehim (anomali olup anomali olduğu saptanan örnek sayısı 54 iken 11 örnek normal lehim ile karıştırılmıştır.



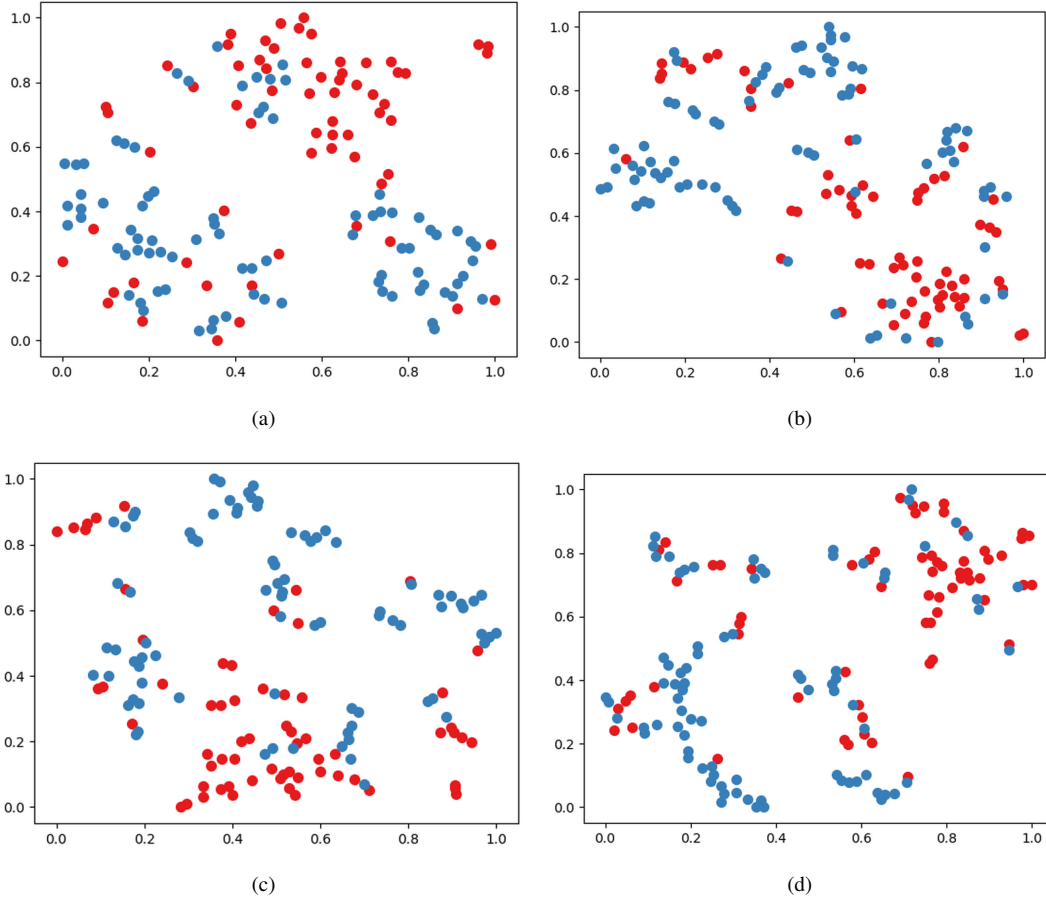
Şekil 5.3. (a) $\beta = 0.1$, (b) $\beta = 1$, (c) $\beta = 3$ ve (d) $\beta = 10$ için t-SNE uygulaması sonrası GradCon ile eğitilen β -VAE'in iki boyutlu gizli uzay temsili. Mavi ve kırmızı daireler sırasıyla normal ve anomali test verisini temsil etmektedir [1].

5.2. Maksimum Entropi Öğrenimi ile Deneysel Çalışmalar

Bu bölümde entropi-düzenleştirme bazlı öğrenme yöntemleri ile elde edilen deneysel sonuçlardan ve kullanılan veri kümesi, model parametrelerinden bahsedilmektedir.

5.2.1. Veri Kümesi ve Üst Parametreler

Gözetimli entropi-düzenleştirme yöntemleri için 3255 hatasız lehim, 745 hatalı lehime sahip bir veri kümesi kullanılmıştır. 659 normal ve 227 hatalı lehim test veri kümesi için ayrılmıştır. Kalan örnekler 5 aşamalı çapraz doğrulama için eğitim ve doğrulama kümelerine



Şekil 5.4. (a) $\beta = 0.1$, (b) $\beta = 1$, (c) $\beta = 3$ ve (d) $\beta = 10$ için t-SNE uygulaması sonrası ELBO ile eğitilen β -VAE'in iki boyutlu gizli uzay temsili. Mavi ve kırmızı daireler sırasıyla normal ve anomali test verisini temsil etmektedir [1].

ayrılmıştır. Tüm lehim görüntüleri 84×84 'e yeniden boyutlandırılıp 0 ortalama değer, 1 varyansa normalize edilmiştir.

İnce-taneli sınıflandırmada önerilen yöntemin gürbüzlüğü test etmek için VGG16 [105], GoogleNet [97], ResNet18 ve ResNet50 [106] gibi farklı model mimarilerinde deneyler yapılmıştır. Problemimiz ikili sınıflandırma olduğundan son katmanlarda aktivasyon fonksiyonu olarak Sigmoid kullanılmıştır. Tüm modeller, toplu (batch) boyutu 64 olan örnekler ile 2.500 döngü (iteration) ile eğitilmiştir. Momentum ile birlikte Ortalama Karesel Sapma ile İlerleme (RMSprop) algoritması, eniyileyici (optimizer) olarak kullanılmıştır. Başlangıçta eniyileyicinin öğrenim oranı $1e - 4$ olarak ayarlanmıştır. Doğrulama kümesinde

en düşük doğrulama kaybı veya en yüksek doğrulama başarımına sahip model, son olarak test kümesinde test edilip raporlanmıştır.

Hem maksimum entropi öğrenimi hem de çarpık JSD için entropi düzenleme miktarını belirleyen terim $\beta = 1$ seçilmiştir. Etiket düzeltmesi için literatürde sıklıkla kullanılan $\epsilon = 0.1$, odak kaybı (focal loss) için $\alpha_t = 0.25$ ve $\gamma = 2$ seçilmiştir. Veri kümesindeki normal ve hatalı veri dengesizliği sebebiyle değerlendirme metriği olarak F1-puanı kullanılmaktadır.

5.2.2. Deneysel Sonuçlar

Model eğitimleri, düzenleme olmadan, entropi-düzenleme bazlı yöntemler: etiket düzeltme, focal kayıp, maksimum entropi öğrenimi ve çarpık JSD ile entropi düzenleme, farklı mimariler kullanılarak karşılaştırılmıştır. Tablo 5.5'te verildiği gibi VGG16, ResNet18 ve GoogleNet için sırasıyla $\alpha = 0.1$, $\alpha = 0.9$ ve $\alpha = 0.75$ için en yüksek F1-puanı elde edilmiştir. Çarpık JSD, yalnızca ResNet50 modeli için maksimum entropi öğreniminin küçük bir farkla gerisinde kalmıştır. Düzenleme olmadan eğitilen model üzerine VGG16 modeli için başarı oranı yaklaşık %1.6, ResNet18 için %1.3 ve GoogleNet için %2.4 arttığı gözlemlenmiştir. Bu sonuçlar entropi-düzenleme bazlı yöntemlerin, özellikle de diğer yöntemleri geride bırakan çarpık JSD'nin lehim sınıflandırması için oldukça etkili olduğunu göstermektedir.

Çarpık JSD ile entropi düzenleme kullanılarak eğitilen ResNet50 modeli için test veri kümesinde elde edilen hata matrisi Tablo 5.6'da verilmiştir. Gerçekte hatasız lehim olan 657 örneğin 654'ü hatasız olarak doğru sınıflandırılırken 3 tanesine model, anomali kararı vermektedir. 226 anomali örneğinin ise model tarafından 214'üne anomali kararı verilirken 12 tanesi normal lehim ile karıştırılmıştır. Gözetimsiz derin üretken modeller ile elde edilen Tablo 5.4'teki hata matrisi ile karşılaştırıldığında gözetimli öğrenimin gözetimsiz göze göre çok daha başarılı olduğu söylenebilir. Ancak eğitimli yaklaşımların, eğitimde yüksek sayıda hatasız lehimde gerektirdiği göz önünde bulundurulmalıdır.

Regularization	VGG16	ResNet18	ResNet50	GoogleNet	
No regularization	96.599	96.035	93.488	95.861	
Focal loss [107]	95.128	95.692	96.128	97.321	
Label smoothing [97] ($\alpha \rightarrow 1$)	97.285	96.018	95.946	96.916	
Maximum entropy [65] ($\alpha \rightarrow 0$)	97.297	97.065	97.039	95.946	
Skewed JSD	$\alpha = 0.1$	98.206	95.175	95.946	97.788
	$\alpha = 0.5$	97.321	95.711	95.281	96.583
	$\alpha = 0.75$	97.297	96.380	96.092	98.214
	$\alpha = 0.9$	97.065	97.321	96.889	97.987

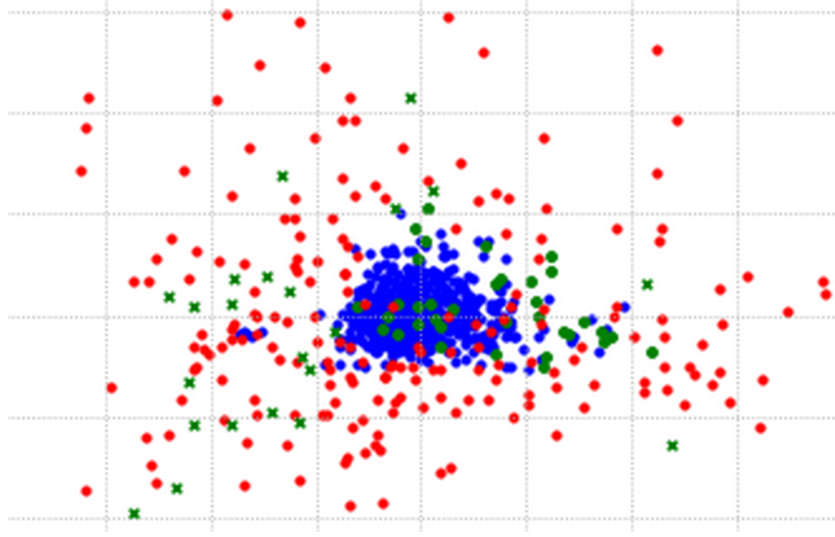
Tablo 5.5. Farklı model mimarileri ve entropi-düzenleştirme yöntemlerinin lehim test kümesi üzerindeki F1-puanı (%). Her model için en yüksek başarı kalın yazılar ile verilir. Tüm sonuçlar 10 testin bir ortalamasıdır.

		Tahmin çıktısı		
		normal	Anomali	Toplam
Gerçek sınıf	normal'	TP = 654	FN = 3	P' = 657
	Anomali'	FP = 12	TN = 214	N' = 226
Toplam		P=666	N=217	

Tablo 5.6. Çarpık JSD ile eğitilmiş ResNet50 modeli için hata matrisi.

Entropi-düzenleştirmenin, düzenleştirme olmadan eğitilen modele göre etkisi modelin test kümesi üzerindeki sınıflandırmasında Şekil 5.5'te verildiği gibi gösterilmiştir. Yeşil noktalar, düzenleştirmesiz eğitim sonucu hatalı sınıflandırılan ancak entropi-düzenleştirmesi için çarpık JSD kullanımı sonucu doğru sınıflandırılan normal ve hatalı lehimleri temsil etmektedir. Normal örnekler ve saçılmış hatalı örnekler arasında birçok hatalı sınıflandırılan örnek vardır. Bu noktalar yeşil ile gösterilmiştir.

Entropi-düzenleştirmesi ve görüntülerde model tahminlerinin etkisini yorumlamak için, Grad-CAM ile herbir test görüntüsünün aktivasyon haritaları görselleştirilebilir. Test

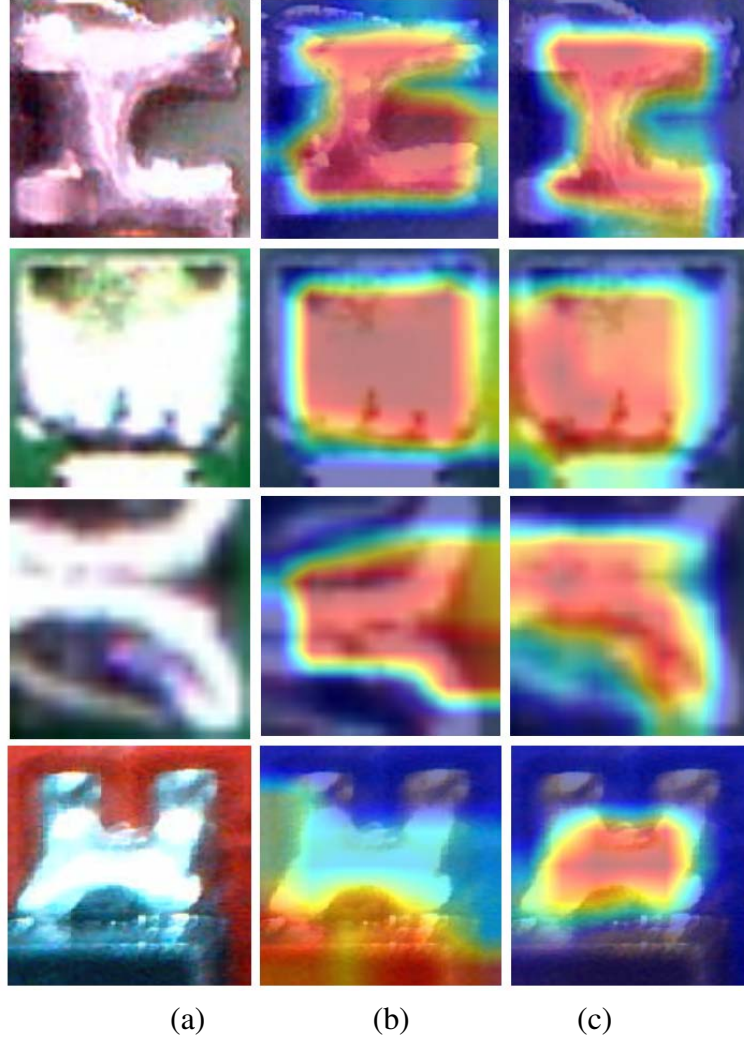


Şekil 5.5. ResNet50 modeli ortalama havuzlama (average pooling) katmanı test verisi çıktı niteliklerinin temel bileşenleri (principle components). Normal ve hatalı lehimler sırasıyla mavi ve kırmızı daireler ile gösterilmektedir. Yeşil daireler, düzenleştirmesiz eğitim ile sınıflandırma sonucu yanlış sınıflandırılırken (yanlış negatifler) çarpık JSD ile doğru sınıflandırılan (doğru pozitifler) hatasız lehimleri göstermektedir. Yeşil çarpılar ise çarpık JSD ile doğru sınıflandırılırken (doğru negatifler) düzenleştirme olmadan yanlış sınıflandırılan (yanlış pozitifler) hatalı lehim örneklerini temsil etmektedir.

görüntüleri, düzenleştirme olmadan ve çarpık JSD ile eğitilen VGG16 modelinin aktivasyon haritaları sırasıyla Şekil 5.6 (a), (b), (c)'de verilmiştir. Şekil 5.6'te gösterildiği gibi çarpık JSD ile düzenleştirelmüş model, düzenleştirelmemiş (çapraz-entropi ile eğitim) modele göre sınıflandırmada daha ayırt edici bölgelere odaklanmaktadır. Ek olarak düzenleştirelmüş model, BDK arkaplanı, lehim pedi hariç bölgeler gibi gürültülerden daha az etkilendiği görülmektedir.

6. SONUÇ

Bu tez kapsamında gözetimsiz derin üretken modeller ile kısıtlanmamış bir bağlamda denetim yöntemi önerilmiştir. Önerilen derin üretken model ile nitelik mühendisliği, referans hatasız lehim/BDK, özel bir aydınlatmaya ihtiyaç duymadan lehim hatalarının tespiti yapılabildiği gösterilmektedir. Literatürde kullanılan farklı anomali puanlama teknikleri



Şekil 5.6. Grad-CAM yaklaşımı sınıf ayırımında belirleyici bölgeleri gösterir. RGB test görüntüleri (a). Düzenlenileştirmesiz (b) ve çarpık JSD (c) ile eğitilen VGG16 modelinin aktivasyon haritaları. Isı haritası, sınıflandırma için modelin odaklandığı kısmı gösterir. Yoğunluk, maviden kırmızıya doğru gittikçe artar. En çok çarpık JSD ile düzenlenileştirilmiş modelin CAM sonuçları lehimlenmiş bölgeler ile örtüşür.

karşılaştırılmıştır. Çözülmüş temsil öğrenen yaklaşımlar ile hatalı ve hatasız lehimler için daha ayrıştırılabilir bir temsil öğrenilebildiği gösterilmiştir.

Hatalı ve hatasız lehimlerin birbirine olan benzerliklerinden dolayı lehim denetimi için önerilen bir diğer yaklaşım ince taneli sınıflandırmadır. Model entropisini dolayısıyla da kararsızlığını arttırarak daha genelleşebilir bir model sayesinde lehim sınıflandırma başarımı arttırılmaktadır. Literatürde mevcut entropi maksimizasyon yöntemlerine göre çarpık JSD

fonksiyonu kullanarak entropiyi maksimize etmenin lehim denetimi probleminde farklı mimarilerde başarıyı arttırdığı gözlemlenmiştir. Çarpık JSD ile entropi düzenlenmesi sayesinde modelin, sınıflar için daha ayrıştırıcı bölgelere odaklanabildiği gösterilmektedir. Hatalı ve hatasız lehim sayısının yeterli olduğu durumlarda ise gözetimli evrimsel ağlar önerilmektedir. Ancak gözetimli modeller gözetimsiz modellerin aksine eğitim sırasında çok sayıda hatalı lehim görüntülerine (anomali örnekler) ihtiyaç duymaktadır. Lehim denetimi problemi gibi hatalı lehim elde etmenin zor olduğu ve nadir rastlandığı bağlamlarda bu durum önemli bir ikilemdir.

YME lehimleri birbiri ile görsel olarak benzerlik gösterse de YME gövdeleri özellikle renk ve şekil açısından büyük farklılıklar gösterebilir. Bu sebepten bu tez kapsamında BDK üzerinde yalnızca YME lehimlerinin teker teker gözetilerek denetlenmesi amaçlanmıştır. Elektronik kart üretim hatları yüksek doğruluk ile üretim gerektirdiğinden kartların kapsamlı denetimi şarttır. Ancak dikey malzeme ve kaymış devre elemanı gibi hataların tespitinde yalnızca lehimler üzerinden denetim yeterli değildir. YME gövdesi ve lehimleri birlikte gözetilerek denetlenmelidir. Bu sebepten devre elemanı gövdesi ve lehimlerini içeren kapsamlı bir veri kümesi üzerinde yaklaşım geliştirilmesi daha doğru olacaktır.

KAYNAKLAR

- [1] Furkan Ulger, Seniha E. Yuksel, and Atila Yilmaz. Anomaly detection for solder joints using β -VAE. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 11(12):2214–2221, **2021**.
- [2] Mihály Janóczki, Ákos Becker, László Jakab, Richárd Gróf, and Tibor Takács. Automatic optical inspection of soldering. *Materials Science-Advanced Topics*, **2013**.
- [3] Siemens Software. PCB 101 academy - learn how printed circuit boards are assembled. <https://www.youtube.com/watch?v=usmDfGM2KmU>, **2020**. Accessed: 2022-02-05.
- [4] Furkan Ulger and Seniha E. Yuksel. A standalone open-source system for optical inspection of printed circuit boards. In *2019 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, pages 105–110. IEEE, **2019**.
- [5] Fupei Wu and Xianmin Zhang. Feature-extraction-based inspection algorithm for IC solder joints. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 1(5):689–694, **2011**.
- [6] Xie Hongwei, Zhang Xianmin, Kuang Yongcong, and Ouyang Gaofei. Solder joint inspection method for chip component using improved Adaboost and Decision tree. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 1(12):2018–2027, **2011**.
- [7] Fupei Wu and Xianmin Zhang. An inspection and classification method for chip solder joints using color grads and boolean rules. *Robotics and Computer-Integrated Manufacturing*, 30(5):517–526, **2014**.

- [8] Wu Hao, Zhang Xianmin, Kuang Yongcong, Ouyang Gaofei, and Xie Hongwei. Solder joint inspection based on neural network combined with genetic algorithm. *Optik*, 124(20):4110–4116, **2013**.
- [9] Andriy Mnih. Modern latent variable models and variational inference. <https://www.youtube.com/watch?v=7Pcvdo4EJeo>, **2020**. Accessed: 2021-03-10.
- [10] Madhav Moganti, Fikret Ercal, Cihan H. Dagli, and Shou Tsunekawa. Automatic PCB inspection algorithms: A survey. *Computer Vision and Image Understanding*, 63(2):287–313, **1996**.
- [11] G. A. W. West. A system for the automatic visual inspection of bare-printed circuit boards. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-14(5):767–773, **1984**. doi:10.1109/TSMC.1984.6313300.
- [12] Wen-Yen Wu, Mao-Jiun J. Wang, and Chih-Ming Liu. Automated inspection of printed circuit boards through machine vision. *Computers in Industry*, 28(2):103–111, **1996**.
- [13] Qian Ye, Nian Cai, Jiaming Li, Feiyang Li, Han Wang, and Xindu Chen. IC solder joint inspection based on an adaptive-template method. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 8(6):1121–1127, **2018**.
- [14] Shih-Chieh Lin, Chih-Hsien Chou, and Chia-Hsin Su. A development of visual inspection system for surface mounted devices on printed circuit board. In *IECON 2007-33rd Annual Conference of the IEEE Industrial Electronics Society*, pages 2440–2445. IEEE, **2007**.
- [15] Kenneth Sundaraj. PCB inspection for missing or misaligned components using background subtraction. *WSEAS Transactions on Information Science and Applications*, 6(5):778–787, **2009**.

- [16] Du-Ming Tsai and Chih-Kai Huang. Defect detection in electronic surfaces using template-based Fourier image reconstruction. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 9(1):163–172, **2018**.
- [17] Roland T. Chin and Charles A. Harlow. Automated visual inspection: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-4(6):557–573, **1982**.
- [18] John F. Jarvis. A method for automating the visual inspection of printed wiring boards. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-2(1):77–82, **1980**.
- [19] Yasuhiko Hara, Nobuyuki Akiyama, and Koichi Karasaki. Automatic inspection system for printed circuit boards. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-5(6):623–630, **1983**.
- [20] Yasuhiko Hara, Hideaki Doi, Koichi Karasaki, and Tadashi Iida. A system for PCB automated inspection using fluorescent light. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(1):69–78, **1988**.
- [21] Alan Crispin and V. Rankov. Automated inspection of PCB components using a genetic algorithm template-matching approach. *The International Journal of Advanced Manufacturing Technology*, 35(3):293–300, **2007**.
- [22] Giuseppe Acciani, Gioacchino Brunetti, and Girolamo Fornarelli. Application of neural networks in optical inspection and classification of solder joints in surface mount technology. *IEEE Transactions on Industrial Informatics*, 2(3):200–209, **2006**.
- [23] Eam K. Teoh, D. P. Mital, B. W. Lee, and Louis Wee. Automated visual inspection of surface mount PCBs. In *[Proceedings] IECON'90: 16th Annual Conference of IEEE Industrial Electronics Society*, pages 576–580. IEEE, **1990**.

- [24] Mahmoud Annaby, Yasser Fouda, and Muhammad A. Rushdi. Improved normalized cross-correlation for defect detection in printed-circuit boards. *IEEE Transactions on Semiconductor Manufacturing*, 32(2):199–211, **2019**.
- [25] Hongwei Xie, Yongcong Kuang, and Xianmin Zhang. A high speed AOI algorithm for chip component based on image difference. In *2009 International Conference on Information and Automation*, pages 969–974. IEEE, **2009**.
- [26] Nian Cai, Jianfa Lin, Qian Ye, Han Wang, Shaowei Weng, and Bingo Wing-Kuen Ling. A new IC solder joint inspection method for an automatic optical inspection system based on an improved visual background extraction algorithm. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 6(1):161–172, **2015**.
- [27] Nian Cai, Yang Zhou, Qian Ye, Gen Liu, Han Wang, and Xindu Chen. IC solder joint inspection via robust principle component analysis. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 7(2):300–309, **2017**.
- [28] Jun Jiang, Jun Cheng, and Dacheng Tao. Color biological features-based solder paste defects detection and classification on printed circuit boards. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 2(9):1536–1544, **2012**.
- [29] Nang S. S. Mar, Prasad K.D.V. Yarlagadda, and Clinton Fookes. Design and development of automatic visual inspection system for PCB manufacturing. *Robotics and Computer-integrated Manufacturing*, 27(5):949–962, **2011**.
- [30] Richard O. Duda and Peter E. Hart. Use of the Hough transformation to detect lines and curves in pictures. *Commun. ACM*, 15(1):11–15, **1972**. ISSN 0001-0782. doi:10.1145/361237.361242.
- [31] Zhou Zeng, Lizhuang Ma, and Masaki Suwa. Algorithm of locating PCB components based on colour distribution of solder joints. *The International Journal of Advanced Manufacturing Technology*, 53(5):601–614, **2011**.

- [32] Zhou Zeng, Li Zhuang Ma, and Zuo Yong Zheng. Automated extraction of PCB components based on specularly using layered illumination. *Journal of Intelligent Manufacturing*, 22(6):919–932, **2011**.
- [33] Hao Wu, Xianmin Zhang, Hongwei Xie, Yongcong Kuang, and Gaofei Ouyang. Classification of solder joint using feature selection based on Bayes and support vector machine. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 3(3):516–522, **2013**.
- [34] Ji-Deok Song, Young-Gyu Kim, and Tae-Hyoung Park. SMT defect classification by feature extraction region optimization and machine learning. *The International Journal of Advanced Manufacturing Technology*, 101(5):1303–1313, **2019**.
- [35] Ong T. Yaow, Zahurin Samad, and Mani M. Ratnam. Solder joint inspection with multi-angle imaging and an artificial neural network. *The International Journal of Advanced Manufacturing Technology*, 38(5-6):455–462, **2008**.
- [36] David W. Capson and S-K Eng. A tiered-color illumination approach for machine inspection of solder joints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(3):387–393, **1988**.
- [37] Tae-Hyeon Kim, Tai-Hoon Cho, Young Shik Moon, and Sung Han Park. Visual inspection system for the classification of solder joints. *Pattern Recognition*, 32(4):565–575, **1999**.
- [38] Shigeki Kobayashi, Yasuaki Tanimura, and Teruhisa Yotsuya. Apparatus for inspecting printed circuit boards and the like, and method of operating same, **1993**. US Patent 5,245,671.
- [39] Fupei Wu, Xianmin Zhang, Yongcong Kuan, and Zhenzhen He. An AOI algorithm for PCB based on feature extraction. In *2008 7th World Congress on Intelligent Control and Automation*, pages 240–247. IEEE, **2008**.

- [40] Giuseppe Acciani, Giocchino Brunetti, and Girolamo Fornarelli. Automatic detection of solder joint defects on integrated circuits. In *2007 IEEE International Symposium on Circuits and Systems*, pages 1021–1024. IEEE, **2007**.
- [41] Sandra L. Bartlett, Paul J. Besl, Charles L. Cole, Ramesh Jain, Debashish Mukherjee, and Kurt D. Skifstad. Automatic solder joint inspection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(1):31–43, **1988**.
- [42] Shao-Nung Chiu and Ming-Hwei Perng. Reflection-area-based feature descriptor for solder joint inspection. *Machine Vision and Applications*, 18(2):95–106, **2007**.
- [43] Yukichi Koji, Naoyoshi Takatsu, and Masanari Oh. Visual solder inspection using neural network. *Systems and Computers in Japan*, 27(1):92–100, **1996**.
- [44] C.L.S.C. Fonseka and J.A.K.S. Jayasinghe. Implementation of an automatic optical inspection system for solder quality classification of THT solder joints. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 9(2):353–366, **2018**.
- [45] T.S. Yun, K.J. Sim, and H.J. Kim. Support vector machine-based inspection of solder joints using circular illumination. *Electronics Letters*, 36(11):949–951, **2000**.
- [46] Kuk W. Ko and Hyung S. Cho. Solder joints inspection using a neural network and fuzzy rule-based classification method. *IEEE Transactions on Electronics Packaging Manufacturing*, 23(2):93–103, **2000**.
- [47] Jae-Hoon Kim and Hyung S. Cho. Neural network-based inspection of solder joints using a circular illumination. *Image and Vision Computing*, 13(6):479–490, **1995**.
- [48] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet classification with deep convolutional neural networks. *NIPS*, 25, **2012**.

- [49] Nian Cai, Guandong Cen, Jixiu Wu, Feiyang Li, Han Wang, and Xindu Chen. SMT solder joint inspection via a novel cascaded convolutional neural network. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 8(4):670–677, **2018**.
- [50] Wenting Dai, Abdul Mujeeb, Marius Erdt, and Alexei Sourin. Soldering defect detection in automatic optical inspection. *Advanced Engineering Informatics*, 43:101004, **2020**.
- [51] Hao Wu, Wenbin Gao, and Xiangrong Xu. Solder joint recognition using Mask R-CNN method. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 10(3):525–530, **2019**.
- [52] Yu-Ting Li, Paul Kuo, and Jiun-In Guo. Automatic industry PCB board DIP process defect detection system based on deep ensemble self-adaption method. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 11(2):312–323, **2020**.
- [53] Abdul Mujeeb, Wenting Dai, Marius Erdt, and Alexei Sourin. One class based feature learning approach for defect detection using deep autoencoders. *Advanced Engineering Informatics*, 42:100933, **2019**.
- [54] Zimu Zheng, Jie Pu, Linghui Liu, Dan Wang, Xiangming Mei, Sen Zhang, and Quanyu Dai. Contextual anomaly detection in solder paste inspection with multi-task learning. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(6):1–17, **2020**.
- [55] Keisuke Goto, Kunihito Kato, Takaho Saito, and Hiroaki Aizawa. Adversarial autoencoder for detecting anomalies in soldered joints on printed circuit boards. *Journal of Electronic Imaging*, 29(4):041013, **2020**.
- [56] Jiaming Li, Nian Cai, Zhuokun Mo, Guang Zhou, and Han Wang. IC solder joint inspection via generator-adversarial-network based template. *Machine Vision and Applications*, 32(4):1–16, **2021**.

- [57] Eva Jabbar, Philippe Besse, Jean-Michel Loubes, and Christophe Merle. Conditional anomaly detection for quality and productivity improvement of electronics manufacturing systems. In *International Conference on Machine Learning, Optimization, and Data Science*, pages 711–724. Springer, **2019**.
- [58] J. H. Kim, H. S. Cho, and S. Kim. Pattern classification of solder joint images using a correlation neural network. *Engineering Applications of Artificial Intelligence*, 9(6):655–669, **1996**.
- [59] Horng-Hai Loh and Ming-Sing Lu. Printed circuit board inspection using image analysis. *IEEE Transactions on Industry Applications*, 35(2):426–432, **1999**.
- [60] Nian Cai, Qian Ye, Gen Liu, Han Wang, and Zhijing Yang. IC solder joint inspection based on the Gaussian Mixture Model. *Soldering & Surface Mount Technology*, **2016**.
- [61] Nang S. S. Mar. *Vision-based Classification of Solder Joints*. Master’s thesis, Queensland University of Technology, Brisbane, Australia, **2010**.
- [62] Xiu-Shen Wei, Yi-Zhe Song, Oisin Mac Aodha, Jianxin Wu, Yuxin Peng, Jinhui Tang, Jian Yang, and Serge Belongie. Fine-grained image analysis with deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2021**.
- [63] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the Inception architecture for computer vision. In *CVPR*, pages 2818–2826. **2016**.
- [64] Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*, **2017**.
- [65] Abhimanyu Dubey, Otkrist Gupta, Ramesh Raskar, and Nikhil Naik. Maximum-entropy fine grained classification. *NIPS*, 31, **2018**.

- [66] Jishnu Mukhoti, Viveka Kulharia, Amartya Sanyal, Stuart Golodetz, Philip Torr, and Puneet Dokania. Calibrating deep neural networks using Focal loss. *NIPS*, 33:15288–15299, **2020**.
- [67] Clara Meister, Elizabeth Salesky, and Ryan Cotterell. Generalized entropy regularization or: There’s nothing special about label smoothing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6870–6886. Association for Computational Linguistics, Online, **2020**. doi:10.18653/v1/2020.acl-main.615.
- [68] Christopher Pramerdorfer and Martin Kampel. A dataset for computer-vision-based PCB analysis. In *2015 14th IAPR International Conference on Machine Vision Applications (MVA)*, pages 378–381. IEEE, **2015**.
- [69] Md A. Reza, Zhenhua Chen, and David J. Crandall. Deep neural network–based detection and verification of microelectronic images. *Journal of Hardware and Systems Security*, 4(1):44–54, **2020**.
- [70] Chia-Wen Kuo, Jacob D. Ashmore, David Huggins, and Zsolt Kira. Data-efficient graph embedding learning for PCB component detection. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 551–560. IEEE, **2019**.
- [71] Hangwei Lu, Dhvani Mehta, Olivia P. Paradis, Navid Asadizanjani, Mark Tehranipoor, and Damon L Woodard. FICS-PCB: A multi-modal image dataset for automated printed circuit board visual inspection. *IACR Cryptol. ePrint Arch.*, 2020:366, **2020**.
- [72] Jiaquan Shen, Ningzhong Liu, and Han Sun. Defect detection of printed circuit board based on lightweight deep convolution network. *IET Image Processing*, 14(15):3932–3940, **2020**.

- [73] Nathan Jessurun, Olivia P. Dizon-Paradis, Jacob Harrison, Shajib Ghosh, Mark M. Tehranipoor, Damon L. Woodard, and Navid Asadizanjani. FPIC: A novel semantic dataset for optical PCB assurance. *arXiv preprint arXiv:2202.08414*, **2022**.
- [74] Yehonatan Fridman, Matan Rusanovsky, and Gal Oren. ChangeChip: A reference-based unsupervised change detection for PCB defect detection. *arXiv preprint arXiv:2109.05746*, **2021**.
- [75] Weibo Huang, Peng Wei, Manhua Zhang, and Hong Liu. HRIPCB: a challenging dataset for PCB defects detection and classification. *The Journal of Engineering*, 2020(13):303–309, **2020**.
- [76] Sanli Tang, Fan He, Xiaolin Huang, and Jie Yang. Online PCB defect detector on a new PCB defect dataset. *arXiv preprint arXiv:1902.06197*, **2019**.
- [77] Diederik P. Kingma and Max Welling. An introduction to Variational Autoencoders. *arXiv preprint arXiv:1906.02691*, **2019**.
- [78] Carl Doersch. Tutorial on Variational Autoencoders. *arXiv preprint arXiv:1606.05908*, **2016**.
- [79] Christopher Bishop. *Pattern Recognition and Machine Learning*. Springer, **2006**.
- [80] David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, **2017**.
- [81] Shakir Mohamed. Bayesian inference. <https://www.youtube.com/watch?v=DIItJbz2OH5U>, **2020**. Accessed: 2021-02-16.
- [82] Nicola Pezzotti. *Dimensionality-Reduction Algorithms for Progressive Visual Analytics*. Ph.D. thesis, **2019**.

- [83] Ian T. Jolliffe. *Principle Component Analysis*. Springer, **2002**.
- [84] Laurens v. d. Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(11), **2008**.
- [85] Martin Wattenberg, Fernanda Viégas, and Ian Johnson. How to use t-SNE effectively. *Distill*, **2016**. doi:10.23915/distill.00002.
- [86] Kentaro Wada. Labelme: Image Polygonal Annotation with Python. doi:10.5281/zenodo.5711226.
- [87] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong J. Lee. YOLACT: Real-time instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9157–9166. **2019**.
- [88] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2117–2125. **2017**.
- [89] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440. **2015**.
- [90] Irina Higgins, Loïc Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *ICLR*. **2017**.
- [91] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, **2013**.

- [92] Yoshua Bengio. Deep learning of representations: Looking forward. In *International Conference on Statistical Language and Speech Processing*, pages 1–37. Springer, **2013**.
- [93] Christopher P. Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in β -vae. *arXiv preprint arXiv:1804.03599*, **2018**.
- [94] Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*, **2013**.
- [95] Gukyeong Kwon, Mohit Prabhushankar, Dogancan Temel, and Ghassan AlRegib. Backpropagated gradient representations for anomaly detection. In *ECCV*, pages 206–226. Springer, **2020**.
- [96] Giacomo Boracchi and Diego Carrera. Tutorial: Anomaly detection in images. *IEEE ICIP*, **2020**.
- [97] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9. **2015**.
- [98] Edwin T. Jaynes. Information theory and statistical mechanics. ii. *Physical Review*, 108(2):171, **1957**.
- [99] Google Research. GPU erişim hizmeti. <https://colab.research.google.com/>, **2022**. Accessed: 2021-01-10.
- [100] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. **2017**.
- [101] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn:

- Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, **2011**.
- [102] Geoffrey E. Hinton and Ruslan R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, **2006**.
- [103] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456. PMLR, **2015**.
- [104] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, **2014**.
- [105] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, **2014**.
- [106] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778. **2016**.
- [107] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pages 2980–2988. **2017**.
- [108] Frank Nielsen. On a generalization of the Jensen–Shannon divergence and the Jensen–Shannon centroid. *Entropy*, 22(2):221, **2020**.
- [109] Frank Nielsen and Sylvain Boltz. The burbea-rao and bhattacharyya centroids. *IEEE Transactions on Information Theory*, 57(8):5455–5466, **2011**.

EKLER

EK 1 – $D_{KL}(q_\phi(\mathbf{z}|\mathbf{X})||p_\theta(\mathbf{z}))$ 'ın Çözümü

ELBO görev fonksiyonundaki KL ıraksaması ifadesi Gaussian olasılıksal dağılımı varsayımına göre matematiksel olarak hesaplanabilir. Gizli değişkenler \mathbf{z} üzerindeki önsel dağılımın $p_\theta(\mathbf{z}) = N(0, \mathbf{I})$ ve sonsal dağılımın $q_\phi(\mathbf{z}|\mathbf{X})$ Gaussian olduğu durumda ise bu ifade, Eşitlik 19'deki gibi hesaplanır [94].

$$\begin{aligned} D_{KL}(q_\phi(\mathbf{z}|\mathbf{X})||p_\theta(\mathbf{z})) &= \int_{\mathbf{z}} q_\phi(\mathbf{z}|\mathbf{X}) \log \frac{q_\phi(\mathbf{z}|\mathbf{X})}{p_\theta(\mathbf{z})} d\mathbf{z} \\ &= \int_{\mathbf{z}} \mathcal{N}(\mu_{\mathbf{z}|\mathbf{X}}, \sigma_{\mathbf{z}|\mathbf{X}}) \log \frac{\mathcal{N}(\mu_{\mathbf{z}|\mathbf{X}}, \sigma_{\mathbf{z}|\mathbf{X}})}{\mathcal{N}(0, \mathbf{I})} d\mathbf{z} \\ &= \frac{1}{2} \left(\sum_{j=1}^k (1 + (\log \sigma_{\mathbf{z}|\mathbf{X}}^2)_j - (\mu_{\mathbf{z}|\mathbf{X}})_j^2 - (\sigma_{\mathbf{z}|\mathbf{X}}^2)_j^2) \right) \end{aligned} \quad (19)$$

EK 2 – KL ıraksaması ve Çarpık JSD

Jensen-Shannon ıraksaması (JSD), KL ıraksamasının simetrik ve düzleştirilmiş bir versiyonudur. JSD, KL ıraksaması cinsinden aşağıda verildiği gibi ifade edilebilir.

$$J(p||q) = \frac{1}{2} [D_{KL}(p||\frac{p+q}{2}) + D_{KL}(q||\frac{p+q}{2})]$$

Çarpık JSD, p ve q olasılıksal dağılımların ağırlıklandırılmış çarpık KL ıraksamalarının toplamları Eşitlik 20'de verilmiştir [108]. $\alpha \in [0, 1]$, ağırlıklandırmayı kontrol eden terimidir.

$$J_{\alpha}(p||q) = (1 - \alpha)D_{KL}(p||(pq)_{\alpha}) + \alpha D_{KL}(q||(pq)_{\alpha}), \quad (20)$$

$$\begin{aligned} (pq)_{\alpha} &= (1 - \alpha)p + \alpha q \\ &= -(1 - \alpha) \sum p \log\left(\frac{(pq)_{\alpha}}{p}\right) - \alpha \sum q \log\left(\frac{(pq)_{\alpha}}{q}\right) \\ &= -(1 - \alpha) \left(\sum p \log(pq)_{\alpha} - \sum p \log p\right) - \alpha \left(\sum q \log(pq)_{\alpha} - \sum q \log q\right) \\ &= -(1 - \alpha) \sum p \log(pq)_{\alpha} - (1 - \alpha)H(p) - \alpha \sum q \log(pq)_{\alpha} - \alpha H(q) \\ &= - \sum (1 - \alpha)p \log((1 - \alpha)p + \alpha q) \\ &\quad - (1 - \alpha)H(p) - \alpha \sum q \log((1 - \alpha)p + \alpha q) - \alpha H(q) \\ &= - \sum ((1 - \alpha)p + \alpha q) \log((1 - \alpha)p + \alpha q) - (1 - \alpha)H(p) - \alpha H(q) \\ &= H((1 - \alpha)p + \alpha q) - (1 - \alpha)H(p) - \alpha H(q) \end{aligned}$$

Çarpık JSD ifadesi, $\frac{1}{\alpha(1-\alpha)}$ terimi ile ölçeklendirilerek sürekli çarpık JSD'ler elde edilmiştir:

$$J_{\alpha}(p||q) = \frac{1}{\alpha(1-\alpha)} [H((1 - \alpha)p + \alpha q) - (1 - \alpha)H(p) - \alpha H(q)].$$

EK 3 – Çarpık JSD'nin Dışbükeyliği ve Sürekliliği Üzerine

Çarpık JSD fonksiyonu negatif entropi $F(x) = -H(x)$ cinsinden Eşitlik 21'de verilmiştir. Negatif entropi fonksiyonunun $F(x)$ ikinci dereceden türevi $x > 0$ için pozitif ($F''(x) > 0$) olduğundan tam dışbükeydir:

$$\begin{aligned} F(x) &= \sum_i x_i \log x_i \\ F''(x) &= \frac{\log e}{x} \end{aligned}$$

Çarpık JSD ise hem dışbükey hem içbükey fonksiyonların kombinasyonlarından oluştuğundan dışbükey veya içbükey bir fonksiyon değildir.

$$J_{\alpha}^F(p||q) = \underbrace{-F((1-\alpha)p + \alpha q)}_{\text{içbükey}} + \underbrace{(1-\alpha)F(p)}_{\text{içbükey}} + \underbrace{\alpha F(q)}_{\text{dışbükey}} \quad (21)$$

Çarpık JSD $J_{\alpha}^F(p||q)$, $\frac{1}{\alpha(1-\alpha)}$ ifadesi ile ölçeklendirilerek $\alpha \in [0, 1]$ için sürekli olan çarpık JSD fonksiyonları elde edilmiştir. $\alpha \in \{0, 1\}$ sınır değerleri için $\lim_{\alpha \rightarrow 0} \frac{1}{\alpha} J_{\alpha}^F(p||q) = F(p) - F(q) - \langle p - q, \nabla F(q) \rangle$ ve $\lim_{\alpha \rightarrow 1} \frac{1}{\alpha} J_{\alpha}^F(p||q) = F(q) - F(p) - \langle q - p, \nabla F(p) \rangle$ [109]. ∇F , F 'in gradyanı, $\langle \cdot, \cdot \rangle$ vektörlerin iç çarpımıdır.

$$J_{\alpha}(p||q) = \begin{cases} \frac{1}{\alpha(1-\alpha)} J_{\alpha}^F(p||q), & \alpha \neq \{0, 1\} \\ F(p) - F(q) - \langle p - q, \nabla F(q) \rangle, & \alpha = 0 \\ F(q) - F(p) - \langle q - p, \nabla F(p) \rangle, & \alpha = 1 \end{cases}$$

EK 4 – Etiket Düzleştiren Düzenleştirme

Bu bölümde etiket düzleştiren düzenleştirmenin $\alpha \in [0, 1]$ için α 'nın bir uç değeri olan $\alpha \rightarrow 1$ için çarpık JSD fonksiyonuna eşit olduğu gösterilmiştir. $\alpha \rightarrow 1$ için çarpık JSD fonksiyonu iki olasılıksal dağılım arasındaki KL ıraksamasına eşittir:

$$\begin{aligned} \lim_{\alpha \rightarrow 1} J_{\alpha}(p||q) &= \lim_{\alpha \rightarrow 1} \frac{1}{\alpha(1-\alpha)} [(1-\alpha)D_{KL}(p||((1-\alpha)p + \alpha q)) + \alpha D_{KL}(q||((1-\alpha)p + \alpha q))] \\ &= \lim_{\alpha \rightarrow 1} \frac{1}{\alpha(1-\alpha)} [(1-\alpha)D_{KL}(p||q) + \alpha D_{KL}(q||q)] \\ &= \lim_{\alpha \rightarrow 1} \frac{1}{\alpha} D_{KL}(p||q) \\ &= D_{KL}(p||q) \end{aligned}$$

$p = \mu$ ve $q = p_\theta$ için,

$$\begin{aligned}
J_1(u||p_\theta(y|\mathbf{X})) &= D_{KL}(\mu||p_\theta) \\
&= \sum_{y \in Y} u(y) \log \frac{u(y)}{p_\theta(y|\mathbf{X})} \\
&= \sum_{y \in Y} u(y) \log u(y) - \sum_{y \in Y} u(y) \log p_\theta(y|\mathbf{X}) \\
&= \log|Y| - \sum_{y \in Y} u(y) \log p_\theta(y|\mathbf{X}) \\
&= - \sum_{y \in Y} u(y) \log p_\theta(y|\mathbf{X}) + N \\
&= H(\mu||p_\theta) + N
\end{aligned}$$

Model eğitiminde düzenleyici olarak $J_1(u||p_\theta(y|\mathbf{X}))$ kullanıldığında Eşitlik 22 fonksiyonu elde edilmiştir. N , sabit terim olduğundan ifadenin θ 'ya göre optimizasyonu etiket düzleştirici düzenleyiciye (label smoothing regularization) eşittir.

$$\begin{aligned}
\mathcal{L}(\theta) &= H(\mathbf{y}||p_\theta) + H(u||p_\theta) + N \\
&= \mathcal{L}_s(\theta)
\end{aligned} \tag{22}$$

EK 5 – Aşırı Güveni Cezalandırma (Confidence Penalty)

Bu bölümde aşırı güveni cezalandırma yaklaşımının $\alpha \in [0, 1]$ için α 'nın bir diğer uç değeri olan $\alpha \rightarrow 0$ için çarpık JSD fonksiyonuna eşit olduğu gösterilmiştir. $\alpha \rightarrow 0$ için çarpık JSD fonksiyonu iki olasılıksal dağılım arasındaki KL ıraksamasına eşittir:

$$\begin{aligned}
\lim_{\alpha \rightarrow 0} J_{\alpha}(p||q) &= \lim_{\alpha \rightarrow 0} \frac{1}{\alpha(1-\alpha)} [(1-\alpha)D_{KL}(p||(1-\alpha)p + \alpha q) + \alpha D_{KL}(q||(1-\alpha)p + \alpha q)] \\
&= \lim_{\alpha \rightarrow 0} \frac{1}{\alpha(1-\alpha)} [(1-\alpha)D_{KL}(p||p) + \alpha D_{KL}(q||p)] \\
&= \lim_{\alpha \rightarrow 0} \frac{1}{(1-\alpha)} D_{KL}(q||p) \\
&= D_{KL}(q||p)
\end{aligned}$$

$p = \mu$ ve $q = p_{\theta}$ için,

$$\begin{aligned}
J_0(p_{\theta}(y|\mathbf{X})||\mu) &= D_{KL}(p_{\theta}||\mu) \\
&= \sum_{y \in \mathbf{Y}} p_{\theta}(y|\mathbf{X}) \log \frac{p_{\theta}(y|\mathbf{X})}{u(y)} \\
&= \sum_{y \in \mathbf{Y}} p_{\theta}(y|\mathbf{X}) \log p_{\theta}(y|\mathbf{X}) - \sum_{y \in \mathbf{Y}} p_{\theta}(y|\mathbf{X}) \log(u(y)) \\
&= -H(p_{\theta}(y|\mathbf{X})) - \log \frac{1}{|\mathbf{Y}|} \sum_{y \in \mathbf{Y}} p_{\theta}(y|\mathbf{X}) \\
&= -H(p_{\theta}(y|\mathbf{X})) - \log \frac{1}{|\mathbf{Y}|} \\
&= -H(p_{\theta}(y|\mathbf{X})) + N
\end{aligned}$$

Model eğitiminde düzenlileştirici olarak $J_0(p_{\theta}(y|\mathbf{X})||\mu)$ kullanıldığında elde edilen Eşitlik 23, aşırı güveni cezalandırmaya eşittir.

$$\begin{aligned}
\mathcal{L}(\theta) &= H(\mathbf{y}||p_{\theta}) - \beta H(p_{\theta}(y|\mathbf{X})) + N \\
&= \mathcal{L}_m(\theta)
\end{aligned} \tag{23}$$

Anomaly Detection for Solder Joints Using β -VAE

Furkan Ulger[✉], Seniha Esen Yuksel[✉], and Atila Yilmaz

Abstract—In the assembly process of printed circuit boards (PCBs), most of the errors are caused by solder joints in surface mount devices (SMDs). In the literature, traditional feature extraction-based methods require designing hand-crafted features and rely on the tiered red green blue (RGB) illumination to detect solder joint errors, whereas the supervised convolutional neural network (CNN)-based approaches require a lot of labeled abnormal samples (defective solder joints) to achieve high accuracy. To solve the optical inspection problem in unrestricted environments with no special lighting and without the existence of error-free reference boards, we propose a new beta-variational autoencoder (beta-VAE) architecture for anomaly detection that can work on both integrated circuit (IC) and non-IC components. We show that the proposed model learns disentangled representation of data, leading to more independent features and improved latent space representations. We compare the activation and gradient-based representations that are used to characterize anomalies and observe the effect of different beta parameters on accuracy and untwining the feature representations in beta-VAE. Finally, we show that anomalies on solder joints can be detected with high accuracy via a model trained directly on normal samples without designated hardware or feature engineering.

Index Terms— β -variational autoencoder (VAE), automated optical inspection (AOI), solder joint inspection (SJI), unsupervised anomaly detection, VAE.

NOMENCLATURE

X	Data point (image).
u	Vector of means.
σ	Vector of standard deviations.
z	Vector of latent variables.
ϕ, θ	Encoder and decoder model parameters.
$p_{\theta}(X)$	Marginal likelihood of data.
$p_{\theta}(X z)$	Likelihood of data.
$p_{\phi}(z X)$	True posterior distribution of latent variables.
$q_{\phi}(z X)$	Approximate posterior dist. of latent variables.
α, γ	Weights for gradient loss.
β	Kullback–Leibler (KL) divergence weight.
ϵ	Random variable for reparametrization.

Manuscript received April 17, 2021; revised August 13, 2021; accepted September 21, 2021. Date of publication October 18, 2021; date of current version December 16, 2021. This work was supported by Aselsan Inc., through a project between Hacettepe University and Aselsan Inc. Recommended for publication by Associate Editor I. Fidan upon evaluation of reviewers' comments. (*Corresponding author: Furkan Ulger.*)

Furkan Ulger is with the Department of Electrical and Electronics Engineering, Hacettepe University, 06800 Ankara, Turkey, and also with Aselsan Inc., 06830 Ankara, Turkey (e-mail: furkan.ulger@stu.ce.hacettepe.edu.tr).

Seniha Esen Yuksel and Atila Yilmaz are with the Department of Electrical and Electronics Engineering, Hacettepe University, 06800 Ankara, Turkey (e-mail: cyuksel@ee.hacettepe.edu.tr; ayilmaz@ee.hacettepe.edu.tr).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TCPMT.2021.3121265>.

Digital Object Identifier 10.1109/TCPMT.2021.3121265

2156-3950 © 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See <https://www.ieee.org/publications/rights/index.html> for more information.

I. INTRODUCTION

IN THE electronics manufacturing industry, surface mount device (SMD) assembly lines produce printed circuit boards (PCBs) with high density at a very fast rate, which makes them prone to errors. Detecting solder joint errors in the early stages of the assembly is critical to reduce rework costs in production and scrap rates. However, this task is difficult due to the highly varied appearance of PCBs, given that even the electronic components used for the same purpose differ enormously in appearance. Furthermore, different types of solder joint errors can occur, such as solder bridging, excessive/insufficient solder, pseudosolder, and solder skips. In addition, quality requirements standardized by the Institute for Printed Circuits (IPC) should be met for high-quality production. In order to fulfill this need, automated optical inspection (AOI) devices are used in assembly lines. These devices achieve very high recognition rates but require dedicated hardware. There are surveys [1], [2] that elaborate on the assembly process and the importance of solder joint inspection (SJI) in the manufacturing industry.

With the advancement in deep learning, a convolutional neural network (CNN), a powerful algorithm for feature extraction, is used for the inspection of solder joints. Despite achieving high accuracy, supervised CNN-based approaches require a lot of labeled defective (abnormal) samples for training for which there is a huge lack. Moreover, abnormalities may occur due to various reasons, and assume diverse shapes that might be hard to capture in a data collection. Yet, to the best of our knowledge, there is no open dataset that has enough abnormal samples, possibly due to confidential reasons.

Therefore, in this article, we propose to treat the solder joint defect detection as an anomaly detection problem and resort to deep generative models, specifically to β -variational autoencoder (VAE) [3], which enables learning solely from normal samples. β -VAE is a variation of VAE [4], [5] that maximizes the probability of generating real data while balancing the likelihood and KL divergence terms with an adjustable hyperparameter β . The β term puts an emphasis on learning statistically independent latent factors that lead to disentangled representations where one single latent unit is sensitive to changes in a single generative factor. This disentanglement property of the β -VAE leads to more successful and more interpretable representations [6]. Once the model describing the normal data is estimated via β -VAE, we generate an anomaly score for each test sample and apply a decision rule to detect anomalies. For generating the anomaly scores, we employ gradient-based representations and activation-based representations, such as the reconstruction error. Finally, the model specifically learned to describe normal data is expected to yield a low anomaly score for normal

data and a high anomaly score for abnormal data. Hence, we detect these anomalies by comparing the resultant score against a threshold. With better disentangled representations obtained using β -VAE, we show that anomalies on solder joints can be detected with high accuracy via a model trained directly on normal samples. Furthermore, we compare this algorithm to the other state-of-the-art approaches, namely, the VAE, adversarial autoencoder (AAE), and the convolutional autoencoder (CAE), and show the increase in detection rates. To show the advantage of disentangled learning, we also apply t-distributed stochastic neighbor embedding (t-SNE) to visualize the latent space representations and demonstrate that the samples from the same class appear in clusters, the classes are more separable, and this leads to higher accuracy.

Approaching the problem as anomaly detection brings several advantages.

- 1) Employing error-free solder joints is enough, and there is no need for defective solder joints in training. Considering that collecting all variations of defective solder joint types in a dataset is hard, whereas error-free solder joints are amply available, anomaly detection can lead to more generalizable results.
- 2) A single model is trained to maximize the probability of reconstructing error-free solder joints. This same model will fail to reconstruct abnormal solder joints; hence, they will yield a high anomaly score. Therefore, the algorithm is convenient for inspection.
- 3) It is often safer to detect any data departing from the normal data distribution.
- 4) Finally, for many applications, there is a need to carry out the inspection on the field. We believe that allowing SMD inspection without using the hemispherical/circular red green blue (RGB) illuminator can enhance the portability of AOI devices.

The main contributions of this study are given in the following.

- 1) We attempt to solve a unique SJI problem in unrestricted domains; as such, it does not require specialized lighting, feature engineering, and abnormal samples in training nor an error-free reference board.
- 2) We design a β -VAE architecture tailored for this problem, analyze it with different anomaly scoring techniques, and investigate the effect of β for SJI. Furthermore, we show the benefits of using β -VAE and show how the latent space representations of normal and abnormal solders become visibly separated.
- 3) Our model can inspect both integrated circuit (IC) and non-IC component solder joints with a single architecture. To the best of our knowledge, there is no such study within the scope of anomaly detection.

II. RELATED WORK

The literature for SJI can be divided into two main groups, namely, the methods based on feature extraction and the methods based on deep learning. Feature extraction-based methods can mostly either detect IC or non-IC solder joint errors since there are no discriminative features presented for both types



Fig. 1. Surface mount device (SMD) parcellation (a) non-IC component image and its solder pad and component body and (b) IC component image and its body, lead, and solder pad regions.

of components. Besides, their successful applications mostly rely on a circular or hemispherical tiered RGB illumination for inspection of both non-IC component solder joints [7]–[12] and IC component solder joints [13], [14]. On the other hand, there are some studies that do not require an RGB illuminator but mostly an error-free reference image to inspect solder joints [15]–[18]. The tiered illumination is used to generate color contours on the solder paste, making use of the specular surface characteristics of the solder defects [19]. These color contours are then analyzed via handcrafted features based on both color and shape to classify solder-joint errors. The most prominent color features are the average intensity and the percentage of a highlight of each binarized RGB channel [8], [9]. Geometric (shape) features in frequent use can be listed as the solder and polarity areas [10], barycenter and distribution features [11], and the template matching feature obtained by normalized cross correlation [15]. Other features include wavelet coefficients [16] and Gabor features [18]. Following the feature extraction step, the traditional classifiers employed in the literature include the Bayes classifier [7], [9], multilayer perceptron [14], [16], learning vector quantization [11], [14], and support vector machines (SVMs) [9], [12], [20]; followed by the multistage classifiers [7], [9] to improve accuracy. In practice, obtaining such reflections on a solder joint to extract these features is difficult; it requires special lighting and prior or manual input to parse the solder regions, as shown in Fig. 1. In addition, traditional feature extraction-based methods require determining distinctive features for each type of solder-joint errors.

In recent years, there have been a few studies that employ deep learning-based methods to inspect solder joints [20]–[22]. Dai *et al.* [20] employed the you only look once (YOLO) detector for solder localization and used semisupervised learning for classification with SVM. Cai *et al.* [21] used cascaded CNNs: one learns the region of interest where solder defect is probable, and the others are used to classify the solder joint of IC component. These studies employed both anomalous and normal data, and still used the tiered RGB illumination. Since anomalous data is rare, it is difficult to gather a representative training set for a supervised CNN to learn to identify all types of solder joint errors. In addition, commonly inspected PCB defects in the literature are good solder, insufficient solder, excessive solder, pseudosolder, missing solder, missing component, and tombstone; however, there are more different types of possible PCB defects, such as shifted and misplaced components, solder balling, solder flags, pinholes, solder dis-

coloration, billboarding, dewetting, and the like. Therefore, an alternative way to handle solder error detection would be the anomaly detection perspective.

For anomaly detection, autoencoder [23] is a widely employed network that can learn to model complex data distributions. Autoencoders are trained directly on normal (error-free) data to minimize the reconstruction error such that, an encoding network, which strongly reduces the input to a compressed form, and a decoding network, which reconstructs its output to resemble the original input, are simultaneously trained. In the context of SJI, Mujeeb *et al.* [22] used a linear autoencoder on RGB illuminated solder joints and proposed using the L_2 distance between the features of a reference image and a defective image as a similarity measure. However, the method requires a reference image that is a good (error-free) sample of the defective sample and tiered RGB illumination. Goto *et al.* [24] trained AAE for anomaly detection of solder joints that are captured by an X-ray. The AAE employs an autoencoder to generate a latent vector and reconstruct an image from it. In addition, it uses a discriminator that predicts whether the input latent vector is sampled from a prior distribution over the latent vector or generated by the autoencoder. The AAE is trained with adversarial training to impose a prior distribution over the latent variables instead of the KL-divergence as in the VAE. Then, the degree of abnormality is measured by Hotelling's T-squared method. Also, Zheng *et al.* [25] proposed using multitask transfer learning for anomaly detection of solder paste that can conform to the physical changes in PCBs. This method requires finding similar tasks to transfer knowledge and clustering them.

Although VAE is already being used for anomaly detection in different contexts, such as dermatology [26] and medical imaging [27], to the best of our knowledge, we are the first to use it for solder error detection and offer a customized architecture for this problem. Also, our problem is unique, in which we consider both IC and non-IC solder joints, and we do not require any special lighting nor do we require an error-free reference image. In doing so, we want to develop a more general solder-defect classifier that is not restricted to laboratory environments.

III. METHODS

We explain the VAE and β -VAE, and discuss the scoring techniques that assign an anomaly score to each test data.

A. VAE and β -VAE

VAE is a generative model that optimizes the lower bound on the marginal likelihood of each data point (image) X . It consists of a probabilistic encoder and a decoder where ϕ and θ are the encoder and decoder parameters, respectively, which include weights and biases, as shown in Fig. 2. The encoder approximates the true posterior distribution $p_\phi(z|X)$ whose distribution is assumed to be a Gaussian, where z is the latent (unobserved) vector of variables. The mean and standard deviation of the approximate posterior $q_\phi(z|X)$ are the output of the encoder.

We also let the prior distribution over latent variables z have a standard Gaussian distribution $p(z) = N(0, I)$. Then, z is sampled from the posterior distribution as $z \sim q_\phi(z|X) = N(\mu, \sigma)$. The decoder is a generator that reconstructs \hat{X} from z . However, we cannot backpropagate the loss function to train the model since sampling operation is not differentiable. Therefore, a reparametrization trick is introduced. This trick uses a random variable ϵ that is sampled from $N(0, I)$, and it is transformed to random variable z shifted by the mean and scaled by the standard deviation, i.e., $z = \mu + \sigma \odot \epsilon$ that becomes a sample from the distribution $z \sim N(\mu, \sigma)$. The illustration of VAE architecture is given in Fig. 2.

To train VAE, the objective is finding the parameters θ and ϕ that maximize the evidence lower bound (ELBO) \mathcal{L} of a data point as follows:

$$\log p_\theta(X) \geq \frac{1}{L} \sum_{\ell=1}^L \log p_\theta(X|z^{(\ell)}) - \beta D_{KL}(q_\phi(z|X)||N(z; 0, I)). \quad (1)$$

The first term on the right-hand side of this equation is maximizing the probability of reconstructing the input X , which corresponds to reconstruction loss. L is the number of samples per data point to use for the Monte Carlo estimator of the reconstruction loss and set to 1 [4]. The second term is minimizing the KL divergence, which is the distance between the approximated posterior (encoder's distribution) and fixed Gaussian distribution with zero mean and unit variance. This leads to the lower bound on the probability density function of our data, as shown in (1). We want to find the parameters (weights and biases) that maximize this objective function. Note that, in the original VAE, $\beta = 1$.

β -VAE introduces a weighting factor β that balances reconstruction accuracy and latent channel capacity in (1). With $\beta > 1$, stronger pressure for the posterior $q_\phi(z|X)$ to be close to the standard Gaussian prior $p(z)$ is introduced. This limits the capacity of latent z to learn a better disentanglement. β -VAE with $\beta = 1$ corresponds to the original VAE. Higgins *et al.* [3] demonstrated that β -VAE with the hyperparameter $\beta > 1$ outperforms VAE on various datasets since it forces learning a more efficient latent representation of the data. This provides increased disentanglement performance. However, there is a tradeoff between reconstruction accuracy and learned disentanglement quality within the learned latent representations. A high β value can lead to poorer reconstruction due to the loss of information while passing through the restricted latent channel. Therefore, it is important to find a balance that ensures a disentangled representation while still being able to reconstruct X . On the other hand, β -VAE requires only tuning β hyperparameter that can be estimated heuristically.

B. Anomaly Score

In testing, a sample X from the test set is inputted to the encoder of the trained VAE. The encoder outputs a vector of means and standard deviations of each image. The reparametrization trick is applied to get the latent vector z ,

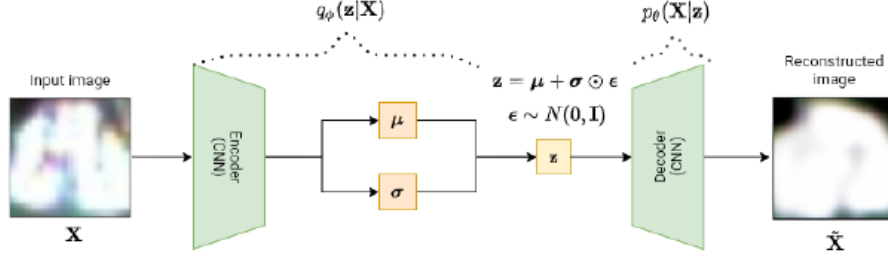


Fig. 2. VAE architecture with isotropic Gaussian prior distribution over latent vector.

which is sampled from the Gaussian distribution with the mean and standard deviation computed by the encoder. Then, the decoder reconstructs the input image from $p_\theta(\mathbf{X}|\mathbf{z})$. The first term in (1) forms the reconstruction loss $\mathcal{A}(\mathbf{X})_{\text{Recon}}$. It is calculated as the anomaly score from $\log p_\theta(\mathbf{X}|\mathbf{z})$ that corresponds to the squared Euclidean distance between the input image and reconstructed image, as given in (2), where $\tilde{\mathbf{X}}$ is the reconstructed image. VAE trained on only normal samples is expected to be able to reconstruct normal samples with low reconstruction error and yield low anomaly score, whereas abnormal samples are reconstructed with high reconstruction error and yield high anomaly score

$$\mathcal{A}(\mathbf{X})_{\text{Recon}} = (\mathbf{X} - \tilde{\mathbf{X}})^2. \quad (2)$$

The objective is maximizing the lower bound on the marginal likelihood of \mathbf{X} ; hence, anomaly score can also be calculated by $\mathcal{A}(\mathbf{X})_{\text{ELBO}} = -\log p_\theta(\mathbf{X})$. Since the data are continuous and encoder's distribution is assumed to have a Gaussian distribution, KL divergence can be computed in closed form as the second term of the right-hand side of (3) as derived by Kingma and Welling [4]. J is the dimensionality of \mathbf{z}

$$\mathcal{A}(\mathbf{X})_{\text{ELBO}} = -(\mathbf{X} - \tilde{\mathbf{X}})^2 - \frac{1}{2} \sum_{j=1}^J (1 + \log((\sigma_j)^2) - (\mu_j)^2 - (\sigma_j)^2). \quad (3)$$

Kwon *et al.* [28] proposed training autoencoders with gradient constraint (GradCon) to model normal data distributions and using gradient-based representations for anomaly detection, motivated to capture information unavailable in the activation-based network representation. Anomalies require more model parameter updates to be represented compared to normal data; hence, backpropagated gradients are used to characterize how much model update is required by input to detect anomalies. The gradients of each decoder layer i with respect to model parameters, $(\partial \mathcal{L})/(\partial \theta_i)$, are calculated by backpropagating the reconstruction loss. The algorithm combines reconstruction and gradient loss as an anomaly score

$$\mathcal{A}(\mathbf{X})_{\text{GradCon}} = \mathcal{A}(\mathbf{X})_{\text{Recon}} + \gamma \mathcal{L}_{\text{grad}} \quad (4)$$

where γ is a scalar and $\mathcal{L}_{\text{grad}}$ is cosine similarity, i.e., the cosine of the angle between average training gradients and

gradient loss of current (k th) iteration as follows:

$$\mathcal{L}_{\text{grad}} = -\mathbf{E}_i \left[\cos \text{SIM} \left(\frac{\partial J^{k-1}}{\partial \theta_{i_{\text{avg}}}}, \frac{\partial \mathcal{L}^k}{\partial \theta_i} \right) \right] \quad (5)$$

$$\frac{\partial J^{k-1}}{\partial \theta_{i_{\text{avg}}}} = \frac{1}{k-1} \sum_{t=1}^{k-1} \frac{\partial J^t}{\partial \theta_i}.$$

J is the loss function that is calculated during training, \mathcal{L} is ELBO, and α is a weight for the gradient loss defined as

$$J = \mathcal{L} + \alpha \mathcal{L}_{\text{grad}}. \quad (6)$$

This anomaly detection algorithm using GradCon for training autoencoders and using a combination of reconstruction error and gradient loss as an anomaly score is called GradCon.

The anomaly score for the AAE is calculated by the squared Mahalanobis distance between the normal data distribution and each test sample over the latent variables, as given in (7). \mathbf{z} is the latent vector of each test sample, $\boldsymbol{\mu}$ is the mean of latent vectors of the normal data, and $\boldsymbol{\Sigma}^{-1}$ is the inverse of the covariance matrix of the normal data

$$T^2 = (\mathbf{z} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{z} - \boldsymbol{\mu}). \quad (7)$$

C. Decision Rule

After calculating the anomaly score, a decision rule is used to determine whether the sample is an anomaly or not by comparing against a threshold [29]. Anomaly score is expected to be low for normal samples and high for never-seen abnormal samples. The threshold for the decision rule is considered a hyperparameter and calculated by a sum of the average anomaly score on the validation set and two times the standard deviation of the scores where most of the normal data is spread. The samples with a higher anomaly score than the threshold are evaluated as an anomaly.

IV. EXPERIMENT

In this section, we describe our dataset and the preprocessing methods that we employ. Then, we provide comparisons between VAE, β -VAE, AAE, and the CAE. The CAE is an unsupervised model that learns a low-dimensional representation of the input [30], which is selected due to its characteristic to separate normal from abnormal data. The AAE imposes a prior distribution over latent variables similar to the VAE.

A. Dataset and Preprocessing

Our dataset has 2619 normal training samples whose 10% is set aside as a validation set to be used for tenfold cross-validation and 183 test samples (66 abnormal and 117 normal). Some abnormal samples are obtained from the open dataset [31]. The abnormal (defective solder joints) samples in the test set include solder bridge, insufficient solder, excessive solder, solder flags, solder dewetting, pinholes, flux residues, missing, and shifted components. Ground-truth labels of these samples are only provided during testing to evaluate anomaly detection performance. The dataset is obtained by segmenting both IC and non-IC solder joints from solder pad regions of several PCBs, as shown in red in Fig. 1. Solder joints are labeled manually using the LabelMe software [32]. Since the inspection is carried out on solder pads, only the relevant parts are labeled to avoid segmentation errors. Finally, the dataset is inspected thoroughly against any segmentation errors, corrupted images, duplicate examples, and blurriness.

For data preprocessing, our data are cropped from the short side to preserve the integrity of solder joint images and resized to 64×64 . Then, the data are normalized to zero mean, unit variance to have the range from -1 to 1 . Some samples are horizontally flipped randomly as data augmentation that is a regularization strategy to prevent overfitting but without increasing the dataset size.

B. Architecture and Hyperparameters

Both VAE and β -VAE share the same architecture since the only difference is the added β term in the objective function. These architectures were shown in Fig. 2 and in more detail with all the parameter sizes in Fig. 3. For comparison, the AAE is tested with both the proposed architecture in [24] and finetuned by changing the number of layers, filters, and size of the latent dimension; then, the highest achieved accuracy is reported. The CAE, on the other hand, has the same encoder-decoder structure with β -VAE except that there is a single convolutional layer at the bottleneck to compress the input. The encoder, which approximates the mean and standard deviation, has five convolutional layers with an increasing number of filters from 3 to 256 and two separate convolutional layers at the bottleneck have ten filters. The decoder has transposed convolutional layers as a counterpart of the encoder. The convolutional layers have a filter size of 3×3 and stride 1. Convolutional layers are followed by batch normalization layers that normalize the feature map to zero mean and unit variance [33]. Leaky rectified linear unit (LReLU) is applied for nonlinearity to all the layers except for the output layer of the decoder, which uses Tanh. Max pooling layers are used to reduce the spatial dimension of feature maps by downsampling, while the upsampling layers increase the spatial size of the image by using bicubic interpolation. The VAE architecture is given in Fig. 3 with all the hyperparameters. As a convention, $\text{Conv}_{a,b,c}$ and $\text{Tconv}_{a,b,c}$ are the convolutional and the transposed convolutional layers, respectively, where a is the number of channels in the input image, b is the number of channels produced by the convolution, and c is the size of the $c \times c$ convolution kernel. Max pooling_{2,2} shows the

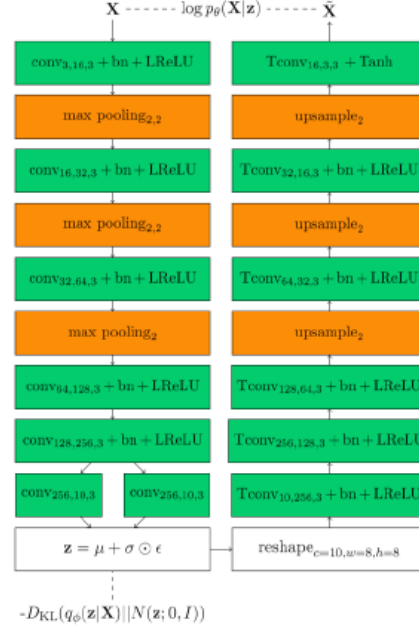


Fig. 3. Sequence of blocks in the first column denote the encoder with five convolutional (conv) and three max pooling layers, and a convolutional bottleneck to obtain the parameters for the sampling of the latent vector z . Blocks in the second column form the decoder that uses six transposed convolution (Tconv) and three upsampling layers to reconstruct the input image from z . The terms bn and LReLU stand for batch normalization and leaky ReLU, respectively.

TABLE I
HYPERPARAMETER SETTINGS FOR β -VAE

Hyperparameters	Value
Initial learning rate	$1e-2$
Learning rate decay when loss plateaus	$\gamma = 0.1$
Number of Epochs	100
Optimizer	Adam
Weight decay	$\lambda = 1e-4$
Batch Size (N)	64
β (KL weight)	3
Latent space dimension	640

max pooling filter size and stride, while upsample_2 provides bicubic interpolation sampling with a scale factor of 2. The latent vector is reshaped to the channel size $10 \times 8 \times 8$ to generate the input image from latent vector z .

The model is trained with the Adam optimizer [34] for 100 epochs with an initial learning rate of $1e-2$ that is decayed as the validation loss stops decreasing. To prevent overfitting to the training dataset, we resorted to $L2$ regularization that penalizes the high weights on features. The selected batch size is 64. β (KL weight) is selected as greater than one to learn better disentanglement of data, the effect of which will

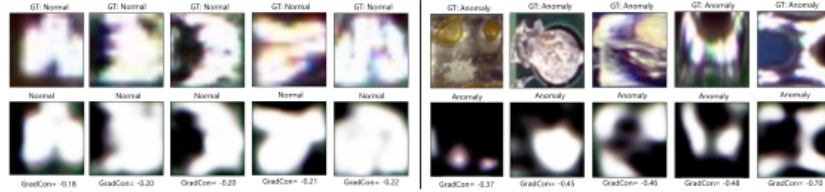


Fig. 4. Reconstructed normal and abnormal samples. The first row shows original input images with corresponding ground-truth labels, and the bottom row shows the reconstructions along with predicted labels and anomaly scores.

TABLE II
COMPARISON OF RECONSTRUCTION-BASED AND GENERATIVE MODELS FOR DIFFERENT ANOMALY SCORES (AVERAGE OF TEN RUNS)

Method	CAE		VAE			β -VAE		AAE [24]	
	Recon	GradCon	Recon	ELBO	GradCon	Recon	ELBO		
Precision	0.768 \pm 0.008	0.73 \pm 0.003	0.759 \pm 0.02	0.737 \pm 0.02	0.753 \pm 0.02	0.719 \pm 0.01	0.728 \pm 0.02	0.767 \pm 0.03	0.648 \pm 0.02
Recall	0.672 \pm 0.009	0.708	0.775 \pm 0.02	0.799 \pm 0.02	0.789 \pm 0.02	0.817 \pm 0.02	0.801 \pm 0.02	0.78 \pm 0.03	0.737 \pm 0.03
F1-score	0.717 \pm 0.007	0.719 \pm 0.002	0.766 \pm 0.02	0.766 \pm 0.01	0.770 \pm 0.01	0.764 \pm 0.01	0.762 \pm 0.01	0.774 \pm 0.02	0.689 \pm 0.02

TABLE III
EFFECT OF HYPERPARAMETER β ON ACCURACY FOR β -VAE TRAINED WITH GRADCON

Beta (β)	0.01	0.1	1 (VAE)	3	10
Precision	0.705 \pm 0.01	0.709 \pm 0.01	0.775 \pm 0.01	0.767 \pm 0.03	0.647 \pm 0.04
Recall	0.712 \pm 0.01	0.752 \pm 0.02	0.752 \pm 0.02	0.78 \pm 0.03	0.686 \pm 0.05
F1-score	0.708 \pm 0.01	0.73 \pm 0.01	0.763 \pm 0.01	0.774 \pm 0.02	0.665 \pm 0.04

be more deeply analyzed in Section V. Finally, the latent z dimension with the size of 640 is selected since too small z cannot model the marginal likelihood of a data point, and too big z degrades the overall accuracy since learned latent representation becomes indistinguishable for abnormal and normal samples. These hyperparameters given in Table I are determined on the validation set.

V. RESULTS

Original versus reconstructed normal and abnormal samples are given in Fig. 4. Normal samples yield a low anomaly score, whereas abnormal samples yield a high anomaly score. Precision, recall, and F1-score are used as evaluation metrics. Precision rate shows what proportion of actual anomalies are detected correctly, and precision is the rate for the correctness of these estimations. F1-score is calculated to show the overall accuracy of the model from these metrics. All the results are an average of ten runs. We resorted to tenfold cross-validation due to the confined dataset; thereby, each sample in the dataset is being tested. In each successive round, a different part of the dataset is held out for validation, while the remaining is used for training, and the model with the least validation error is selected. The final model is evaluated on the test set.

The generative models, VAE and β -VAE, are compared based on the evaluation metrics given in Table II, namely, the reconstruction loss (Recon), ELBO, and the GradCon anomaly score (the combination of reconstruction and

gradient loss). The CAE model is evaluated with reconstruction loss and GradCon. The highest accuracy is obtained for β -VAE for $\beta = 3$ with a score of 0.774 with GradCon anomaly, outperforming CAE, VAE, and AAE. The ELBO loss is on par with the reconstruction loss for VAE and β -VAE. GradCon made improvements over the generative models with activation-based representations. The AAE is evaluated by using the Mahalanobis distance as an anomaly score. Although the AAE yields sharper reconstructions than the VAE due to the adversarial training, it achieves lower accuracy.

Disentangled representations emerge when the right balance is found between the reconstruction accuracy and latent capacity restriction [3]; hence, the effect of different β term is investigated on β -VAE that is trained with GradCon, as shown in Table III. $\beta = 3$ is selected for β -VAE, which yields the highest F1-score. Using larger or smaller β values also led to lower accuracy, while larger beta values resulted in poorer reconstruction accuracy.

In order to see the effect of different β values on the latent space representations, t-SNE [35], which is a nonlinear dimensionality reduction algorithm that is useful for visualizing high dimensional data, is employed. Test images are encoded in the β -VAE that is trained with GradCon and ELBO objective function for different β values; then, the resultant mean vector of the approximated posterior distribution is passed to t-SNE with the perplexity of 5 since the dataset is small. Then, t-SNE maps 640-dimensional latent space into

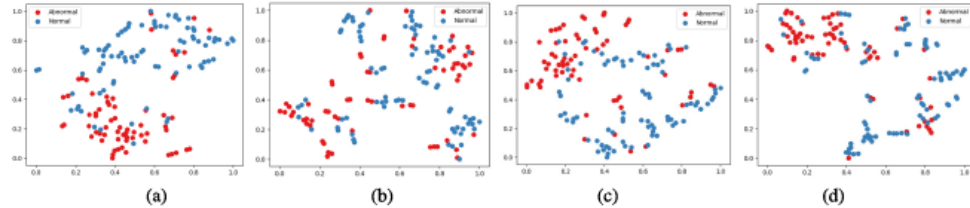


Fig. 5. 2-D latent space representation of β -VAE trained with GradCon for (a) $\beta = 0.1$, (b) $\beta = 1$, (c) $\beta = 3$, and (d) $\beta = 10$ after applying t-SNE. Blue and red circles represent the normal and abnormal test data, respectively.

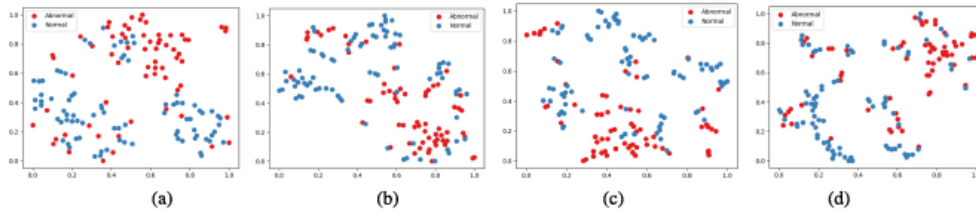


Fig. 6. 2-D latent space representation of β -VAE trained with ELBO for (a) $\beta = 0.1$, (b) $\beta = 1$, (c) $\beta = 3$, and (d) $\beta = 10$ after applying t-SNE. Blue and red circles represent the normal and abnormal test data, respectively.

2-D space. The embeddings are scaled to bring all the values to $[0, 1]$. Latent space visualization of β -VAE that is trained with GradCon and ELBO objective is given in Figs. 5 and 6, respectively. The t-SNE is run 100 times, and the embeddings with the lowest KL divergence are selected for each model. As shown in Figs. 5 and 6, samples from the same class are closer to each other and appear in clusters. For β -VAE that is trained with GradCon, clusters of the classes are entangled, i.e., not easily separable for the β values of 1 and 10, as shown in Fig. 5(b) and (d). This behavior is observed for the β values of 0.1 and 10 for β -VAE trained with ELBO objective, as given in Fig. 6(a) and (d). However, we can see that the classes are more separable for $\beta = 3$ although they are split into parts, as given in Figs. 5 and 6(c) for both models. For the real-world complex samples, it is difficult to achieve clear disentanglement or visualize it in two dimensions, but we can deduce that a better disentangled representation is learned for $\beta = 3$.

VI. CONCLUSION

In this study, solder joint defect detection in PCBs is considered an anomaly detection problem. We propose using β -VAE, which is a generative model for disentangled representation learning, for anomaly detection to detect errors in both IC and non-IC solder joints. β -VAE disentangles factors of variation in data, with which data understanding and classification become easier. We compared different anomaly scoring techniques based on activation and gradient-based representations and investigated the effect of different β values on accuracy. The highest accuracy is achieved when the combination of reconstruction and gradient loss is used as an anomaly score.

Despite the challenging environment of the problem due to the varying shape and small size of solder joints, we show that high accuracy can be achieved without using special lighting, handcrafted features, or labeled data for supervised learning. In the context of SJI where normal samples are in abundance and there are few abnormal samples, we show that the designed β -VAE architecture is effective and leads to disentangled latent space representations.

REFERENCES

- [1] M. Moganti, F. Ercal, C. H. Dagli, and S. Tsunekawa, "Automatic PCB inspection algorithms: A survey," *Comput. Vis. Image Understand.*, vol. 63, no. 2, pp. 287–313, 1996.
- [2] M. Janóczy, A. Becker, L. Jakab, R. Goóf, and T. Takács, "Automatic optical inspection of soldering," in *Materials Science: Advanced Topics*. London, U.K.: InTech, 2013.
- [3] I. Higgins *et al.*, "Beta-VAE: Learning basic visual concepts with a constrained variational framework," in *Proc. ICLR*, 2017, pp. 1–12.
- [4] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*. [Online]. Available: <http://arxiv.org/abs/1312.6114>
- [5] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 1278–1286.
- [6] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.
- [7] T.-H. Kim, T.-H. Cho, Y. S. Moon, and S. H. Park, "Visual inspection system for the classification of solder joints," *Pattern Recognit.*, vol. 32, no. 4, pp. 565–575, Apr. 1999.
- [8] X. Hongwei, Z. Xiannin, K. Yongcong, and O. Gaofei, "Solder joint inspection method for chip component using improved AdaBoost and decision tree," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 1, no. 12, pp. 2018–2027, Dec. 2011.
- [9] H. Wu, X. Zhang, H. Xie, Y. Kuang, and G. Ouyang, "Classification of solder joint using feature selection based on Bayes and support vector machine," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 3, no. 3, pp. 516–522, Mar. 2013.

- [10] F. Wu, X. Zhang, Y. Kuan, and Z. He, "An AOI algorithm for PCB based on feature extraction," in *Proc. 7th World Congr. Intell. Control Autom.*, 2008, pp. 240–247.
- [11] F. Wu and X. Zhang, "An inspection and classification method for chip solder joints using color grads and Boolean rules," *Robot. Comput.-Integr. Manuf.*, vol. 30, no. 5, pp. 517–526, Oct. 2014.
- [12] J.-D. Song, Y.-G. Kim, and T.-H. Park, "SMT defect classification by feature extraction region optimization and machine learning," *Int. J. Adv. Manuf. Technol.*, vol. 101, nos. 5–8, pp. 1303–1313, Apr. 2019.
- [13] F. Wu and X. Zhang, "Feature-extraction-based inspection algorithm for IC solder joints," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 1, no. 5, pp. 689–694, May 2011.
- [14] K. W. Ko and H. S. Cho, "Solder joints inspection using a neural network and fuzzy rule-based classification method," *IEEE Trans. Electron. Packag. Manuf.*, vol. 23, no. 2, pp. 93–103, Apr. 2000.
- [15] A. J. Crispin and V. Rankov, "Automated inspection of PCB components using a genetic algorithm template-matching approach," *Int. J. Adv. Manuf. Technol.*, vol. 35, nos. 3–4, pp. 293–300, Dec. 2007.
- [16] G. Acciani, G. Brunetti, and G. Fornarelli, "Application of neural networks in optical inspection and classification of solder joints in surface mount technology," *IEEE Trans. Ind. Informat.*, vol. 2, no. 3, pp. 200–209, Aug. 2006.
- [17] F. Ulger and S. E. Yuksel, "A standalone open-source system for optical inspection of printed circuit boards," in *Proc. Signal, Algorithms, Architectures, Arrangements, Appl.*, Sep. 2019, pp. 105–110.
- [18] N. S. S. Mur, P. K. D. V. Yarlagadda, and C. Fookes, "Design and development of automatic visual inspection system for PCB manufacturing," *Robot. Comput.-Integr. Manuf.*, vol. 27, no. 5, pp. 949–962, Oct. 2011.
- [19] D. W. Capson and S.-K. Eng, "A tiered-color illumination approach for machine inspection of solder joints," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, no. 3, pp. 387–393, May 1988.
- [20] W. Dai, A. Mujeeb, M. Erdt, and A. Sourin, "Soldering defect detection in automatic optical inspection," *Adv. Eng. Informat.*, vol. 43, Jan. 2020, Art. no. 101004.
- [21] N. Cai, G. Cen, J. Wu, F. Li, H. Wang, and X. Chen, "SMT solder joint inspection via a novel cascaded convolutional neural network," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 8, no. 4, pp. 670–677, Apr. 2018.
- [22] A. Mujeeb, W. Dai, M. Erdt, and A. Sourin, "One class based feature learning approach for defect detection using deep autoencoders," *Adv. Eng. Informat.*, vol. 42, Oct. 2019, Art. no. 100933.
- [23] B. Zong *et al.*, "Deep autoencoding Gaussian mixture model for unsupervised anomaly detection," in *Proc. ICLR*, 2018, pp. 1–19.
- [24] K. Goto, K. Kato, T. Saito, and H. Aizawa, "Adversarial autoencoder for detecting anomalies in soldered joints on printed circuit boards," *J. Electron. Imag.*, vol. 29, no. 4, 2020, Art. no. 041013.
- [25] Z. Zheng *et al.*, "Contextual anomaly detection in solder paste inspection with multi-task learning," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 6, pp. 1–17, Nov. 2020.
- [26] Y. Lu and P. Xu, "Anomaly detection for skin disease images using variational autoencoder," 2018, *arXiv:1807.01349*. [Online]. Available: <http://arxiv.org/abs/1807.01349>
- [27] D. Zimmerer, F. Isensee, J. Petersen, S. Kohl, and K. Maier-Hein, "Unsupervised anomaly localization using variational auto-encoders," in *Proc. MICCAI*, 2019, pp. 289–297.
- [28] G. Kwon, M. Prabhushankar, D. Temel, and G. AlRegib, "Backpropagated gradient representations for anomaly detection," in *Proc. ECCV*, 2020, pp. 206–226.
- [29] G. Boracchi and D. Carrara, "Tutorial: Anomaly detection in images," in *Proc. IEEE ICIP*, Oct. 2020, p. 91.
- [30] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [31] H. Lu, D. Mehta, O. Paradis, N. Asadizanjani, M. Tehranipoor, and D. L. Woodard, "FICS-PCB: A multi-modal image dataset for automated printed circuit board visual inspection," *IACR Cryptol. ePrint Arch.*, vol. 2020, p. 366, Mar. 2020.
- [32] K. Wada. (2016). *Labelme: Image Polygonal Annotation With Python*. [Online]. Available: <https://github.com/wkentaro/labelme>
- [33] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
- [34] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: <http://arxiv.org/abs/1412.6980>
- [35] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2580–2605, 2008.



Furkan Ulger received the B.S. degree in electrical and electronics engineering from Hacettepe University, Ankara, Turkey, in 2019, where he is currently pursuing the M.S. degree.

He is working with Aselsan Inc., Ankara. His research interests include visual inspection, deep learning, image processing, and pattern recognition.



Seniha Esen Yuksel received the B.S. degree in electrical and electronics engineering from Middle East Technical University, Ankara, Turkey, in 2003, the M.S. degree in electrical and computer engineering from the University of Louisville, Louisville, KY, USA, in 2005, and the Ph.D. degree in computer engineering from the University of Florida, Gainesville, FL, USA, in 2011.

She is currently an Associate Professor with the Department of Electrical and Electronics Engineering, Hacettepe University, Ankara. She is also the Director of the Pattern Recognition and Remote Sensing Laboratory (PARRSLAB), Hacettepe University, where she is doing research on machine learning and computer vision with applications in the defense industry. Her latest projects include fusion of hyperspectral and light detection and ranging (LIDAR) data, target detection from multiple sensors, such as multispectral, hyperspectral, and synthetic aperture radar (SAR) satellites, and radar and X-ray images.



Atilla Yilmaz received the M.Sc. degree in control and information technology from the Control Systems Centre, The University of Manchester, Manchester, U.K., in 1992, and the Ph.D. degree from the Biomedical Engineering Division, University of Sussex, Brighton, U.K., in 1996.

He is currently a Professor with the Department of Electrical and Electronics Engineering, Hacettepe University, Ankara, Turkey. His research interests include artificial intelligence and neural networks in control and biomedical applications, especially for image and biomedical signal processing.