

Masterarbeit

**Entwicklung eines Multi-Class Defekt
Segmentierungsalgorithmus für Ladungsträger-Selektive
Solarzellen anhand von photolumineszenz Bilddaten**

Manuel Eckert

(75516)

12. März 2024



Hochschule für Technik, Wirtschaft und Kultur Leipzig
Fakultät Informatik und Medien

angefertigt am

Fraunhofer-Institut für Solare Energiesysteme ISE

Freiburg im Breisgau



Erstprüfer: Prof. Kirian Varanasi
Lehrstuhl für Virtuelle und Erweiterte Realität
HTWK Leipzig

Zweitprüferin: Dr. Theresa Trötschler
Fraunhofer Institut für Solare Energiesysteme (ISE)

Betreuer/in: Dr. Mathias Demant
Fraunhofer Institut für Solare Energiesysteme (ISE)
Dr. Theresa Trötschler
Fraunhofer Institut für Solare Energiesysteme (ISE)
Ms. Alexandra Wörnhör
Fraunhofer Institut für Solare Energiesysteme (ISE)

Eingereicht am: 12. März 2024

Erklärung

Ich versichere wahrheitsgemäß, die Masterarbeit selbständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

.....
Manuel Eckert

Leipzig, den 12. März 2024

Sperrvermerk

Die vorliegende Masterarbeit mit dem Titel „Entwicklung eines Multi-Class Defekt Segmentierungsalgorithmus für Ladungsträger-Selektive Solarzellen anhand von photolumineszenz Bilddaten“ enthält vertrauliche Daten des Fraunhofer Institut für Solare Energie Systeme.

Die Masterarbeit darf nur dem Erst- und Zweitgutachter sowie befugten Mitgliedern des Prüfungsausschusses zugänglich gemacht werden. Eine Veröffentlichung und Vervielfältigung der Masterarbeit ist - auch in Auszügen - nicht gestattet.

Eine Einsichtnahme der Arbeit durch Unbefugte bedarf einer ausdrücklichen Genehmigung des Verfassers und des Fraunhofer Institut für Solare Energie Systeme.

Abstract

Die Heterojunction (HJT) Solarzellen verwenden ein neuartige Bauweise, welche eine höheren Wirkungsgrad im Vergleich zu herkömmlichen Solarzellen haben. Das Potenzial dieser Technologie ist enorm was zu einer vermehrten Forschung selbiger führt. Für eine erfolgreiche Produktion von HJT Solarzellen müssen Methoden zur Qualitätssicherung entwickelt werden. Dies ist das Ziel dieser Arbeit. Der Schwerpunkt liegt dabei auf der Charakterisierung von Produktionsdefekten, mittels Photolumineszenz Messungen, welche während der Produktion entstehen.

Bisher werden im Projekt Produktionsdefekte nur mit herkömmlichen Bildverarbeitungsverfahren und mit überwachten Lernverfahren innerhalb des Deep-Learnings analysiert. Diese können Defekte oftmals sehr genau charakterisieren, bedürfen aber einem Hohen Maß an Anpassung auf die entsprechenden Messdaten. In den vergangenen Monaten und Jahren haben Unüberwachte Lernverfahren des Deep-Learnings enorme Fortschritte in der Vorhersagegenauigkeit erzielt.

Innerhalb dieser Arbeit sollen als Teil einer genauen Charakterisierung die Defekte welche während der Prozessierung der Zellen auftreten segmentiert werden. Weiterhin wird untersucht ob diese Defektstrukturen, mittels Unüberwachten Lernmethoden, in der gleichen Güte erkannt werden können. Dabei werden vielversprechende Deep-Learning Methoden eingesetzt, welche keine Datenpaar zwischen Bild und Annotationen benötigen. Für das Ziel der Segmentation der Defekte wird ein gepaarter Datensatz erstellt, welcher aus Messungen verschiedener Waferrezepten besteht.

Für eine detaillierte Analyse werden die implementierten Modelle sowohl quantitativ als auch qualitativ ausgewertet. Die Auswertung zeigt, dass Deep-Learning Modelle in der Lage sind, Defekte anhand von Photolumineszenz Messungen erfolgreich zu segmentierten

und diese wertvolle Informationen für die Charakterisierung der Zellen enthalten.

Inhaltsverzeichnis

Abbildungsverzeichnis	II
Tabellenverzeichnis	V
1 Einleitung	1
1.1 Motivation	1
1.2 Beitrag dieser Arbeit	4
2 Grundlagen	7
2.1 Struktur der HJT Zelle	7
2.2 Datengrundlage	9
2.3 Messverfahren	11
2.4 Bildsegmentierung	11
2.4.1 Traditionelle Bildverarbeitung	11
2.4.2 Maschinelles Lernen	13
2.5 Neuronalen Netzes	13
2.5.1 Überwachtes Lernen	14
2.5.2 Unüberwachtes Lernen	15
2.5.3 Vision Transformer	16
2.6 Verlustfunktionen	18
2.6.1 Tensorkorrelation	18
2.6.2 Binary Cross Entropy	18
2.7 Metriken	19
3 Übersicht der Defekte und Daten	25
3.1 Datenausprägung	25
3.1.1 Defektbeschreibung	27
3.1.2 Herausforderung der Daten	29
4 Ansatz	31
4.1 Übersicht	31
4.2 U-Net	32
4.2.1 U-Net pretrained	33
4.3 DINO: Self-DIstillation with NO labels	34
4.4 STEGO: Self-supervised Transformer with Energy-based Graph Optimization	36
4.5 STEgeo: Self-supervised Transformer with GEometric feature loss	36

5 Experimentalteil	42
5.1 Gestaltung der Experimente	42
5.2 Datensatz	42
5.2.1 Datensatz Überblick	42
5.2.2 Daten Vorverarbeitung	45
5.3 Segmentationsmodelle	46
5.3.1 Überblick Netzwerktraining	46
5.3.2 Experiment 1: U-Net	47
5.3.3 Experiment 2: DINO	47
5.3.4 Experiment 3: STEGO	47
5.3.5 Experiment 4: STEgeo	47
6 Ergebnisse	48
6.1 Überwachtes Lernen	48
6.1.1 U-Net	48
6.2 Unüberwachtes Lernen	52
6.2.1 DINO	52
6.2.2 STEGO	53
6.3 Vergleich der Modelle	54
6.3.1 Vergleich Experimente unüberwachtes Lernen	54
6.3.2 DINO	54
6.3.3 DINO + STEGO	56
6.3.4 Vergleich Modelle	57
7 Diskussion	59
8 Zusammenfassung und Ausblick	61

Abbildungsverzeichnis

Abbildungsverzeichnis

Abbildung 1.1:	Schematischer Aufbau eines Wafers als Silicon Heterojunction-Solarzelle	1
Abbildung 1.2:	Darstellung des Handlings innerhalb eines Produktionsteilschrittes	2
Abbildung 1.3:	Entwicklung der Moduleffizienz von 2012 bis 2022; es wird der Median und Interquartile Abstand für mono-Si SHJ(HJT), mono-Si und poly-Si Module gezeigt [hjtrendKrÄdling2022]	3
Abbildung 2.1:	Blick in die Produktion mit Transportvorrichtungen wie Greifern und Förderbändern[tray_img23]	8
Abbildung 2.2:	eines Vakuum Greifers mit resultierenden Partikeln. Aufgenommen mit einem Rasterelektronenmikroskop [particle_image_Fischer2019] .	9
Abbildung 2.3:	Korrelation des normalisierten Pseudoverlustes η durch eine bei der Handhabung induzierten Defekt. Diese wird für verschiedene Temperatur und Material abhängige Teilchendichte angegeben [impactParticlesFischerrefId0] PU = Polyurethan , PEEK = Polyetheretherketon	10
Abbildung 2.4:	Exemplarische Darstellung einer PL Messung	12
Abbildung 2.5:	Darstellung einer überwachten Lernstrategie. Daten-Label Paar trainiert Neuronales Netz. Testdaten können mit einem erfolgreich trainierten Netz in korrekt gelernte Klassen überführt werden.	14
Abbildung 2.6:	Beispielhafte Darstellung eines überwachten Lernverfahren. Neuronales Netz trainiert lediglich mit Daten ohne . Testdaten können mit einem erfolgreich trainierten Netz in korrekt gelernte Klassen überführt werden.	16
Abbildung 2.7:	Exemplarische Darstellung eines Vision Transformers [ViTdosovitskiy2021image]	
Abbildung 3.1:	Überblick der PL Messdaten mit verschiedenen Prozess Rezepten .	26
Abbildung 3.2:	Datenbeispiel aus Datensatz <i>HJT _1</i> (links) und <i>SHJ _1</i> (rechts) .	27
Abbildung 3.3:	Überblick der differenzierten Defektstrukturen	28
Abbildung 4.1:	Einzelner Convolution Block in dem modifizierten U-Net. Dieser besteht aus einem conv Schicht, BN und ReLU als Aktivierung. . .	32
Abbildung 4.2:	Schematischer Aufbau eines U-net Modells mit einer Auflösung von 32x32 Pixeln am tiefsten Punkt	33
Abbildung 4.3:	DINO Architektur mit exemplarischen Daten	34

Abbildung 4.4:	STEGO Architektur mit exemplarischen Daten. Berechnung K-Nächste Nachbarn(Oben). Berechnung Loss (Mitte und Links). Darstellung des Segmentations Kopfes(Rechts). Vorhersagemodell(Unten)	37
Abbildung 4.5:	STEgeo Architektur mit exemplarischen Daten. Bildausschnitts Distanzen mit Gewichtungsfunktion(Oben Links) Bildausschnitts Erstellung und Positions berechnung (Oben Rechts). Berechnung Loss (Mitte und Links). Darstellung des Segmentations Kopfes(Rechts). Vorhersagemodell(Unten)	38
Abbildung 4.6:	Funktion zur Gewichtung der Distanz zwischen zwei Bildausschnitten mit c: Steigungskonstante und t: x-Achsen Verschiebung	39
Abbildung 4.7:	Visualisierung der Distanzvektoren anhand von Beispielen bei Punktsymmetrie	40
Abbildung 4.8:	Visualisierung der Distanzvektoren anhand von Beispielen bei Achsensymmetrische	40
Abbildung 5.1:	Übersicht des Experiments mit verschiedenen Modellen	42
Abbildung 5.2:	Erstellungsprozess der Annotationen mit Fraunhofer ISE internem Werkzeuge	43
Abbildung 5.3:	Datenpaar aus Datensatz <i>DS_sup</i>	44
Abbildung 5.4:	Unaliniertes Rohbild (links), welches Verzerrung und Verzeichnung aufweist und aliniertes Bild(rechts)	45
Abbildung 6.1:	Exemplarischer Vergleich aus Experiment U-Net_unwgtLoss mit Ausgangsbild (a), Annotation(b) und Vorhersage(c)	49
Abbildung 6.2:	Trainingsverlauf des U-Net Modells mit ungewichteter Verlustfunktion	50
Abbildung 6.3:	Gegenüberstellung von Ausgangsbild, Annotation und Vorhergesagtem Bild	51
Abbildung 6.4:	Trainingsverlauf des U-Net Modells mit gewichteter Verlustfunktion	51
Abbildung 6.5:	Visualisierung der Aufmerksamkeitskarten von DINO	52
Abbildung 6.6:	Segmentation anhand der Aufmerksamkeitskarten aus dem DINO Modell ohne Pretraining	52
Abbildung 6.7:	Vergleich der berechneten Tensorkorrelationen	53
Abbildung 6.8:	Verlust des Modells	54
Abbildung 6.9:	Genauigkeit	54
Abbildung 6.10:	Veränderung der Vorhersage über Epochen	55
Abbildung 6.11:	Verlust des Modells	55
Abbildung 6.12:	Genauigkeit	56
Abbildung 6.13:	Veränderung der Vorhersage über Epochen ohne Transformationen des Ausgangsbildes	56
Abbildung 6.14:	Verlust des Modells	57
Abbildung 6.15:	Genauigkeit des Modells	57

Abbildung 6.16: Veränderung der Vorhersage über Epochen mittel geometrischer
Verlustfunktion 58

Tabellenverzeichnis

Tabellenverzeichnis

Tabelle 5.1: Überblick des Datensatzes und verwendeten Modelle	43
Tabelle 5.2: Überblick der Klassen mit korrespondierenden Farben in den Annotations	44

1 Einleitung

1.1 Motivation

Ladungsträger-Selektiven Solarzellen sind delikate Komponenten, die hinsichtlich jedes Prozess-Schritts optimiert werden müssen, um einen hinreichenden Wirkungsgrad zu erreichen. Bei der Herstellung von Heterostrukturierten-Solarzellen(HJT), auch bekannt als Silizium-Heterostrukturierte-Solarzellen(SHJ), wird innerhalb mehreren Prozessschritten, amorphe Silizium (a-Si), kristaline Silizium (c-Si), Indium Zinn Oxid(ITO) und transparente leitende Oxid(TCO) Schichten, auf einen Silizium Wafer aufgebracht. Eine genau Abfolge der verschiedenen nacheinander aufgetragenen Schichten ist in Abbildung 1.1 ersichtlich.

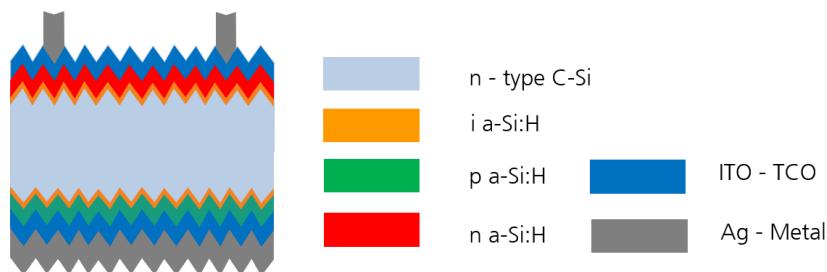


Abbildung 1.1: Schematischer Aufbau eines Wafers als Silicon Heterojunction-Solarzelle

Die Wafer müssen von einem Prozessschritt zum nächsten transportiert werden. Beim Kontakt der Wafer mit den Transportapparaturen können sich Partikel auf der Oberfläche des Wafers ablagern. Dies führt zu einem Einschluss dieses Partikels zwischen zwei Schichten. Durch den generierten Einschluss hat der Wafer an dieser Stelle eine verminderte Ladungsträgerlebensdauer und somit an dieser Stelle einen Defekt.

Der Transport der Wafer zu den einzelnen Produktionsschritten wird über Hilfsvorrichtungen, wie in Abbildung 1.2 dargestellt, realisiert.

- a) Carrier** Bei weiteren Strecken oder zur Zwischenlagerung werden die Wafer in Carrier verwahrt. Diese können mehrere Wafer gleichzeitig halten.
- b) Transportbändern** Innerhalb einer Maschine werden die Wafer mittels Transportbändern bewegt. Damit die Wafer korrekt auf den Transportbändern ausgerichtet sind, werden diese mittels Shifters positioniert.
- c) Greifer** Das Laden zwischen Carriern und Transportbändern wird mittels Vakuum Greifern realisiert. Diese

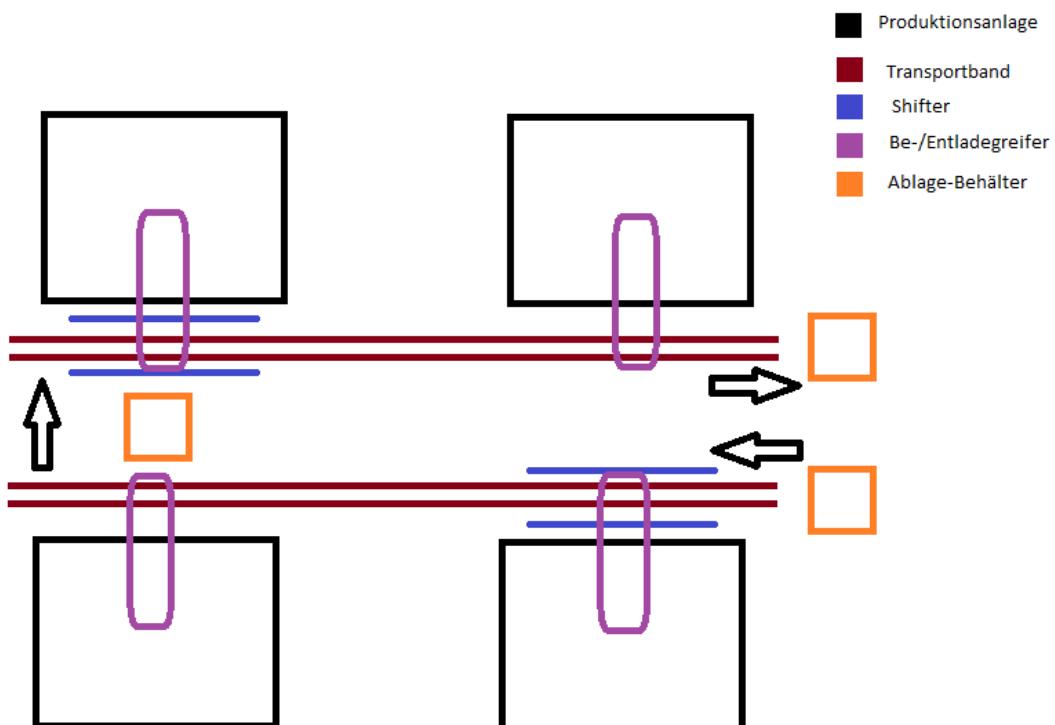


Abbildung 1.2: Darstellung des Handlings innerhalb eines Produktionsteilschrittes

Hier text zu Effizienz von HJT Zellen

In Folge der mechanischen Handhabung, gibt es mikroskopischen Abrieb der Gerätschaften aber auch eine Akkumulation von Verunreinigungen, welche auf den Wafern

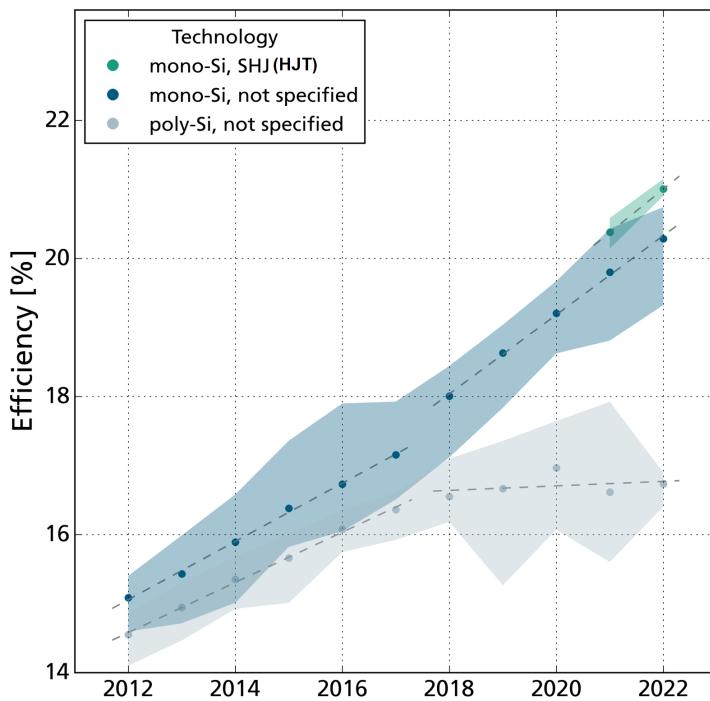


Abbildung 1.3: Entwicklung der Moduleffizienz von 2012 bis 2022; es wird der Median und Interquartile Abstand für mono-Si SHJ(HJT), mono-Si und poly-Si Module gezeigt [hjtrendKrÄdling2022]

zurückbleibt[Fischer.2019] [Fischer.2022] [particle_image_Fischer2019]. Zudem werden bei verschiedenen Prozess Schritten hohe Temperaturen zum Aufbringen der Schicht benötigt. Sowohl der mechanische als auch der Wärmeeinfluss führen beim Auftrag einer Schicht zu einem Defekteinschluss. Dies bedeutet wiederum, dass an dieser Stelle des Zelle eine niedrigere Effizienz erzielt werden kann. In der Schlussfolgerung bedeutet dies, dass diese Solarzelle insgesamt eine verringerte Lebensdauer und somit auch einen verringerten Wirkungsgrad hat.[Fischer.2019b]

Infolgedessen ist eine umfassende Qualitätssicherung unerlässlich, um eine möglichst hohe Qualität der prozessierten Solarzellen sicher zu stellen.

Eine Qualitätssicherung beinhaltet immer die Auswertung einer Datengrundlage.

Maschinelles Lernen kann einen erheblichen Beitrag in der Analyse und Charakterisierung

erzeugen, indem sich anhand von Daten genaue Vorhersagen treffen lassen, um eine aussagekräftige Entscheidung für ein gegebenes Problem zu erzielen.

Für viele Ansätze des Maschinellen Lernens müssen die Daten erst aufwändig bearbeitet werden, bevor sie von einem Netzwerk als gepaarte Daten im Training verwendet werden können. Diese Ansätze werden überwachtes Lernen genannt. Einen überproportional großer Anteil der Kosten der Vorverarbeitung nimmt dabei das Annotieren der Bilder ein, da für Annotationen mit geforderter Qualität ein großes Fachwissen benötigt wird.

Um diesen Teil der Vorverarbeitung der Daten zu eliminieren, gibt es eine Klasse an Methoden, die sich unter dem Begriff des un-überwachten Lernens zusammenfassen lassen. Diese sollen zum einen eine Reduktion der Notwendigkeit von Experten/Expertinnen im Annotationsprozess und gleichzeitig eine robustere Anwendung auf neue Datensätze ermöglichen. Dies bedeutet somit, dass die Kosten der Daten-Vorbereitung sinken.

Ob ein Datensatz gut ist lässt sich anhand von drei Faktoren ermitteln. Der Qualität , der Quantität und der Variabilität des Datensatzes. Da diese Faktoren bei der Erstellung eines Datensatzes genau berücksichtigt werden müssen, ist jede Verringerung der Komplexität ein Zugewinn. Zudem entscheidet die Güte der Daten auch direkt über die benötigte Zeit, welche ein Modell benötigt um zu konvergieren.

1.2 Beitrag dieser Arbeit

In dieser Thesis werden unüberwachte Lernmethoden verwendet, um Defektstrukturen auf Solarzellen, anhand von photolumineszens-Messungen, zu segmentieren. Die Anwendung von unüberwachten Lernmethoden zur Segmentierung von Defekten in Photolumineszenz-Messungen wurde bisher noch kaum getestet. Um die Ziele dieser Arbeit zu erreichen, wird ein Datensatz aufbereitet, überwachte und unüberwachte Lernmodelle implementiert, ein unüberwachtes Modell erweitert und eine Analyse der verschiedenen Ergebnisse hinsichtlich ihrer Qualität durchgeführt. Diese Arbeit soll die Grenzen der verwendeten überwachten und unüberwachten Methodiken identifizieren und gegenüberstellen.

Datensatzerstellung

Es wurden drei Datensätze erstellt und aufbereitet. Der Datensatz DS_unsup wurde aus verschiedenen Messdaten, welche einen Querschnitt von Industrieproben darstellen, zusammengestellt. Für den Datensatz DS_sup wurde eine repräsentative Auswahl von Daten getroffen, welche mit Klassenzuordnungen auf Pixelebene annotiert wurden. Dieser ermöglicht es, Modelle mit einem überwachten Lernansatz zu trainieren und die Analyse der verschiedenen Lernmodelle quantitativ zu bewerten. Zur weiteren Aufbereitung wurden die Daten, als Teil der Augmentation wie in Kapitel ?? beschrieben, aliniert.

Anwendung von verschiedenen überwachten und unüberwachten Lernmodellen

Es wurden vier verschiedene Modelle implementiert und trainiert. Davon sind zwei dieser Modelle mit einem überwachten Lernprozess und zwei mittels einer unüberwachten Methode trainiert.

1. Für einen Referenzpunkt wurde ein U-Net [**UNETRonneberger2015**] für sieben Klassen implementiert. Es wurden verschiedene Gewichtungen der Klassen, Optimierungsfunktionen sowie Verlustfunktionen analysiert. Das Modell wird in Kapitel ?? beschrieben.
2. Für einen zweiten Referenzpunkt als “zero-shot” Transfer Ansatz wurde Segment Anything [**SAMKirillov2023**] implementiert. Dieser Ansatz wurde mit dem erstellten Datensatz trainiert, um eine Aussage über die “zero-shot” Qualität des Diskriminators treffen zu können. Die Modellparameter wurden für eine breitere Vergleichbarkeit angepasst. Das Modell wird in Kapitel ?? beschrieben.
3. Als unüberwachten Lernmodell wird auf der Grundlage eines Vision Transformer [**ViTdosovitskiy2021image**] das Verfahren von Caron et. al. [**caron2021emerging**] als Self-Supervised Ansatz angewendet. Das verwendete Modell wird in Kapitel ?? beschrieben. Das Modell wird mit variierenden Parametersätzen trainiert.

4. Als weiteres unüberwachtes Lernmodell wird das von Hamilton et. al. verwendete Verfahren [**STEGOhamilton2022unsupervised**] implementiert und trainiert. Das Modell wird in Kapitel ?? beschrieben. Dieses Modell wird zusätzlich verwendet um eine auf die Daten angepasste Verlustfunktion zu trainieren. Die Parameter wurden gezielt variiert, um eine bestmögliche Anpassung auf den Datensatz DS_unsup zu erreichen. Die Dokumentation der entwickelten Verlustfunktion befindet sich in Kapitel ??

Analyse der Ergebnisse

Die Ergebnisse werden hinsichtlich der Eignung für die entwickelten Datensätze, der Fehleranalyse, Relevanz und Einfluss der einzelnen Modellparameter, im Vergleich zu den implementierten Modellen und auch im Vergleich zur Literatur analysiert. Mit den im Datensatz DS _test vorhandenen Annotationen, lassen sich die Ergebnisse qualitativ und quantitativ vergleichen. Die Ergebnisse und eine detaillierte systematische Modellanalyse werden in Kapitel 6 vorgestellt und in Kapitel ?? eingehend diskutiert.

2 Grundlagen

Zu Beginn soll hier für die technologischen Grundlagen ein Verständnis vermittelt werden. Zudem werden alle für diese Arbeit wichtige wissenschaftliche Arbeiten vorgestellt. Zunächst wird ein Überblick über die Grundlagen von Netzwerken gegeben. Im Anschluss werden dann verschiedene Metriken erläutert, mit welchen die semantische Segmentierung bewertet wird. Zudem werden die verwendeten Datensätze beschrieben und ihre Herkunft beschrieben.

2.1 Struktur der HJT Zelle

Eine Silizium Heterojunction-Solarzelle besteht aus einem Stapel von kristallinen und amorphen Silizium Schichten, wobei durch die Nutzung der unterschiedlichen Absorptionsfenster der beiden Silizium-Materialien ein höherer Wirkungsgrad erreicht wird als bei Mono-Junction Solarzellen. Die mittlere Schicht, bestehend aus dem negativ dotierten (n) kristallinen Silicium-Wafer ist dabei zunächst von einer Schicht intrinsischem (i) amorphen Silicium ummantelt, welcher die eigentliche "Junction", also den Übergang, bildet. Durch Abscheidung einer positiv dotierten (p) a-Si Schicht auf der Rückseite erfolgt die Emitterbildung, während auf der Vorderseite eine n-dotierte a-Si-Schicht aufgebracht wird. Somit ergibt sich eine p-i-n-i-n-Schichtung mit pn-Übergang auf der Rückseite. Für eine gleichzeitige Reduktion der Reflexion und der Erhöhung der lateralen Leitfähigkeit werden schließlich die Transparent Conductive Oxide(TCO) und Indium Tin Oxit(ITO) [SHJDeWolfDescoedresHolmanBallif2012] Schichten aufgebracht. Die Herstellung der Metallkontakte, auch Metallisierung genannt, wird nun als letztes aufgetragen, welche für die Leitfähigkeit aus der Solarzelle sorgt. Eine Beispielhafte Konfiguration ist in Abbildung 1.1 zu sehen.

Während des Herstellungsprozesses werden Defekte in das Material eingebracht, insbesondere durch das Aufbringen der a-Si, ITO- und TCO-Schichten und die damit verbundenen Transportvorgänge. Dafür müssen die Wafer zu den einzelnen Apparaturen, welche das Schichtauftragsverfahren durchführen, transportiert werden. Der Transport wird durch Förderbänder, Greifer und Carrier realisiert. Bei größeren Distanzen werden die Wafer mittels Greifern in ein Carrier gelagert und bei der nächsten Anlage mit weiteren Greifern aus selbigen auf ein Förderband gehoben.



Abbildung 2.1: Blick in die Produktion mit Transportvorrichtungen wie Greifern und Förderbändern[tray_img23]

Durch die Handhabung der Wafer werden Partikel auf die Waferoberfläche aufgebracht, welche in optischen Messsystemen beobachtet werden können und kritisch für die Solarzelle sind. Bei dem Kontakt mit den Fördereinrichtungen kommt es zu einem mechanischen Abrieb, welcher in den Wafer diffundiert. Dies wird durch die Temperatur von bis zu 145°C, welche für den Schichtauftragsprozess benötigt wird, verstärkt. Ein Raster-Elektronen-Mikroskopbild der, auf die Waferoberfläche übertragenen Partikel nach einem Greiferkontakt ist in Abbildung 2.2 zu sehen. In Rot sind dabei die Partikel gekennzeichnet, welche sich nach dem Kontakt auf dem Wafer befinden. WICHTIG Die Teile was häufig zu einem Einschluss zwischen den Schichten führt.

Zusätzlich kann es bei den einzelnen Transportvorgängen zu einer mechanischen Verformung

des Wafermaterials kommen. Dies geschieht beispielsweise beim Entladevorgang, da die Vakuum-Greifer mit einem hohen Unterdruck angesteuert werden müssen um die Wafer aus dem Carrier lösen zu können.

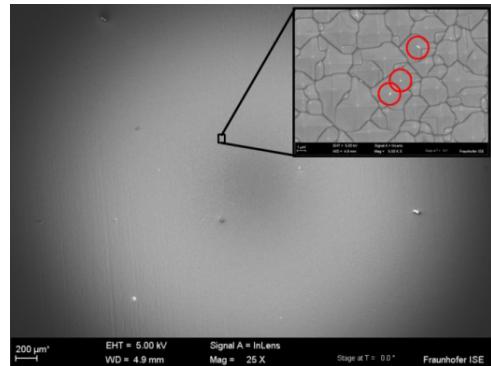


Abbildung 2.2: eines Vakuum Greifers mit resultierenden Partikeln. Aufgenommen mit einem Rasterelektronenmikroskop [particle_image_Fischer2019]

Einschlüsse aus Fremdmaterialien zwischen den aufgetragenen Schichten und mechanische Verformungen wirken sich negativ auf die gesamt Effizienz der Solarzelle aus. Fischer et. al [Fischer.2022] zeigen, dass Materialeinschlüsse zu einer geringeren Effizienz führen. Die Korrelation zwischen der prozentualen Partikelabdeckung und dem relativen Effizienzverlust ist in Abbildung 2.3 zu sehen. Dies ist für verschiedene Greifertypen bei verschiedenen Materialtemperaturen dargestellt.

2.2 Datengrundlage

Um ein Netzwerk trainieren und evaluieren zu können, werden Daten benötigt. Die Auswahl eines repräsentativen Datensatzes ist die Grundlage für den Erfolg des Trainings und der Qualität der in der Auswertung erzielten Ergebnissen.

Daten für Segmentierungsaufgaben

Für die Segmentierung eines Bildes muss die Qualität, Quantität und Variabilität der Daten in einer Güte vorliegen, dass die Möglichkeit besteht, ein ausreichend gutes Ergebnis zu erzielen.

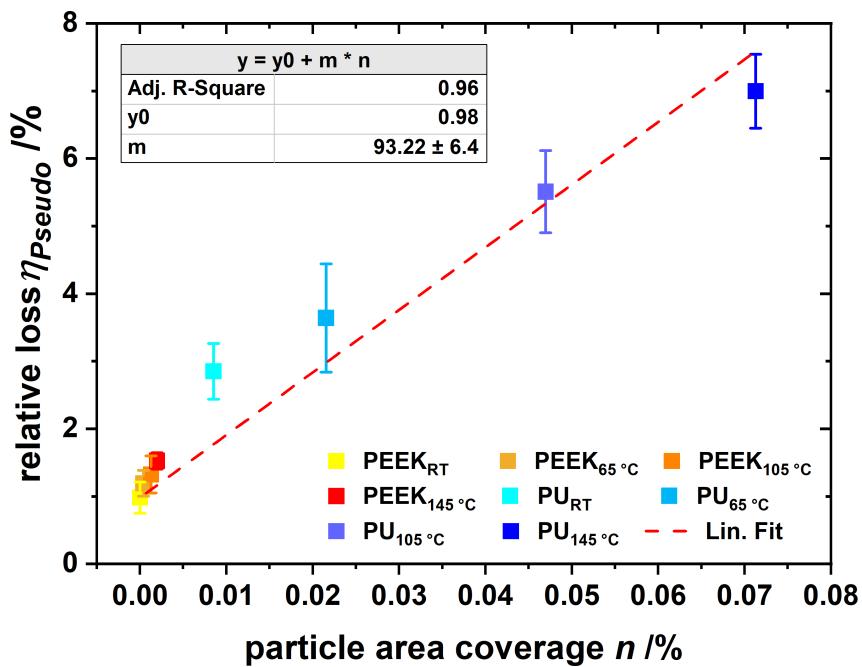


Abbildung 2.3: Korrelation des normalisierten Pseudoverlustes η durch eine bei der Handhabung induzierten Defekt. Diese wird für verschiedene Temperatur und Material abhängige Teilchendichte angegeben [impactParticlesFischerrefId0]
PU = Polyurethan , PEEK = Polyetheretherketon

Dabei ist ein Faktor für die hinreichende Repräsentativität, dass genügend Daten vorhanden sind. Zusätzlich sollte möglichst auf eine Gleichverteilung der einzelnen Klassen geachtet werden, damit keine Über- oder Unter-Repräsentation von einzelnen Klassen stattfindet. Dies ermöglicht dem trainierenden Netz eine ausreichende Varietät einer Klasse zu erkennen und somit die relevanten Eigenschaften dieser Klasse filtern.

Damit eine Vorhersage getroffen werden kann, müssen die Daten als Bilddaten vorliegen. Für überwachte Trainingsalgorithmen ist es zudem essentiell, dass diese als Bildpaare vorliegen. Das Datenpaar besteht dabei aus dem eigentlichen Bild und einer Ground Truth (GT). Das meist mit großem Aufwand händisch erzeugte Grundwissen teilt das Bild in entsprechende Klassen auf, so dass jeder Pixel einer Klasse angehört.

2.3 Messverfahren

Bildgebende Photolumineszenz

Die Bildgebende Photolumineszenz(PL) macht intrinsische, aber auch defekt-bedingte elektronische Übergänge in einem zu untersuchenden Material sichtbar. Ein großer Vorteil dieser Technik ist, dass sie nicht invasiv ist und somit keinen direkten physischen Kontakt mit dem zu untersuchenden Wafer benötigt. Dies eliminiert Artefakte, die durch einen Kontakt mit der Probe entstehen könnten.

Die lokale elektronische Qualität eines Silizium-Wafers wird in der bildgebenden Photolumineszenz durch einen Detektion von ausgesandtem Licht in der Bandlücke charakterisiert.[**PL_imaging**] Dazu werden die Elektronen in einem Halbleiter, mittels Lichtabsorption in einen elektronisch angeregten Zustand gebracht. Die Entstandenen Elektron-Elektron Loch-Paare rekombinieren nach ihrer Lebensdauer. Die Energie, welche bei der strahlenden Rekombination frei wird, kann somit gemessen werden.

Der Exemplarische Aufbau einer PL Messung ist in Abbildung 2.4 dargestellt. Der Wafer wird durch ein Laser angeregt. Die freiwerdende Energie wird von deiner ladungsgekoppelten Kamera erfasst. In den entstandenen Photolumineszenz Bildern wird somit die elektrische Qualität der untersuchten Probe dargestellt. Dieser wird in Grauwerten angegeben. Ein niedriger Grauwert spiegelt dabei eine niedrigere elektrische Leitfähigkeit und umgekehrt ein höherer Grauwert eine größere elektrische Leitfähigkeit dar.

2.4 Bildsegmentierung

2.4.1 Traditionelle Bildverarbeitung

In der Bildverarbeitung sind automatisierte visuelle Erkennungsaufgaben für Anwendungsgebiete wie Bildklassifikation, Objekterkennung und Bildsegmentierung ein wichtiger Bestandteil. Traditionelle Verfahren zur Merkmalserkennung verwenden Heuristiken,

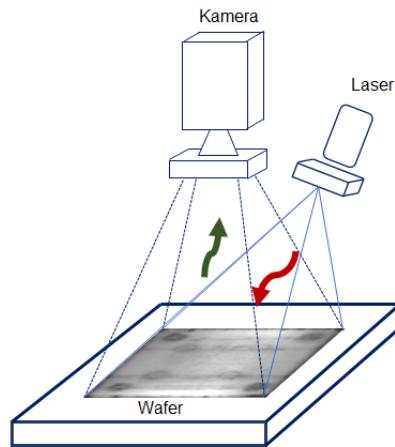


Abbildung 2.4: Exemplarische Darstellung einer PL Messung

welche speziell angepasste Parameter benötigen, um Eigenschaften aus Bildern für eine entsprechende Schlussfolgerung zu extrahieren. Oftmals sind diese Verfahren sehr effektiv, allerdings sind sie häufig nicht robust auf eine Veränderung der Daten und Parameter eingestellt. Dies macht es manchmal schwer für ein entsprechendes Aufgabengebiet einzustellen. Dies ist auch der Grund, warum diese Art von Merkmalserkennung schwer verallgemeinerbar und übertragbar ist.

Thresholding. Thresholding ist ein Verfahren, welches sehr effektiv und simpel für eine Aufteilung eines Bildes in verschiedene Klassen sein kann. Hierbei wird mittels eines Schwellwertes bestimmt, zu welcher Klasse die betrachtete Stelle in einem Bild gehört. Das Thresholding hat viele verschiedene Varianten wie das Durchschnittsverfahren, ein Adaptives Verfahren oder das von Otsu[otsu4310076] entwickelte. Diese Verfahren haben allerdings immer das Problem, dass diese an Helligkeitswerte gekoppelt sind und auch schwer auf andere Datenquellen übertragbar sind. Thresholding wird allerdings oftmals als letzter Schritt in einer Kette von Prozess-Schritten ausgeführt, was diese Nachteile ausgleichen kann.

Ein Thresholding kann so beispielsweise innerhalb der Prozesskette von Transformern (vgl. Kapitel 2.5.3) zur Entwicklung von verschiedenen Klassen aus einzelnen Attention-Heads helfen.

Canny Kantendetektor. Eine weit verbreitete Techniken der Bildsegmentierung ist die Canny-Kantenerkennung [Canny1986]. Dieser Algorithmus basiert auf einer Faltung mit dem Sobel-Filter, Non-maxima-suppression und einem Schwellwertverfahren.

2.4.2 Maschinelles Lernen

Neben den klassischen Verfahren zur Bildsegmentierung werden zur Segmentierung von Bilddaten auch maschinelles Lernen eingesetzt. Da in dieser Arbeit maschinelles Lernen in Form von überwachtem und unüberwachten Lernmethoden zur Detektion von Defekten eingesetzt wird, wird hier ein kurzer Überblick über die einzelnen Verfahren gegeben.

2.5 Neuronalen Netzes

Neuronale Netze ermöglichen es, die Komplexität von Daten in einen kleinen Datenraum zu überführen und in diesem eine komprimierte Darstellung der wichtigen Eigenschaften zu erhalten. Diese Repräsentations-Methoden ermöglichen es somit dem Modell, verschiedene Repräsentationsebenen für eine Eingabe zu lernen [deepLearningLeCun2015]. Der Unterschied zu traditionellen Methoden ist, dass die Eigenschaften der Daten somit nicht individuell betrachtet wurden, sonder mit einer allgemein anwendbarer Methode gelernt wurde.

Neuronale Netze haben durch die erzielten Ergebnisse bewiesen, dass diese für viele Bereiche geeigneter sind. Die Bildverarbeitung ist ein Bereich, in welchem sich enorme Fortschritte durch die Einführung von tiefen neuronalen Netzen ergeben haben. Dies erstreckt sich über beispielsweise die Entwicklung von Convolutional Neural Networks (CNNs) [CNNsCunHBB99], die Erstellung von Datensätzen wie SA-1B [SAMKirillov2023]

mit 11 Millionen hochauflösten Bildern und 1,1 Milliarden qualitativ hohen Annotationen auf Pixelebene, oder neuen Lernstrategien, welche Labels überflüssig machen (vgl. 2.5.2). Dies spricht sehr stark dafür, dass tiefe Neuronale Netze für viele Bildverarbeitungsprobleme essenziell sind.

Da diese Arbeit besonders den Unterschied zwischen überwachten und unüberwachten Lernmodellen analysiert, wird im Folgenden besonders auf verschiedene Lernstrategien im Bezug auf Bildsegmentierung eingegangen.

2.5.1 Überwachtes Lernen

Überwachtes Lernen repräsentiert einen fundamentalen Ansatz des maschinellen Lernens und für das Lösen von Bildverarbeitungsproblemen. Bei diesem Lernansatz wird ein Modell auf Basis von annotierten Trainingsdaten entwickelt. Ein gelabelte Datenpaar besteht aus Grunddaten, welche das zu lösende Problem darstellt und zugeordneten Lösungen des Problems. Für ein Segmentationproblem wäre dies beispielsweise ein Bild und ein Label, welches jedes Pixel des Bildes einer Zielklasse zuordnet.

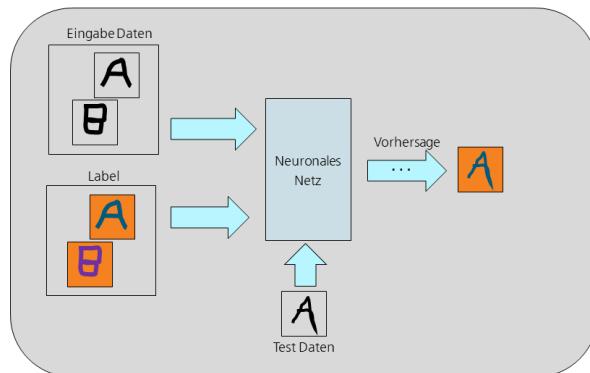


Abbildung 2.5: Darstellung einer überwachten Lernstrategie. Daten-Label Paar trainiert Neuronales Netz. Testdaten können mit einem erfolgreich trainierten Netz in korrekt gelernte Klassen überführt werden.

Das primäre Ziel des überwachten Lernens besteht darin, ein Modell zu entwickeln, welches die komplexen Zusammenhänge und Muster zwischen den Eingaben und den zugehörigen

Ausgaben erkennt. Dieser Prozess umfasst in der Regel die Minimierung einer Fehlerfunktion, die die Abweichung zwischen den vom Modell vorhergesagten Ausgaben und den tatsächlichen Ausgaben quantifiziert. Die Minimierung der Fehlerfunktion wird durch Anpassung von Modellparametern erreicht. Wenn sehr klare Informationen für die Ausgabedaten vorhanden sind, ist diese Art von Lerntyp sehr gut anwendbar. Dies bedeutet somit, dass für Aufgabengebiete, in denen Annotationen für Bilder nur schwer erlangbar sind, das überwachte Lernen keine geeignete Architektur darstellt.

Populäre Modelle innerhalb dieser Strategie sind CNNs [**CNNsCunHBB99**], wie beispielsweise das Unet [**UNETRonneberger2015**], Generative Adversarial Networks (GAN) [**GAN_GoodfellowNIPS2014_5ca3e9b1**] wie pix2pix [**GANisola2018imagetoimage**] oder Transformer [**Transformer_Vaswani2017**] wie Segment Anything [**SAMKirillov2023**].

2.5.2 Unüberwachtes Lernen

Unüberwachtes Lernen konzentriert sich, im Gegensatz zum überwachten Lernen, auf das Erkennen von Strukturen und Mustern in Daten, ohne dass annotierte Daten vorhanden sind. Es gibt zwei zentrale Herausforderungen. Diese umfassen die Dimensionsreduktion der Daten um eine bessere Repräsentation der Merkmale entwickeln und erfassen zu können und das Problem des Clusterings, in welchem ähnliche Datenpunkte in Gruppen oder Cluster zusammengefasst werden.

Der Vorteil dieser Architektur liegt darin, dass keine annotierten Daten für den Trainingsprozess benötigt werden. Dies ermöglicht es auch für hochspezialisierte Gebiete, in welchen nur wenige oder keine Annotationen vorhanden sind, maschinelles Lernen einzusetzen, ohne einen immensen Aufwand durch das Annotieren von Daten zu generieren. Unüberwachte Lernstrategien werden in verschiedensten Modellen zur Bildsegmentierung eingesetzt. Relevante Repräsentanten sind IIC [**iicji2019**], DINO [**caron2021emerging**] oder STEGO [**STEGOhamilton2022unsupervised**].

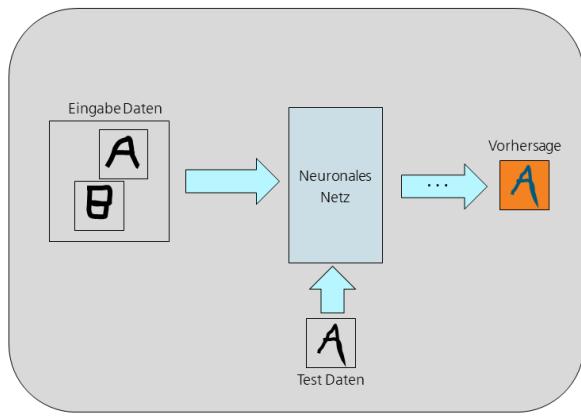


Abbildung 2.6: Beispielhafte Darstellung eines überwachten Lernverfahrens. Neuronales Netz trainiert lediglich mit Daten ohne . Testdaten können mit einem erfolgreich trainierten Netz in korrekt gelernte Klassen überführt werden.

2.5.3 Vision Transformer

Die Transformer Architektur [Transformer_Vaswani2017] ist der Standard bei Problemstellungen in der Verarbeitung von natürlicher Sprache. Diese Architektur benötigt zum Vortrainieren eine große Menge an Daten. Der Vorteil dieser Modells liegt darin, dass für das Anpassen an ein spezifisches Problem nur eine kleine Datenmenge benötigt wird. Die Berechnungseffizienz und Skalierbarkeit von Transformern ist ausgezeichnet, was es ermöglicht, selbst Modell mit weit mehr als 100 Milliarden Parametern zu trainieren. Da die Architektur der Transformer noch nicht so umfangreich, wie beispielsweise Convolutional Neural Networks, erprobt wurden, muss noch gezeigt werden, dass dieser Ansatz in Robustheit und Qualität für reale Anwendungen eingesetzt werden. Allerdings zeigt sich vermehrt, dass Transformer Architekturen Stand der Technik Resultate erzeugen können[stateOfTheArt_Mo_2022].

Dosovitskiy et. al. [ViTdosovitskiy2021image] haben den Transformer-Ansatz auf Bilddaten angewendet. Die Architektur wurde mit minimalen Änderungen von der Verarbeitung natürlicher Sprache auf die Verarbeitung von Bildern übertragen. Dieser Ansatz wird schematisch in Abbildung 2.7 dargestellt.

Die Grundidee dieser Arbeit besteht darin, ein Bild in Ausschnitte zu zerlegen und diese

als Abfolge von Objekten zu interpretieren. Hierzu werden aus den einzelnen Ausschnitten mithilfe von Linearprojektionen Einbettungen erstellt. Vor die Sequenz der Bildausschnitte wird zusätzlich eine lernfähige Einbettung gesetzt, die sicherstellt, dass nur die letzte Einbettung in die Klassifikationsschicht einfließt. Die Einbettung kann als aggregierte Darstellung der Bildausschnitte interpretiert werden. Alle Ausschnitte werden mit einer Positionseinbettung versehen, damit die Positionsinformation erhalten bleiben. Anschließend wird diese Sequenz in den Kodierer des Transformers überführt, welcher aus mehreren gleich aufgebauten Schichten besteht. In jeder Schicht durchlaufen die Einbettungen einen Multihead Self-Attention Block und werden durch einen mehrlagigen Perzepton-Block interpretiert. Vor jedem Block wird die Eingabe normalisiert und nach jedem Block wird eine Residualverbindung erzeugt.

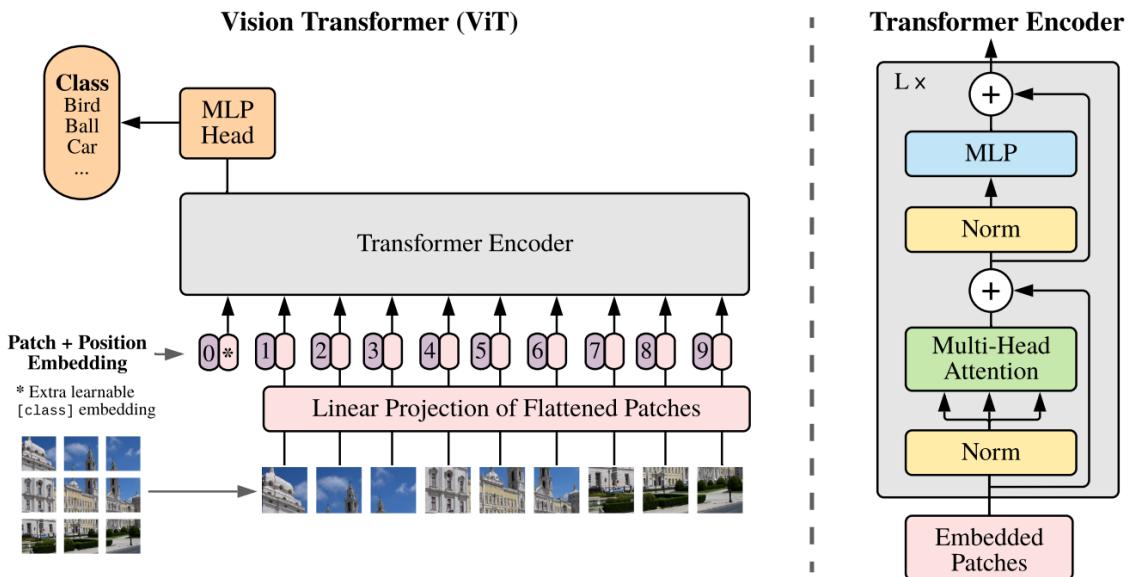


Abbildung 2.7: Exemplarische Darstellung eines Vision Transformers [ViTdosovitskiy2021image]

Um die Fokussierung auf unterschiedliche Räume zu ermöglichen, die wiederum unterschiedliche semantische oder syntaktische Bedeutungen haben können, wird der Multi-Head Attention Block verwendet. Der Attention Block vergleicht alle Eingabeelemente miteinander und verändert das entsprechende Ausgabeelement. Dies bedeutet, dass die

Eingabesequenz speziell für jeden Eintrag nach dem entsprechenden key-value durchsucht und dann der Ausgabesequenz zugeordnet wird.

Die verschiedenen Versionen des Vision Transformers, welches sich beispielsweise durch eine andere Ausschnittsgröße und somit auch Parameterraum unterscheidet, erzielt bessere Ergebnisse als die Modelle, welche davor Stand der Technik waren [**ViTdosovitskiy2021image**]. Es wird gezeigt, dass auf populären Bild Datensätzen mit dieser Technik eine sehr hohe Genauigkeit erzielt wird. Beispielsweise wird auf dem Cityscapes [**CityscapesCordts2016**] Datensatz eine durchschnittlich Intersection over Union von 84,5 % als multi-scale Inferenz bei der Verwendung der Mask2Former [**Mask2Former_Cheng_2022_CVPR**] Strategie.

2.6 Verlustfunktionen

Für das Training eines Modells ist eine Verlustfunktion notwendig. Im Bildsegmentierungs Kontext gibt dieser den Unterschied zwischen den vorausgesagten Daten und den tatsächlichen Zieldaten an. Repräsentanten von Lossfunktionen sind Mean Absolut Error(MAE), Mean Squared Error(MSE) und Binary Entropy (BCE).

2.6.1 Tensorkorrelation

2.6.2 Binary Cross Entropy

Dieser errechnet den Unterschied der Wahrscheinlichkeiten zwischen vorhergesagtem und realen Bild für alle vorhanden Klassen. Dies Art von Verlustfunktion kann sehr gut in Überwachten Lernverfahren eingesetzt werden, da direkt eine Übereinstimmung zwischen den errechneten Ergebnissen und den tatsächlichen Ergebnissen berechnet werden kann. Trotzdem kann diese Funktion auch in unüberwachten Szenarien eingesetzt werden. So wird beispielsweise in DINO [**caron2021emerging**] die Distanz zwischen den Parametern

der beiden Netzwerke errechnet. Eine genauere Beschreibung von DINO findet sich in Kapitel ??.

Die BCE Verlustfunktion berechnet sich wie in Funktion 2.1 dargestellt, mit den durchschnittlichen Zielklassen y_i und den vorhergesagten Daten \hat{y}_i und einer Anzahl an Daten n .

$$L_{BCE} = - \sum_{i=1}^n y_i \cdot \log(\hat{y}_i) + (1 - y_i) \cdot \log(1 - \hat{y}_i) \quad (2.1)$$

Die Verlustfunktion selbst kann auch als Metrik verwendet werden. Über die Größe der Änderung des Verlustes kann gedeutet werden, ob das verwendete Modell konvergiert oder nicht. Dies ist in jedem Fall allerdings nur ein hinreichendes Evaluationsmaß und kein ausreichendes. Somit macht es Sinn, den Verlust in die Bewertung des verwendeten Modells einfließen zu lassen, aber nicht als einziges Merkmal zu benutzen.

Wenn eine Verlustfunktion konvergiert ist dies ein Indikator, dass das Training des Modells hinsichtlich der verwendeten Parameter ein Minimiert wurde. Diese Minimum muss allerdings interpretiert werden, da eine Minimum auch immer eine Triviale Lösung darstellen kann.

2.7 Metriken

Damit ein Ergebnis evaluiert werden kann, muss ein Maß entwickelt werden, welches verschiedene Punkte in gegebenen Daten miteinander vergleichen kann. Metriken lassen sich in analytische und empirische Bewertungsmethoden unterteilen. [metricsPOLAK2009] Während die analytischen Methoden die Effizienz, Komplexität und die Strategie im Vorgehen bewerten, werten die empirischen Methoden das Ergebnis selbst aus. Die Empirische Auswertung lässt sich weiter in empirische Güte und Diskrepanz unterteilen. In der Segmentierung von Bilddaten bedeutet dies, dass ein Bild mit der Segmentierten Version dessen verglichen wird. Die Güte misst dabei Faktoren wie die Einheitlichkeit von segmentierten Flächen, den Kontrast zwischen den einzelnen Regionen oder die Form

des Segments. Die Diskrepanz hingegen bewertet den konkreten Unterschied zwischen einer Segmentierung und ihrem oftmals händisch annotierten Referenzbild. Eigenschaften dafür sind Menge und Position an falsch segmentierten Pixeln, die Anzahl von Objekten in einem Bild und geometrische Merkmale wie die Fläche, der Umfang oder die Form des segmentierten Objekts.

Jaccard-Koeffizient

Der Jaccard-Koeffizient, welcher auch Intersection-over-Union (IoU) genannt wird, ist eine wichtige Metrik im Auswertungsprozess. Dieser ist als Schnittmenge zwischen segmentiertem Bereich X und der Ground Truth Y definiert. Berechnet wird dieser durch:

$$IoU = \frac{(X \cap Y)}{(X \cup Y)} \quad (2.2)$$

Der berechnete Wert liegt somit zwischen Null und Eins. Eine Erweiterung des Jaccard-Koeffizienten ist die durchschnittliche Intersection-over-Union (mIoU) welches beim Betrachten von mehreren Segmenten einer Klasse, der durchschnittliche IoU Wert dieser ist. Dieser Wert gibt eine gute Übersicht, ob sich ein verwendetes Modell eine gute Vorhersage erzeugt hat oder nicht. Die IoU beziehungsweise mIoU ist der Standard, welcher als Vergleichsmaß zwischen Modellen genutzt wird. **nur IoU da IoU und Genauigkeit die gleiche Berechnung mit verschiedener Skalierung ist**

Genauigkeit, Präzision, Sensitivität und F1-Maß

Ein weiteres oft genutztes Maß in der empirischen Diskrepanzauswertung von Segmentierungsergebnissen ist die Genauigkeit (Accuracy). Diese beschreibt die Anzahl an richtig

Segmentierten Testpunkten aus der Menge der Gesamtstellen. Dies lässt sich über:

$$\text{Genauigkeit} = \frac{TP}{TP + FP} \quad (2.3)$$

TP(Echt Positiv stellt die tatsächlich richtig klassierten Stellen dar, während *FP(Falsch Positiv)* die falsch segmentierten Elemente der entsprechenden Klasse symbolisiert. Die Präzision(Precision) ist hingegen durch

$$\text{Präzision} = \frac{TP}{TP + FN} \quad (2.4)$$

definiert. Hier stellt *FN(Falsch Negativ* die Menge an Punkten dar, welche zu einer entsprechenden Klasse gehören, allerdings nicht segmentiert wurden. Ein in der Literatur stark genutzte Metrik ist die Sensitivität(Recall). Dieses Maß stellt die tatsächlich korrekt klassierte Menge an Datenpunkten dar. Sie lässt sich durch

$$\text{Sensitivität} = \frac{TP}{TP + FN} \quad (2.5)$$

darstellen

Als vierter in dieser Reihe der Metriken ist das F1-Maß zu nennen. Dieses kombiniert die Genauigkeit mit der Präzision und lässt sich als

$$F1 = \frac{2 * \text{Präzision} * \text{Genauigkeit}}{\text{Präzision} + \text{Genauigkeit}} \quad (2.6)$$

darstellen. Mittels dieser Gleichung lassen sich sehr einfach die Faktoren für Präzision und Genauigkeit zusammenfassen.

Metriken für Unüberwachtes Lernen

Da in den vergangenen Jahren versucht wird, vermehrt Unüberwachtem Lernenverfahren zum Einsatz kommen, müssen Metriken gefunden werden, welche für diese Modelle geeignet sind. Da das Unüberwachten Lernen (vgl. Kapitel 2.5.2) keine Annotationen zur Verfügung hat, macht das die beschriebenen Metriken wie der Jaccard-Koeffizient oder das F-Maß nicht direkt anwendbar. Es ist schwer, ein Maß zu definieren, welches ein Segmentation bewerten soll, ohne dass ein Wahrheitsbild von der entsprechenden Klasse besteht.

Die Faktoren ob eine Segmentation gut geclustert wurde, können durch Skalierungsinvarianz, Fülle und Konsistenz eines Clusters [evaluationMetricsUnsupervisedLearningAlgorithmsP] bewertet werden. Dieses Axiom wurde von Kleinberg [clusterAxiomKleinberg2002] erstellt. Es besagt, dass diese Eigenschaften ein gutes Segmentationsergebnis ausmachen, es allerdings unmöglich ist, alle drei Eigenschaften gleichzeitig zu erfüllen.

Skalierungsinvarianz bedeutet in diesem Fall, dass ein Algorithmus, welcher entsprechende Segmentierungen clustert, das Ergebnis nicht verändert, wenn alle Abstände zwischen diesen mit einem konstanten Faktor skaliert werden. Die Fülle einer Segmentation ist gegeben, wenn der Algorithmus eine beliebige Zusammenstellung der segmentierten Ergebnisse eines Datensatzes erzeugen kann. Ein Clustering Verfahren für Segmenationen ist konsistent, wenn die berechneten Ergebnisse sich nicht ändern, obwohl sich die Abstände der Punkte innerhalb einer Segmentationsklasse verkeinen oder die Abstände zwischen einzelnen Clustern größer wird.

Eine oftmals genutzte Möglichkeit, diese Eigenschaften zu beschreiben, ist die Berechnung der Kohäsion einer geclusterten Klasse und die Trennung von einzelnen Segmentationen.

$$\text{Kohäsion}(C_i) = \sum_{x \in C_i, y \in C_i} \text{Nachbarschaft}(x, y)$$

C stellt das entsprechende Cluster und x beziehungsweise y jeweils ein Beispiel in dem

Cluster dar.

Damit lässt sich auch die Trennung zwischen einzelnen Clustern darstellen.

$$\text{Trennung}(C_i, C_j) = \sum_{x \in C_i, y \in C_j} \text{Nachbarschaft}(x, y)$$

Sowohl die Kohäsion als auch die Trennung basieren auf einer Approximation der Ähnlichkeit von zwei Daten Paaren. Diese bedeutet somit, dass ein gutes Clustering eine hohe Trennung zwischen Clustern und eine hohe Kohäsion innerhalb der Klasse hat.

Da diese Metriken komplex sind und nicht jedes mal separat berechnet werden müssen, wurden verschiedene Metriken entwickelt, welche Kohäsion und Trennung zusammenfassen.

Ungarische Zuordnungsmethode

Da bei Unüberwachten Lernmodellen aus den Vorausgesagten Bilddaten nicht ersichtlich ist, welche Klasse zu welcher Zielklasse gehört gibt es Verfahren, welche dies Approximieren. Ein Verfahren was in vorausgehenden Arbeiten und der Literatur verwendet wird, ist die Ungarische Zuordnungsmethode.[**hungarian_Kuhn_1955**] Diese Methode misst, wie konsistent sich die vorausgesagten semantischen Segmente in den tatsächlichen Annotations Widerspiegeln und wie invariant diese gegen Permutation der Klassen ist.

Die Zuordnung jedes Pixels der Vorhersage zu einer Zielklasse kann als $m \times m$ Matrix X , mit $m \times m$ Kostenmatrix C dargestellt werden kann. Das Ziel der Ungarischen Zuordnungsmethode ist nun eine optimale Zuordnung zu finden, welche die Gesamtkosten minimiert. Die Minimierung kann durch die Formel 2.7, mit Pixeln i in der Vorhersage und Zielklassen j dargestellt werden.

$$\min(C) = \sum_{i=1}^m \sum_{j=1}^m C_{i,j} X_{i,j} \quad (2.7)$$

Die Minimierung der Kosten wird mit der ungarischen Methode wie von Kuhn et al. [hungarian_Kuhn_1955] beschrieben durchgeführt. Diese umfasst eine Zeilen und Spaltenweise Reduktion um dem kleinste Zeilen beziehungsweise Spalten Element der Kostenmatrix. Bei einer noch nicht optimalen Zuordnung wird das Kleinste Element der Kostenmatrix gefunden. Dieser Wert wird nun von den Zeilen welche keine 0 enthalten abgezogen und den Zeilen welche eine 0 enthalten hinzugefügt. Diese Schritte werden so lange durchgeführt bis eine optimale Zuordnung der Nullen möglich ist. Das Optimum ist gefunden wenn $\min(C) = 0$ ist.

3 Übersicht der Defekte und Daten

Für lernbasierte Modelle sind die Daten das wichtigste Element, da die von den Modellen erlernte Korrelation direkt aus den Datenelementen extrahiert werden.

Damit die Modelle erfolgreich trainiert werden können, müssen die Daten eines Datensatzes möglichst Gleich verteilt und angepasst auf das zu lösende Problem sein. Hier werden zwei Datensätze erstellt, welche für Überwachte und Unüberwachte Lernmodelle entwickelt werden.

3.1 Datenausprägung

Die vorliegenden Daten kommen von Wafern, welche ein Industriepartner des Institut für Solare Energie der Fraunhofer Gesellschaft prozessiert hat. Die Zellen wurden in einem regulären industrienahen Prozess gefertigt und im Anschluss innerhalb des Fraunhofer Institutes vermessen. Dabei durchlaufen die einzelnen Proben mehrere Messverfahren. Diese Messverfahren, wie in Abbildung ?? dargestellt, können dabei vom Fraunhoferinstitut durchgeführt werden. Für diese Arbeit liegen lediglich die Daten aus den photolumineszenz Messungen vor. Die SHJ Zellen, welchen einen Aufbau wie in Abbildung 1.1 haben, wurden vor dem Auftrag der Metallisierungsschicht vermessen.

Die entstandenen Messdaten liegen in Form von Photolumineszenz Bildern von 4279 Wafern vor. Entsprechend der Prozessunterschiede werden diese in zwei Datensätze unterteilt. Die einzelnen Datensätze umfassen dabei Wafer, welche mit verschiedenen Rezepten prozessiert werden. Ein Überblick der verschiedenen Daten wird in Abbildung 3.1 gegeben. Aus der Abbildung ist ersichtlich, dass sich Zellen deutlich in den Messungen unterscheiden. Die Helligkeitsintensität der einzelne Proben zeigen eine große Streuung und es sind viele verschiedene Strukturen zu erkennen.

Die Daten liegen als Photolumineszenz Messungen vor und wurden mit einer Größe von 1024 x 1024 Pixeln im verlustfreien Tagged Image File Format aufgenommen. Die vermessenen Proben haben eine Kantenlänge von 166 mm

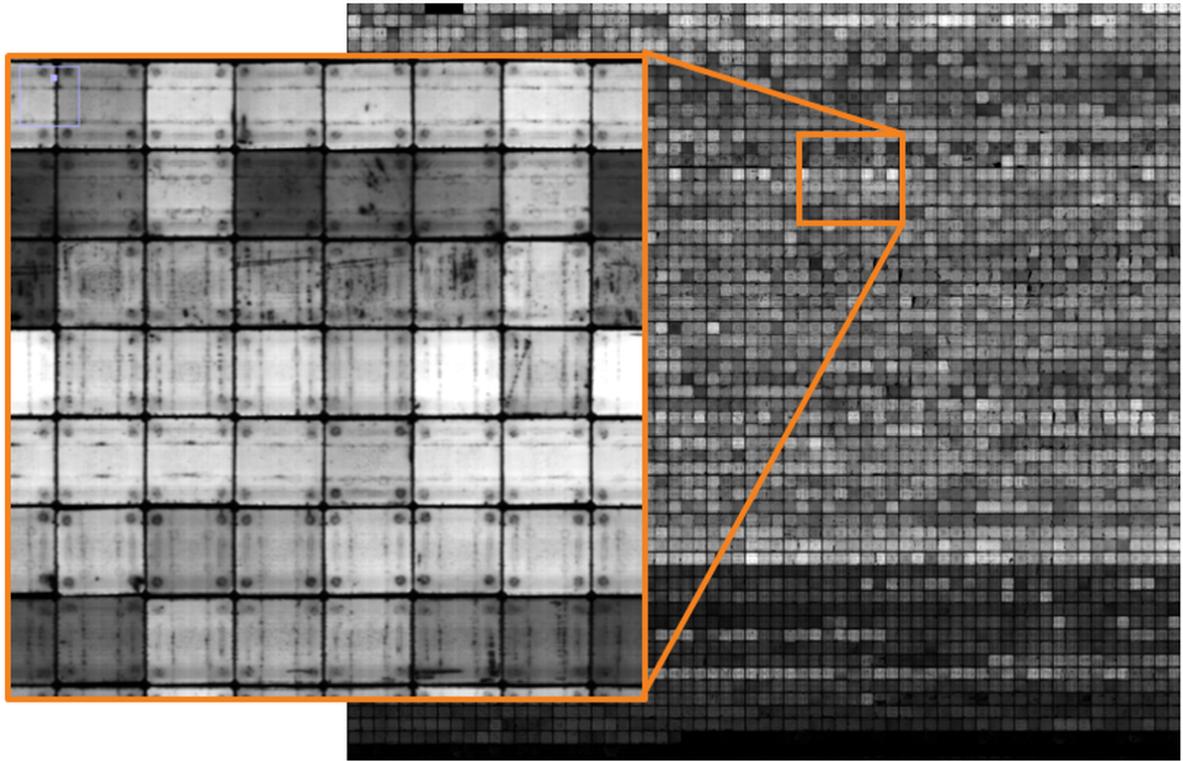


Abbildung 3.1: Überblick der PL Messdaten mit verschiedenen Prozess Rezepten

Die Datensätze *SHJ_1*, *SHJ_2* enthalten Bilder auf welchen Defekte sehr stark ausgeprägt sind. Die Datensätze *HJT_1* und *HJT_2* enthalten dagegen Daten in auf welchen nur wenige Defekte sichtbar sind.

Aus Diesen Daten wurden die Datensätze *DS_sup*, *DS_unsup* und *DS_test* erstellt. Die drei genannten Datensätze haben keine gemeinsamem Schnitt Menge. Eine exemplarische Auswahl an Daten aus den Datensätzen *SHJ_1* und *HJT_1* ist in Abbildung 3.2.

Während der Wafer des Datensatzes *HJT_1* nur schwache und wenige dunkle Defektstrukturen aufweist, sind in dem Beispiel aus dem Datensatz *SHJ_1* viele dunkle Strukturen,

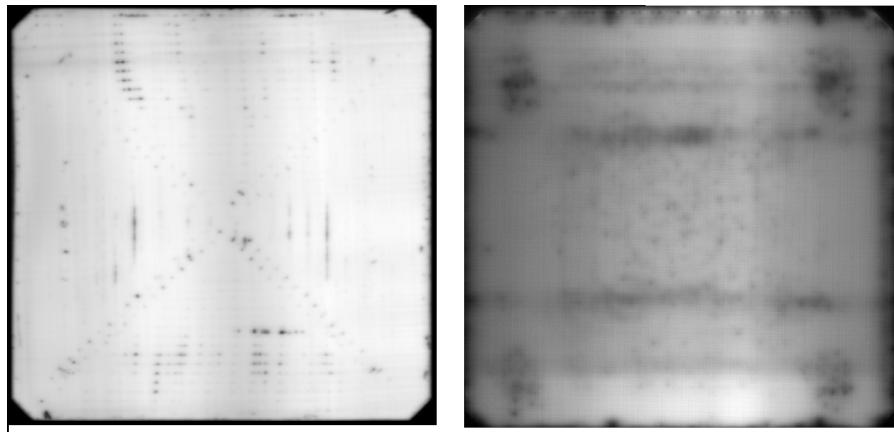


Abbildung 3.2: Datenbeispiel aus Datensatz *HJT_1* (links) und *SHJ_1* (rechts)

welche Defekte und Materialeinschlüsse widerspiegeln, zu sehen.

3.1.1 Defektbeschreibung

Auf den Daten der Datensätze kommen sehr viele verschiedene Defekte vor. Dabei sind sechs Defektesignaturen signifikant und sollen im Zuge dieser Arbeit Segmentiert werden. Die Strukturen lassen sich dabei in Greifer-, Band-, Rand-, Ripple-, Riss- und Moving Beam-Ausprägungen unterteilen. Die Greifer Strukturen lassen sich weiter in Be- und Entladegreifer unterteilen. In Abbildung 3.3 sind eine Beispielhafte Ausprägung der Defekte ersichtlich.

Die einzelnen Defekte sind dabei sehr klar in ihrer Strukturellen Ausprägung zu unterscheiden. Die Be- und Entladegreifer haben eine runde Struktur und sind Punktsymmetrisch zu einander angeordnet. Die Entladegreifer befinden sich absolut Näher am Waferrand, verglichen mit den Beladegreifer. Diese Strukturen kommen überwiegend als vierer Paar vor. Diese Defekte werden von entsprechenden Unterdruckgreifern beim Be- und Entladen der Wafer erzeugt. Die Entladegreifer befinden sich näher am Waferrand, da für das Lösen der Greifer aus dem Carrier wesentlich mehr Energie benötigt wird als für das Beladen selbiger. Banddefekte kommen als rechteckige Struktur über die gesamte Breite des Wafers vor. Diese entstehen über den Kontakt des Wafers zu den Transportbänder, über welche

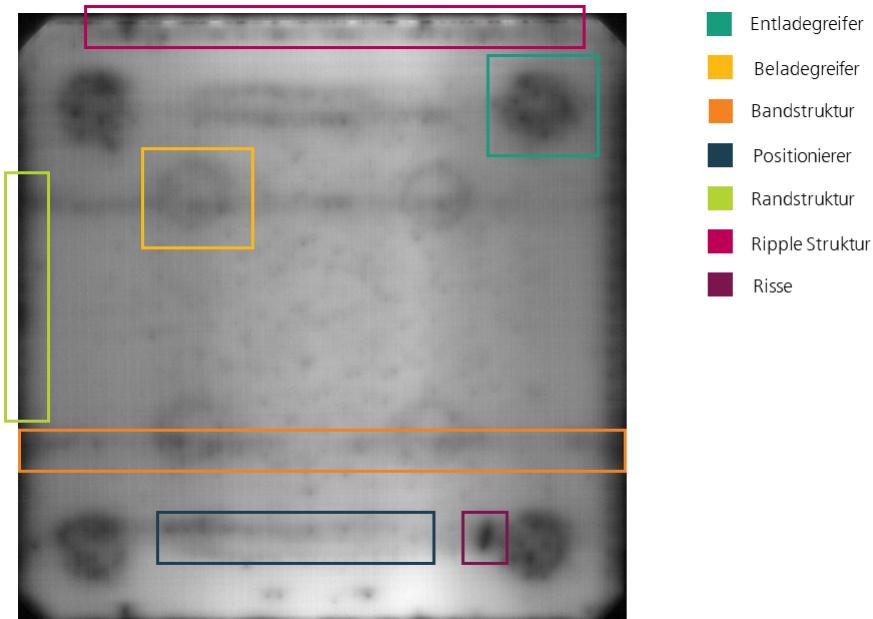


Abbildung 3.3: Überblick der differenzierten Defektstrukturen

sie bei der Produktion geleitet werden um zwischen den einzelnen Bearbeitungsschritten hin und her transportiert zu werden.

Die Defektklasse der Positionierer (Walking Beam) lässt sich als rechteckige Struktur die parallel zu den Banddefekten läuft beschreiben. Diese befinden sich um das vertikale Zentrum der Zelle und entsteht beim Positionieren des Wafers auf dem Transportband über zwei Aktuatoren.

Unregelmäßige Strukturen, welche allerdings meist Punktformig oder länglich sind, lassen sich zu der Defektklasse der Risse zählen. Dies können durch viele verschiedene Faktoren, während der gesamten Prozessdauer entstehen. Diese Defekte sind oftmals besonders stark auf den Photolumineszenz Bilder zu sehen.

Die Rand defekte zeichnen sich dadurch aus, dass diese an den Rändern des Wafers auftreten. Diese Können verschiedene Formen annehmen. Diese Defekte entstehen durch nicht korrekter Auftrag der einzelnen Schichten auf den Wafer. Als letzter signifikanter Defekt lässt sich der Ripple-Defekt deklarieren. Dieser tritt ebenfalls parallel zu den Bandabdrücken und den Posotionierern auf. Die Ripple Struktur zeichnet sich wiederkehrende

starke Punktstrukturen ab oberen Rand des Wafers aus. Der Ripple Defekt wird von einer Apparatur erzeugt, welche über den Wafer läuft. Die Solarzellen weisen weitere Defektstrukturen, welche sich durch dunklere Farbverläufe in den photolumineszenz Bildern darstellen, auf, welche sich aber nicht eindeutig auf einen speziellen Defekt zurückführen lassen. Diese werden aus diesem Grund aus der Defektbetrachtung in dieser Arbeit außer Acht gelassen. Da die Intensität der Helligkeit die elektrische Leitfähigkeit darstellt, sind diese kleineren und relativ hellen Defekten im Vergleich zu beispielsweise den viel dunkleren und flächenmäßig größeren Beladegreifer-Defekten zu vernachlässigen.

3.1.2 Herausforderung der Daten

Da die Daten aus realistischen Quelldaten generiert wurden, enthalten diese sehr spezifische Herausforderungen. Die Datensätze SHJ_1 und sHJ_2 beinhalten fast ausschließlich Daten, welche einige Defekte immer aufweisen. Dies gilt besonders für Greifer und Banddefekte. Die verschiedenen Defekte werden in Abschnitt 3.1.1 erläutert. Dies bedeutet, dass für die Segmentierung einige Klassen überrepräsentiert sind und andere unterrepräsentiert sind. Dies kann zur Folge haben, dass das zu trainierende Netzwerk eine Neigung zu den überrepräsentierten Klassen hat oder eine geringere Sensitivität gegenüber der unterrepräsentierten Klasse. Dies könnte allerdings auch eine Hürde bei der Auswertung, wie in Genauigkeitsmaß, darstellen. Dies könnte beispielsweise bedeuten, dass obwohl eine unterrepräsentierte Klasse nie erkannt wurde trotzdem eine hohe Genauigkeit erzielt werden kann.

Eine weitere Herausforderung ist die Abstinenz von Annotationen. Dies macht die Auswertung von Netzwerken wesentlich schwerer, da diese erwarteten Segmentationsklassen nicht direkt mit einer Grundwahrheit verglichen werden können. Eine Möglichkeit dem entgegen zu treten sind ein anderer Satz an Auswertungen. Einige Auswertungsmethoden, welche keine Grundwarheiten benötigen werden in Kapitel 2.7 erläutert. Zudem ist für das trainieren von überwachten Netzwerken eine annotierte Version der Daten zwingend

Notwendig. Diese muss somit in ausreichender Güte erstellt werden.

Die Defekte kommen oftmals als Überlagerung der einzelnen Defekte vor. Besonders die Defekte, welche dynamisch vorkommen und somit nicht immer im gleichen Bereich des Bildes vorkommt. Durch die Überlagerung der Defekte können diese nur sehr schwer von einandere getrennt werden und somit auch nur schwer einzeln von einander Klassifiziert und Segmentiert werden.

Die Aufnahmen aus den Photolumineszenz Messungen weisen für die Defekte eine hohe Unschärfe auf. Dies macht es sehr schwer die einzelnen Defektkanten klar zu trennen. Da viele Defekte nur eine schwache Ausprägung haben, stellt dies eine weitere Hürde für das Annotieren der Daten auf Pixelebene aber auch für die Netzwerke.

4 Ansatz

4.1 Übersicht

In dieser Arbeit werden Modelle zur Detektion von Defekten auf HJT Wafern anhand von photolumineszenz Messungen ausgewählt, entwickelt und ausgewertet. Diese Verfahren umfassen Überwachte Modelle und unüberwachte Modelle.

Überwachtes Lernen mit U-Net: Als überwachtes Lernmodell wurde ein U-Net Modell ausgewählt. Die Konfiguration des Netzwerkes wird in ???. Die für das Training und die Evaluation verwendeten Daten werden in Kapitel ?? beschrieben. Die benötigten Annotationen für dieses Modell sind in Kapitel ?? erläutert.

Unüberwachtes Lernen mit DINO und STEGO: Um Defektstrukturen ohne entsprechende Annotationen segmentieren zu können wird das unüberwachte Lernmodell DINO [caron2021emerging] verwendet. Das Modell wird in Kapitel ?? und die verwendete Konfiguration der Parameter in Kapitel ?? beschrieben.

Eine Erweiterung zu DINO stellt das Modell STEGO [STEGOhamilton2022unsupervised] dar. Dieses wird in Kapitel ?? und die exakte Modellkonfiguration in Kapitel ?? beschrieben.

Unüberwachtes Lernen mit STEgeo (geometrische Verlustfunktion) Viele der in den Photolumineszenz Messungen sichtbaren Defekte folgen einer geometrischen Anordnung. Daraus wurde eine neuartige Verlustfunktion kreiert, welche in das STEGO Modell integriert wurde. Diese Verlustfunktion wird im Detail in Kapitel ?? beschrieben.

4.2 U-Net

Das populäre und oft genutzte U-Net [UNETRonneberger2015] wird hier als Modell für eine Segmentierung mittels überwachter Lernmethode genutzt. Dieses I Das hier verwendete U-Net ist eine leichte Abwandlung des Originalmodells. Das Modell hat eine Tiefe von 5 Blöcken. Die Blöcke4.1 bestehen aus einer convolution, Batch Normalisation und einer Aktivierung mittels Rectified Linear Unit(ReLU). Innerhalb des Encoders wird die Dimensionsreduktion über einem Max-Pooling mit einer Kernelgröße von (2x2) erzielt. Die Dimensions-Erweiterung innerhalb des Decoders wird über ein upsampling erzieht.

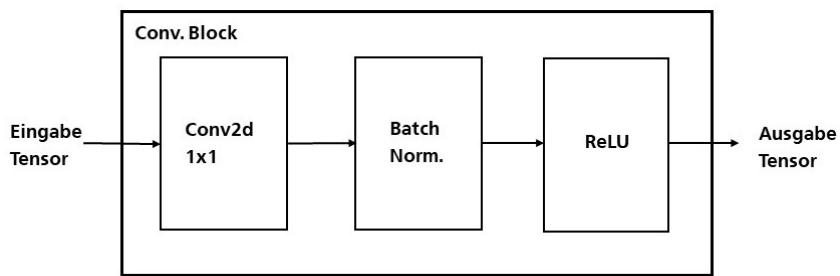


Abbildung 4.1: Einzelner Convolution Block in dem modifizierten U-Net. Dieser besteht aus einem conv Schicht, BN und ReLU als Aktivierung.

In Abbildung 4.2 werden als Eingabedaten Photolumineszenz Messungen mit einer Größe von 256 x 256 px² und 8 Eingabekanälen genutzt. Über die 4 max-pooling Reduktions-schichten wird eine Komprimierung auf 16 x 16 Pixel im Bottleneck des Modells erzielt. Im Decoder des U-Nets wird dann die ursprüngliche Dimension von 256x256 Pixel wieder über ein upsampling hergestellt.

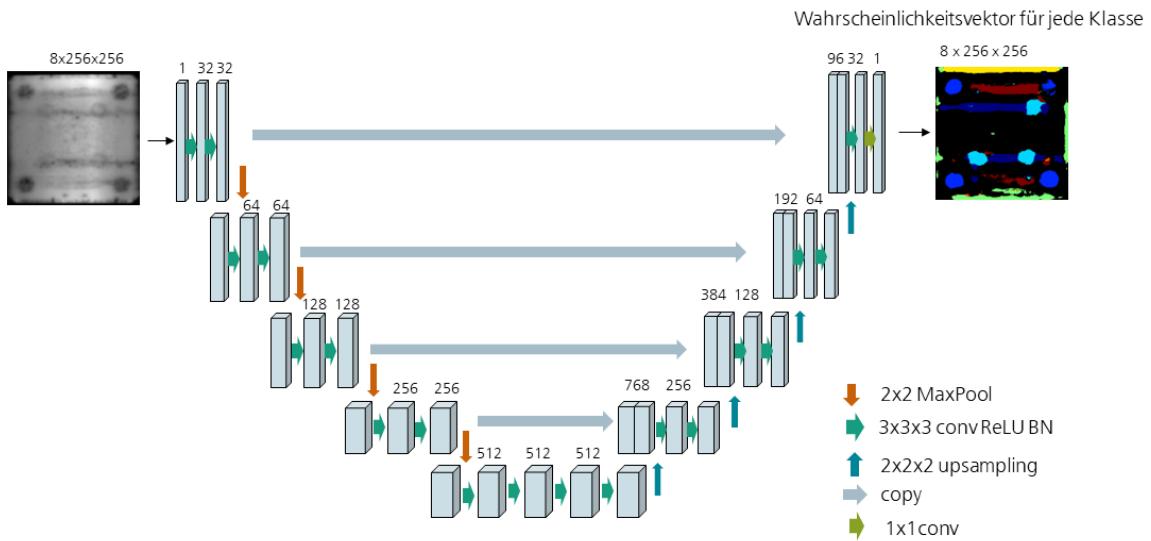


Abbildung 4.2: Schematischer Aufbau eines U-net Modells mit einer Auflösung von 32x32 Pixeln am tiefsten Punkt

4.2.1 U-Net pretrained

Als Erweiterung des U-Nets wird eine abgewandelte Version des U-Nets verwendet, in welchem ein Teil des Netzwerkes mit einem Vortrainierten Netzwerk ausgetauscht wird. Der Encoder Teil des U-Nets wird dabei durch ein ResNet [[resNet2015](#)] ausgetauscht. Die skip-Verbindungen des U-Nets werden dabei Aufrecht erhalten. Die einzelnen ResNet-Blöcke bestehen aus 3x3x3 Faltungen.

Während des Trainingsprozesses wird lediglich der Decoder Teil des Netzwerkes trainiert. Dies ermöglicht es die Vortrainierten Eigenschaften des ResNet Modells innerhalb des Encoders zu nutzen.

Das hier verwendete ResNet wurde auf dem Datensatz ImageNet [[imagenetdeng2009](#)] trainiert und hat eine top1-Genauigkeit von 80,53% erzielt.

Die Bildgröße innerhalb der Vorverarbeitung wurde auf 244x244 verkleinert, da das Training des ResNet Modells mit dieser Bildgröße durchgeführt wurde.

4.3 DINO: Self-Distillation with NO labels

Aufbauend auf den bemerkenswerten Ergebnissen der in Kapitel 2.5.3 beschriebenen Vision Transformern wird eine unüberwachter Lernansatz genutzt um das Modell zu trainieren. Caron et. al. zeigen in dem Paper Emerging Properties in Self-Supervised Vision Transformers [caron2021emerging], dass es möglich ist, mit einem selst-überwachtem Ansatz ein außergewöhnliches Ergebnis mit 78,3 % als top-1 k-NN und 80,1% als linearen Klassifizierer auf dem ImageNet[imagenetdeng2009] Datensatz zu erreichen.

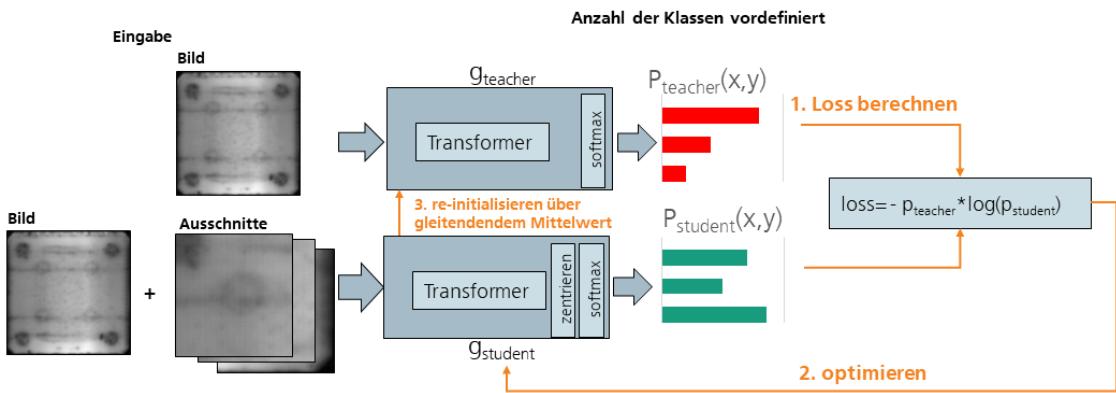


Abbildung 4.3: DINO Architektur mit exemplarischen Daten

Ihr Modell DINO interpretiert den selbst-überwachte Ansatz dabei als eine Wissens-Distillation. Wissens Distillation ist ein Prozess in welchem die Parameter in eimem Netzwerk g_{θ_s} so angepasst werden, dass es der Ausgabe eines zweiten Netzwerkes g_{θ_t} entspricht. Die Ausgabe ist eine Wahrscheinlichkeitsmatrix über K Dimensionen, welche durch eine Softmax Normalisierung berechnet wird.

Die Wahrscheinlichkeitsverteilung P_s lässt sich durch,

$$p_s(x)^{(i)} = \frac{\exp(g_{\theta}(x)^i / \tau_s))}{\sum_{k=1}^K \exp(g_{\theta}(x)^k / \tau_s))}$$

darstellen, wobei τ_s ein Temperatur Parameter darstellt, welcher die Intensität Ausgabeverteilung bestimmt. Das Netzwerk g_{θ_t} verhält sich equivalent dazu. Die Netzwerke g_{θ_s} und g_{θ_t} selbst, bestehen aus je einem Vision Transformer als Backbone wie in Kapitel 2.5.3 dargestellt und einem Projektionskopf. Der Projektionskopf besteht dabei aus einem drei-lagigen multi-layer Perzeptron mit einer versteckten Dimension von 2048. Auf diese folgt eine l_2 Normalisation und ein gewichtnormalisierter Fully Connected Layer.

Die Verteilungen des Netzwerkes g_{θ_t} werden von g_{θ_s} über ein cross-entropy loss angenähert, in dem die Parameter des Netzwerkes g_{θ_s} durch einen stochastischen Gradientenabstieg angepasst werden. Dies stellt sich durch $H(p_1, p_2) = -p_2 \log p_1$ dar. Damit dies selbstüberwacht geschieht, werden verschiedene verändere Sichten V von einem Bild generiert, welche zwei globale Sichten und mehrere kleinerer lokale Ansichten. Während das Netzwerk g_{θ_t} nur die globalen Sichten zur Verfügung stehen, bekommt das Netzwerk g_{θ_s} zusätzlich die lokalen Ansichten. Diese fördert eine Korrespondenz von den einzelnen geringeren skalierten Sichten zu den globalen Ansichten.

In jeder Iteration werden die Parameter des Netzwerkes g_{θ_t} durch einen gleitenden Durchschnitt (EMA) [**emaHe2019**] der vorherigen Iterationen des Netzwerkes g_{θ_s} neu initialisiert. Dabei werden die Parameter mit

$$\theta_t \leftarrow \lambda \theta_t + (1 - \lambda) \theta_s$$

reinitialisiert. λ folgt während dem Training einem Cosinus-Scheduler von 0.995 bis 1 [**bootstrapSelfSupGrill2020**]. Dies kann als einen Mittlung der stochastische Approximation mit einem exponentiellen Decay verstanden werden. Caron et. al. zeigen damit, dass das durch den Scheduler initialisierte Netzwerk g_{θ_t} qualitativ bessere Ergebnisse als das Netzwerk g_{θ_s} erzielt. Dadurch wird das Training des Netzwerks g_{θ_s} , welches versucht die Ergebnisvektoren von g_{θ_t} zu approximieren, geleitet.

Gegen einen Dimensionskollaps wird in DINO mit einer Zentrierung und Schärfung des gleitenden Durchschnitts der Ausgabe des Netzwerkes g_{θ_t} . Selbstüberwachte Methoden wie DINO tendieren dazu, nach trivialen Lösungen zu suchen. Um dies zu verhindern muss ein Mechanismus gegen Dimensionskollaps implementiert werden. Zentrieren hilft dem Modell dabei, dass sich keine der Dimensionen als dominant hervorhebt aber verstärkt den Kollaps. Das Schärfen verringert hingegen den Kollaps zu gleichmäßigen Verteilung aber lässt eine Dimension dominant werden. beim Ausführen beider Operationen, wird ein Gleichgewicht zwischen diesen hergestellt.

4.4 STEGO: Self-supervised Transformer with Energy-based Graph Optimization

Das STEGO [STEGO^{hamilton2022unsupervised}] Modell wird in seiner Standardkonfiguration verwendet. Diesem Netzwerk liegt das DINO ?? Modell F als Backbone zugrunde. Aufbauend darauf ist eine Segmentations-Head S integriert, welches ein simples Feed-Forward Netzwerk darstellt. Der Aufbau dieses besteht aus einem Dropout gefolgt von einem linearen Netzwerk aus einem conv2d Layer und anschließendem sequenziellen Block aus conv2d, ReLU und einer weiteren conv2d. Die Faltungen werden jeweils mit einem rezeptiven Feld von (1x1) ausgeführt.

Während des gesamten Trainings wird lediglich das Netzwerk S trainiert. Das Backbone F wird hierbei eingefroren. Dies ermöglicht es diesem Netzwerk eine schnellere Konvergenz zu erzielen. Um das Modell Eingabegrößen unabhängig

4.5 STEgeo: Self-supervised Transformer with GEometric feature loss

Das Modell STEgeo ist eine Abwandlung des STEGO [STEGO^{hamilton2022unsupervised}] Modells. STEgeo soll sich die symmetrisch auftretenden Defektestrukturen für die Segmentation zu nutze machen. Da der Wafer eine Symmetrie zu seinem Zentrum aufweist und die Defektklassen (vgl. Kapitel ??) der äußeren und inneren Greifer, Bandstrukturen

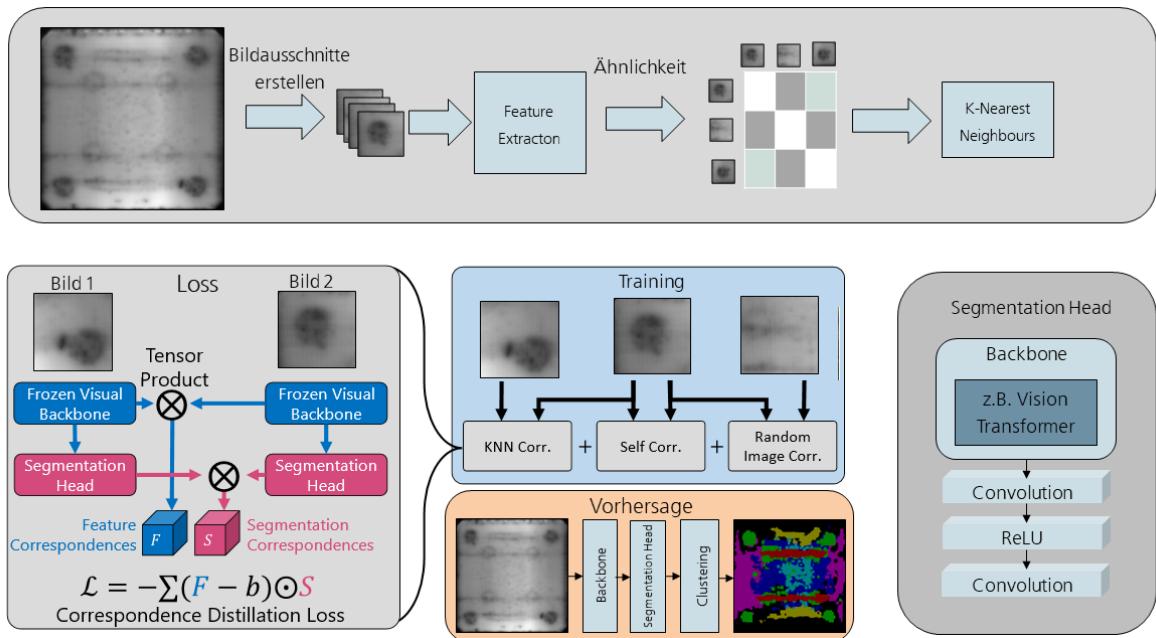


Abbildung 4.4: STEGO Architektur mit exemplarischen Daten. Berechnung K-Nächste Nachbarn(Oben). Berechnung Loss (Mitte und Links). Darstellung des Segmentations Kopfes(Rechts). Vorhersagemodell(Unten)

und Positionierer zum einen als Paarkombination aber auch symmetrisch zum Zentrum vorkommen, kann eine Verlustfunktion für die vorliegenden Daten erarbeitet werden.

Der neu gebildete Verlust schließt die Position der betrachteten Bildausschnitte, p_1 und p_2 , mit in die Berechnung des Verlustes ein. Für den Verlust G_{geo} wird das Produkt aus der gewichtete Distanz wd und der Tensorkorrelation ?? der Bildausschnitte gebildet.

$$L_{geo} = W_{p1,p2} * T_{corr_{p1,p2}} \quad (4.1)$$

$W_{p1,p2}$ (4.2) berechnet die Gewichtung der Bildausschnitte aus dem Abstand zwischen den Bildausschnitten. Die dafür notwendige Distanz $d_{p1,p2}$ wird über die L1 Distanz der Bildpunkte berechnet. Der Abstand zwischen den Ausschnitten wird dann auf den Wertebereich zwischen 0 und 1 skaliert. Dabei stellt 1 den Maximalen Abstand d_{max} 4.4, mit der Bildbreite i_b bzw. Bildhöhe i_h und Bildausschnittsbreite p_b bzw. Bildausschnittshöhe p_h ,

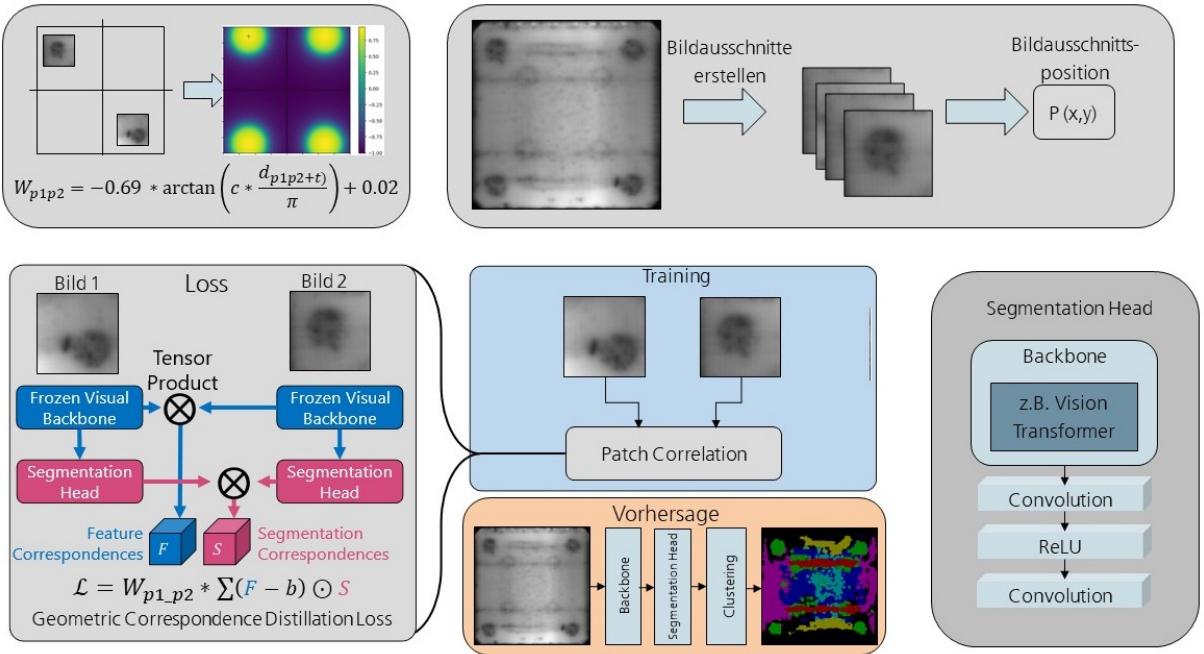


Abbildung 4.5: STEgeo Architektur mit exemplarischen Daten. Bildausschnitts Distanzen mit Gewichtungsfunktion(Oben Links) Bildausschnitts Erstellung und Positionsrechnung (Oben Rechts). Berechnung Loss (Mitte und Links). Darstellung des Segmentations Kopfes(Rechts). Vorhersagemodell(Unten)

dar.

$$W_{p1p2} = -0.69 * \arctan\left(\frac{c * (d_{p1p2} + t)}{\pi} + 0.02\right) \quad (4.2)$$

$$d_{p1p2} = \sqrt{(|p1_x| + |p2_x|)^2 + (|p1_y| + |p2_y|)^2} \quad (4.3)$$

$$\begin{aligned} d_{max_x} &= \frac{i_b}{2} - p_b \\ d_{max_y} &= \frac{i_h}{2} - p_h \\ d_{max} &= \sqrt{d_{max_x}^2 + d_{max_y}^2} \end{aligned} \quad (4.4)$$

Mit den verschiedenen Faktoren von W_{p1p2} können verschiedene Szenarien realisiert werden. Für eine geringere Toleranz der Verschiebung zwischen den beiden Bildausschnitte ist ein kleineren Wert für t zu wählen und für einen steileren Übergang zwischen positiver und negativer Verstärkung der Korrelation ist ein größeres c zu wählen. Die Abbildung 4.6 illustriert diese Gewichtung mit verschiedenen Parametern.

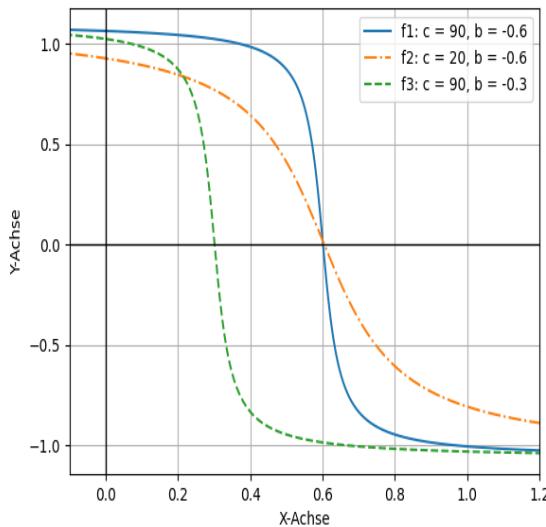


Abbildung 4.6: Funktion zur Gewichtung der Distanz zwischen zwei Bildausschnitten mit c : Steigungskonstante und t : x-Achsen Verschiebung

Beispiele zur Gewichtung zwischen zwei Bildausschnitten ist für Punktsymmetrie in Abbildung 4.7, und für Achsensymmetrie in X beziehungsweise Y Richtung in Abbildung 4.8 zu sehen.

dashed und cross erklären ->

Für die Tensorkorrelation L_{corr} (4.5) wird das Tensorprodukt der Feature Tensoren gebildet.

Dabei werden die Tensoren F und S Elementweise miteinander multipliziert [STEGOhamilton2022u]

Der Tensor F ist dabei aus dem Backbone und der Tensor S aus dem darauf aufgesetzten Netzwerk wie in Kapitel 4.4 beschrieben.

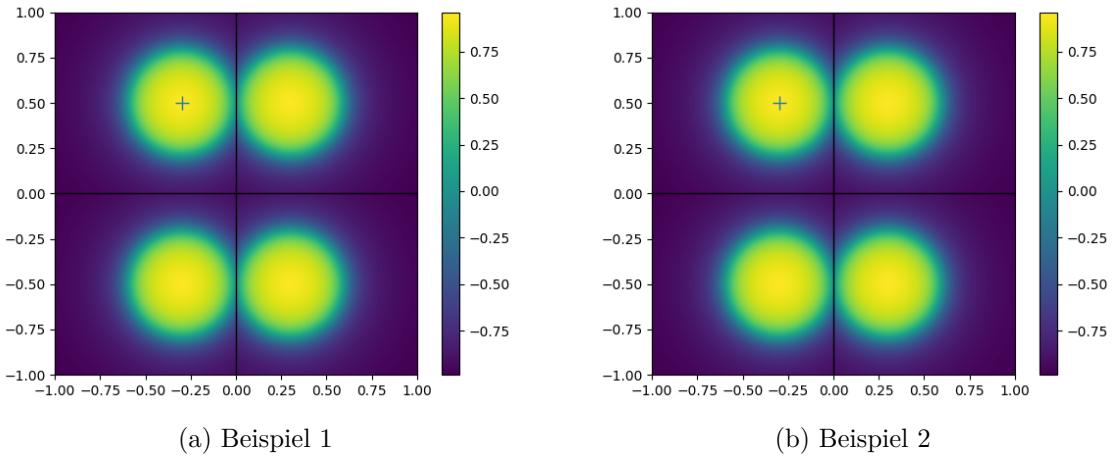


Abbildung 4.7: Visualisierung der Distanzvektoren anhand von Beispielen bei Punktsymmetrie

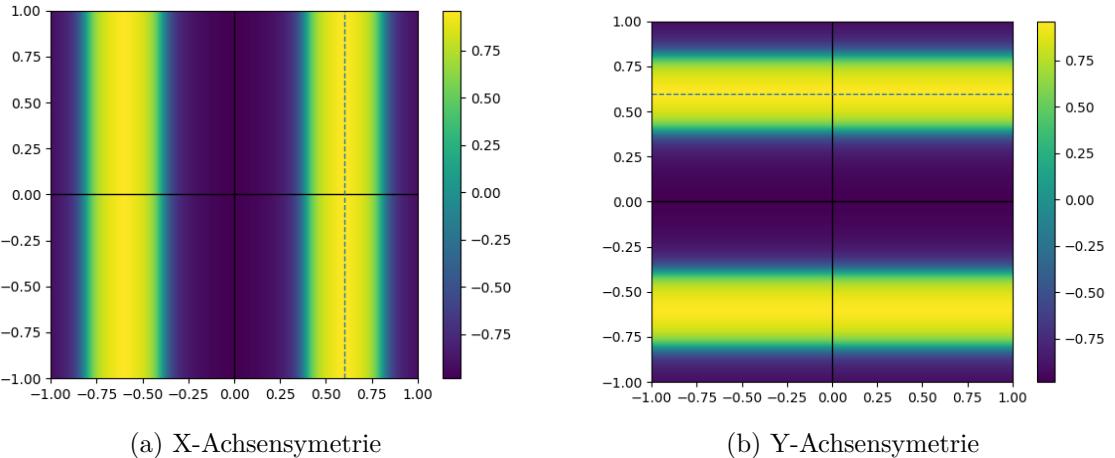


Abbildung 4.8: Visualisierung der Distanzvektoren anhand von Beispielen bei Achsensymmetri-sche

$$L_{corr}(x, y) = - \sum_{hwij} F_{hwij} * S_{hwij} \quad (4.5)$$

Ein Eigenschaftstensor lässt sich dabei mithilfe von Formel 4.6 berechnen. C sind dabei die Bildkanäle und h, w sowie i, j die räumlichen Dimensionen. Diese spiegeln die Kosinus Ähnlichkeit der Eigenschaften von Tensor f an Position h, w und Tensor g an Position i, j

wieder.

$$F_{hwij} := \sum_c \frac{f_{chw}}{|fhw|} \frac{g_{cij}}{|gij|} \quad (4.6)$$

5 Experimentalteil

5.1 Gestaltung der Experimente

In dieser Thesis wurde ein Überwachtes und drei Unüberwachte Modelle auf einem erstellten Datensatz trainiert, um Waferdefekte anhand von Photolumineszenz Messungen zu detektieren. Abbildung 5.1 zeigt den Prozess des Experiments. Nach der Aufbereitung des Datensatzes werden für das Unüberwachte Modell Annotationen erstellt. Darauf folgend werden alle Modelle entsprechend ihrer Konfigurationen Trainiert. Im Anschluss werden die Modelle ausgewertet und mit einander verglichen.

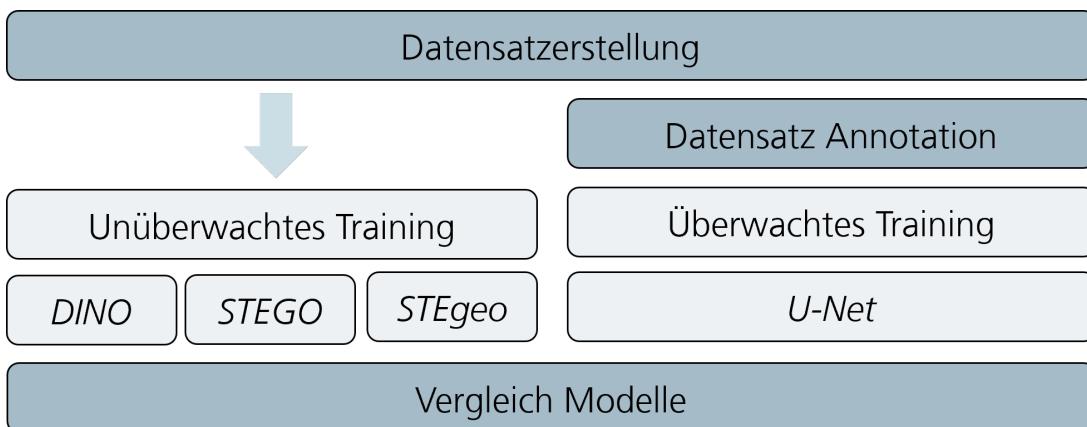


Abbildung 5.1: Übersicht des Experiments mit verschiedenen Modellen

5.2 Datensatz

5.2.1 Datensatz Überblick

Für die verschiedenen Modelle wurde ein Datensatz erstellt. Dieser teilt sich auch in einen Unüberwachten Datensatz, welcher Daten mit verschiedenen Waferrezepten enthält und ein Überwachten Datensatz, welcher eine Teilmenge des Unüberwachten Datensatzes, mit

	Proben	Annotationen	Bildgröße	Modell
DS_uns	4279	0	1024x1024	DINO, STEGO, STEgeo
DS_sup	60	60	1024x1024	U-Net

Tabelle 5.1: Überblick des Datensatzes und verwendeten Modelle

händisch generierten Annotationen, widerspiegelt. Ein Überblick über die Anzahl der verwendeten Proben und den Verwendungszweck der Datensatzteile ist in Tabelle 5.1 gezeigt.

Eine Gegenüberstellung einer Probe ist in Abbildung 5.3 zu sehen. Dabei ist auf der linken Seite die Photolumineszenz Messung und auf der Rechten Seite die Annotationen der einzelnen Klassen auf Pixel Ebene. Eine Aufschlüsselung der Farben mit entsprechenden Klassen ist in Tabelle 5.2 gegeben. Die Annotationen wurden mit einem internen Labeling Werkzeug 5.2 des Fraunhofer ISE erstellt und im Anschluss von Kantenpunkten der einzelnen Klassen zu Flächenkarten in welcher jeder Pixel einer Klasse zugeordnet ist umgewandelt.

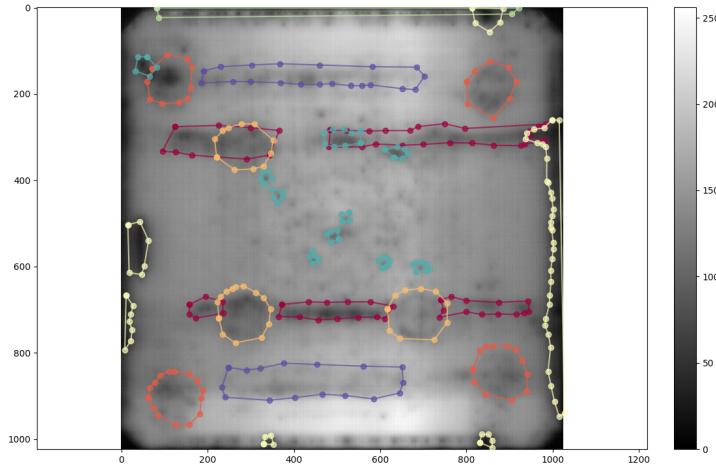


Abbildung 5.2: Erstellungsprozess der Annotationen mit Fraunhofer ISE internem Werkzuge

Der Datensatz DS_{uns} besteht aus 4279 Proben welche mittels einer Photolumineszenz Messung aufgenommen wurden. Die Messungen wurden einer Vorverarbeitung, wie in Kapitel ?? beschrieben, unterzogen. Für alle verwendeten Modelle wurden die Daten in

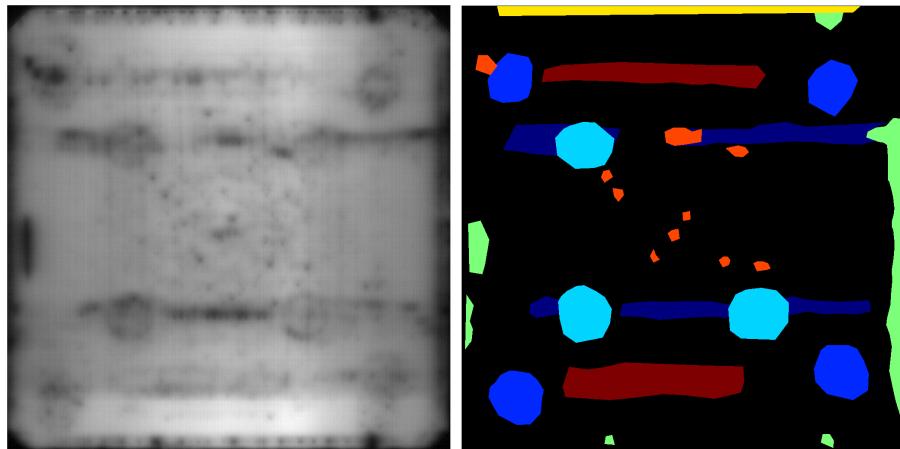


Abbildung 5.3: Datenpaar aus Datensatz DS_{sup}

Klasse	Farbe
Greifer außen	Blau
Greifer innen	Hellblau
Transportband	Dunkelblau
Positionierer	Rot
Ripple	Gelb
Randdefekt	Grün
Risse/Sonstiges	Orange
Wafer	Schwarz

Tabelle 5.2: Überblick der Klassen mit korrespondierenden Farben in den Annotationen

einen Trainings- und Validations-Teil aufgeteilt, mit respektive 3900 und 379 Messungen. Der Datensatz DS_{sup} besteht aus 60 Photolumineszenz Messungen. Diese Messungen wurden ebenfalls einer Vorverarbeitung, welche eine Korrektur der Verzerrung, Verzeichnung und Alinierung beinhaltet, unterzogen. Der Datensatz ist in einen Trainingsteil mit 50 Messungen und einen Validationsteil mit 10 Messungen unterteilt. Als Testdaten werden hier Proben aus dem Datensatz DS_{uns} verwendet.

Die Daten des Datensatz DS_{sup} bestehen aus einem Datenpaar, bestehend aus der Photolumineszenz Messung und einer entsprechend angefertigten Pixelweisen Annotation.

5.2.2 Daten Vorverarbeitung

?? Die rohen Bilddaten sind mit Verzerrung, Verzeichnung und ungleicher Alinierung behaftet. Die Verzerrung und Verzeichnung kommt dabei aus der Apparatur, mit der die Messungen aufgenommen wurden. Die inkonsistente Alinierung hingegen durch händische Platzierung der Solarzelle unter dem Messgerät. Damit die Daten den zugrunde liegenden Wafern entsprechen, wurden diese Artefakte entfernt. Den Unterschied zwischen Rohbild und Aliniertem Bild ist in Abbildung 5.4 zu sehen.

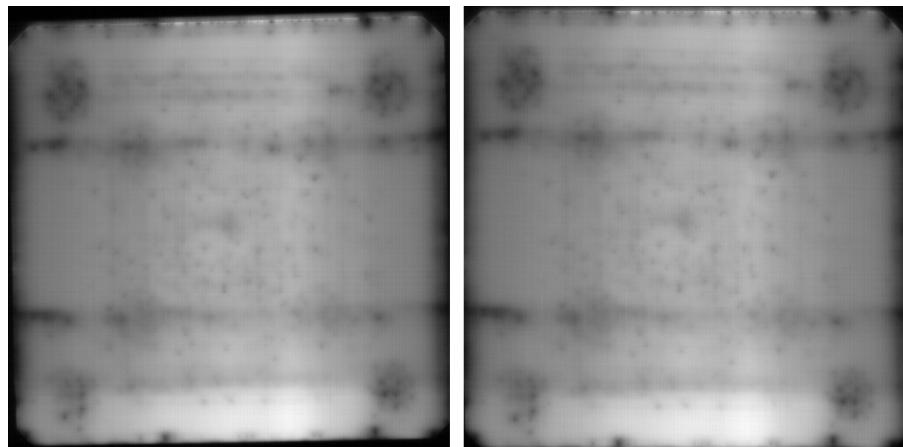


Abbildung 5.4: Unaliniertes Rohbild (links), welches Verzerrung und Verzeichnung aufweist und aliniertes Bild(rechts)

Um Messungen, welche möglichst Nahe an den Daten der realen Wafern sind, zu bekommen, wurden die Artifakte die durch die Messung erzeugt wurden negiert. Die Alinierung wurden über eine Erkennung der Waferkanten mit anschließender Entzerrung der Messung und Rotation des Bildes durchgeführt. Somit ist eine Darstellung die Näher an der Realität ist gewährleistet.

5.3 Segmentationsmodelle

5.3.1 Überblick Netzwerktraining

Die Überwachte Segmentation wird mit einem U-Net Modell und die Unüberwachte Segmentation mit den Modellen DINO, STEGO und STEgeo durchgeführt. Alle Modelle wurden mit den Vorverarbeiteten Daten trainiert.

Damit eine größere Variation der Daten gegeben ist, werden verschiedene Bildverarbeitungen im Trainingsprozess angewendet. Diese Vorverarbeitungsschritte sind wichtig um ein Neuronales Netzwerk effizient trainieren zu können. Innerhalb dieses Schrittes wurden folgende Daten-Daten-Augmentationen vorgenommen.

- Bildgrößenänderung
- Zufälliger Auswahl eines Bildausschnitts
- Horizontales und vertikales Spiegeln
- Helligkeit, Kontrast, Sättigung und Farbton
- Batch-Normalisierung

Die hier aufgelisteten Verarbeitungsschritte werden auf den allinierten, entzerrten und entzeichneten Photolumineszenz Messungen ausgeführt. Wie in Abbildung ?? gezeigt wird auf das Ausgangsbild ein zufälliger Bildausschnitt gewählt. Dieser hat entweder die Bildgröße mit der das Vortraining ausgeführt wurde oder beim Ende-zu-Ende Training eine Größe von 256 Pixel x 256 Pixel.

Zusätzlich wurden die Daten vor dem Trainingsprozess Standardisiert. Dies Standardisierung wurde mittels Gleichung 5.1 durchgeführt, in der x das entsprechende Bild, μ_x der Mittelwert und σ_x die Standartabweichung.

$$x_{norm} = \frac{x - \mu_x}{\sigma_x} \quad (5.1)$$

Alle Modelle wurden auf Nvidia Titan XP GPUs mit 12GB Speicher und Linux Debian Distribution 5.10.0 mit CUDA Version 11.7.

5.3.2 Experiment 1: U-Net

Innerhalb dieses Experiment wurden einzelne Konfigurationen des U-Nets, wie in Kapitel 4.2 beschrieben, mittels dem Datensatz DS_{sup} trainiert. Diese exp1 -> standart U net exp2 -> unet mit gewichtetem Loss exp3 -> Encoder ausgetauscht mit Pretrained ResNet50d

5.3.3 Experiment 2: DINO

exp1 -> DINO NotPretrained exp2 -> DINO Pretrained

5.3.4 Experiment 3: STEGO

exp1 -> STEGO with NotPretrained DINO exp2 -> STEGO with Pretrained DINO

5.3.5 Experiment 4: STEgeo

exp1 -> pointsym with best DINO exp2 -> axissym with best DINO

6 Ergebnisse

Die in Kapitel 5 beschriebenen Experimente umfassen eine Vielzahl an Parameterkombinationen. Um die Ergebnisse übersichtlich darstellen zu können werden die Ergebnisse in drei Teilen dargestellt. In Kapitel 6.1 werden die experimentellen Ergebnisse der Überwachten Lernmethoden dargestellt. Kapitel 6.2 enthält die Ergebnisse aus den Experimenten der Unüberwachten Lernmodellen. Es werden in jedem dieser Kapitel exemplarisch Bilder der entsprechenden Vorhersagen und der entsprechenden Metriken gezeigt.

Als letztes wird dann in Kapitel 6.3 ein quantitativer Vergleich zwischen allen Experimenten gegeben.

6.1 Überwachtes Lernen

6.1.1 U-Net

Das U-Net wurde auf dem Datensatz für das Überwachte Lernen trainiert. Das Modell hat 8 Eingabe Kanäle welche die einzelnen Klassen des Modells repräsentieren.

U-Net mit ungewichteter Verlustfunktion (U-Net_unwgtLoss)

Im Experiment [ref U-Net_unwgtLoss](#) wird eine Standardversion des U-Net Modells genutzt und der Verlust wird über die Kreuzentropie berechnet. Dabei werden die einzelnen Klassen innerhalb der Verlustfunktion nicht gewichtet um ein Ungleichgewicht der Klassenvorkommen der Daten auszugleichen.

Die Abbildung 6.1 zeigt exemplarisch das Eingabebild, die Annotation und die Vorhersage mit diesem Modell. Die Annotation zeigt 7 verschiedene Klassen. Diese setzen sich aus 6

Defektklassen und dem Hintergrund zusammen.

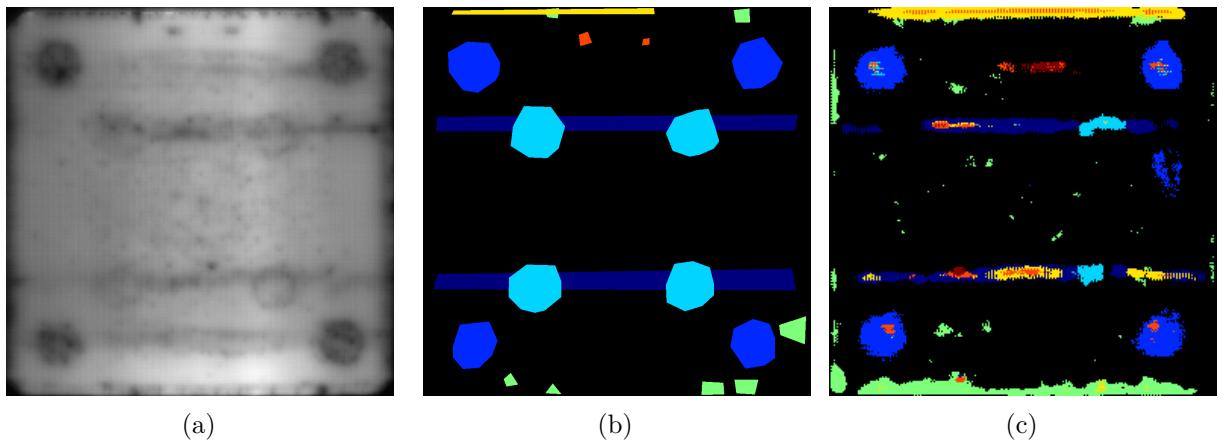


Abbildung 6.1: Exemplarischer Vergleich aus Experiment U-Net_unwgtLoss mit Ausgangsbild (a), Annotation(b) und Vorhersage(c)

In der qualitativen visuellen Auswertung der Vorhersage, werden die äußeren Greifer (hellblau) sehr gut erkannt und haben eine große Übereinstimmung mit der Annotation. Die Bandabdrücke (dunkel blau) werden ebenfalls in einer hohen Güte erkannt und stimmen sehr gut mit der händisch gelabelten Bild überein. Am oberen Rand der Vorhersage wird der Ripple Defekt (gelb) ebenfalls gut vorhergesagt, allerdings ins dies Klasse auch in weiteren Bereichen des Bildes zu sehen und stellt eine falsche Klassifizierung dar. Die Entlade Greifer (cyan), Positionierer(rot) und Randdefekte (grün) werden alle nur partielle erkannt und enthalten eine hohe Fehlerquote. Für diese Defekte wurde keine gute Segmentation gefunden. Der Wafer (schwarz) und die umliegende Hintergrundfläche wird gut als ein Klasse erfasst. Über die Risse (orange) kann in diesem Exemplar keine Aussage gemacht werden, da keine vorhanden sind.

Eine quantitative Evaluation des Experimentes ist in Abbildung 6.2 aufgetragen. Es ist eine maximale Intersection over Union von **XX** zu sehen. Dies ist auf ... zurückzuführen.
hier noch Precision und Recall

Der Trainings- und Validierungsverlauf des Verlustes zeigt, dass dies ein gut geeignetes

Modell ist, und kein Overfitting oder Underfitting stattfindet. Sowohl der Trainings- als auch Validierungsverlust fallen sehr schnell und bleiben nach 70 Epochen sehr konstant. Der Validierungsverlauf ist dabei leicht größer als der Trainingsverlauf.

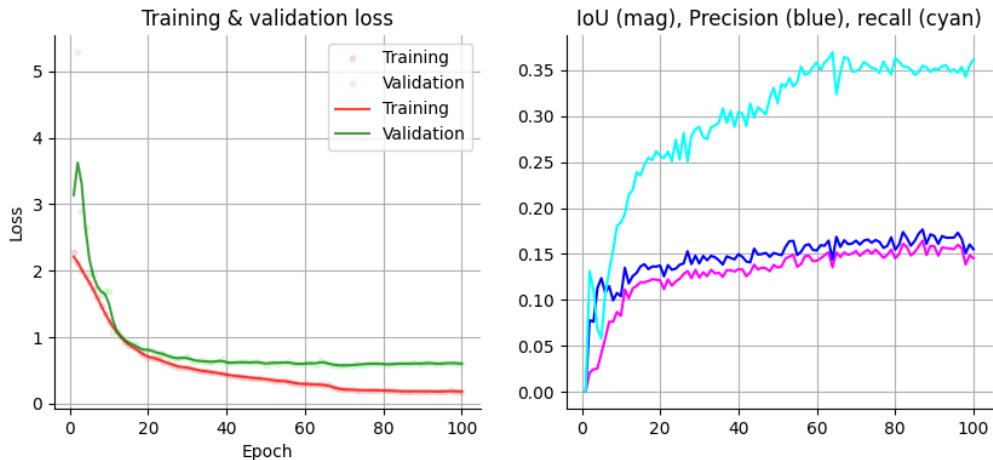


Abbildung 6.2: Trainingsverlauf des U-Net Modells mit ungewichteter Verlustfunktion

U-Net mit gewichtete Verlustfunktion (U-Net_wgtLoss)

Das Experiment [ref U-Net_wgtLoss](#) benutzt eine ähnliche Architektur wie Experiment [ref U-Net_unwgtLoss](#). Die Berechnung der Kreuzentropie Verlustfunktion wird jedoch gewichtet um einem Ungleichgewicht der Klassen vorkommen im verwendeten Datensatz entgegen zu steuern.

Die Qualitative Auswertung zeigt, dass eine Hohe Übereinstimmung zwischen der Annotation und dem Vorhersagebild besteht besonders gut werden Ripple(gelb), Entladegreifer(hellblau), Beladegreifer(cyan), Transportbänder(dunkel blau) erkannt. Auch wenn es für die Klassen Randdefekte(grün), Positionierer(rot) und Risse(orange) eine Übersegmentierung gibt, sind diese klar zu erkennen. Die Uniformität der einzelnen Segmentationen ist gegeben. Besonders kleine Strukturen wie Riss-Defekte werden schlecht erkannt.

Die Quantitative Auswertung in Abbildung 6.4 zeigt, dass mit diesem Modell eine IoU

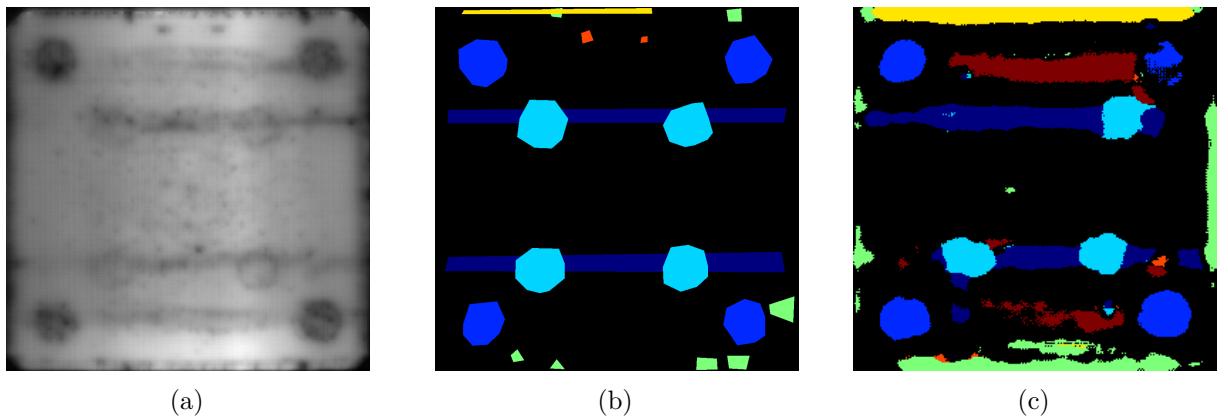


Abbildung 6.3: Gegenüberstellung von Ausgangsbild, Annotation und Vorhergesagtem Bild

von xxx erreicht werden kann. Der Recall ist besonder hoch mit xxx und einer Precision von xxx . Nach 40 Epochen ist schon das Maximum der Intersection over Union erreicht und steigt bis zum Ende des Experiments nicht weiter.

Der Verlauf der Verlustfunktionen des Trainings und der Validation verlaufen die ersten 20 Epochen nahezu identisch. Ab Epoche 80 verlaufen die Graphen nahezu horizontal und der Verlust ändert sich nur noch minimal. Der Verlust ist nach 100 Epochen mit 0.01 im Training und mit 0.0045 in der Validierung sehr gering.

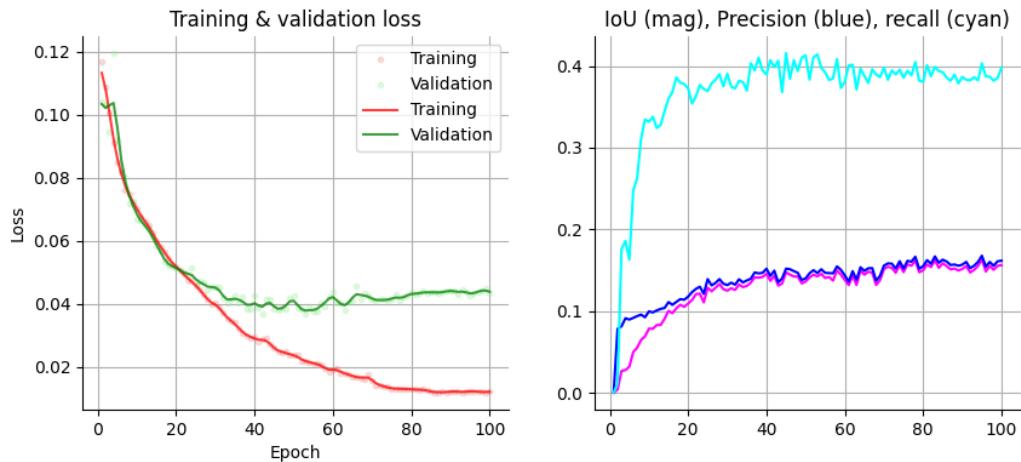


Abbildung 6.4: Trainingsverlauf des U-Net Modells mit gewichteter Verlustfunktion

U-Net mit vortrainiertem Encoder

6.2 Unüberwachtes Lernen

6.2.1 DINO

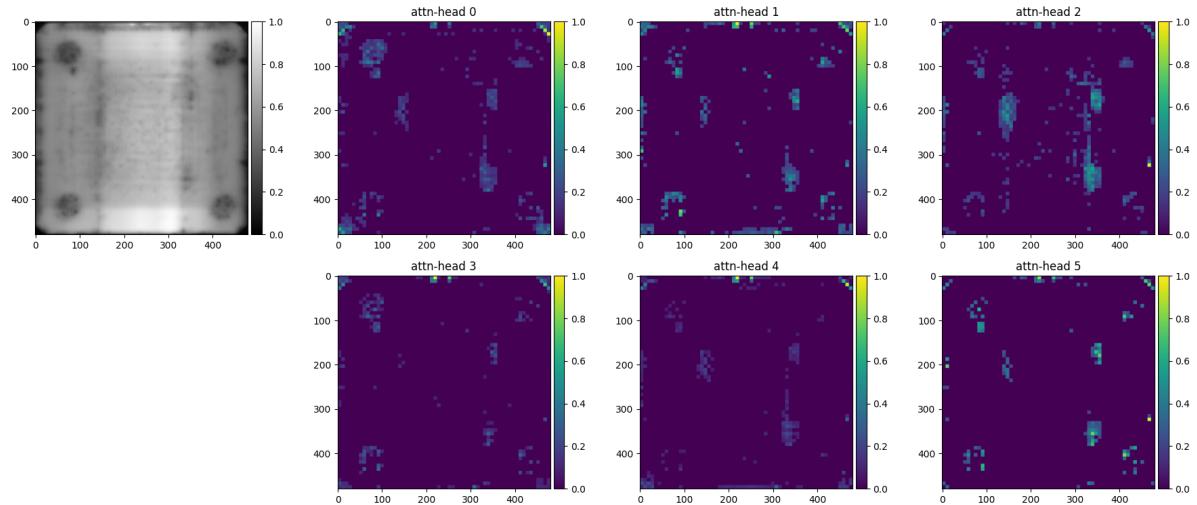


Abbildung 6.5: Visualisierung der Aufmerksamkeitskarten von DINO

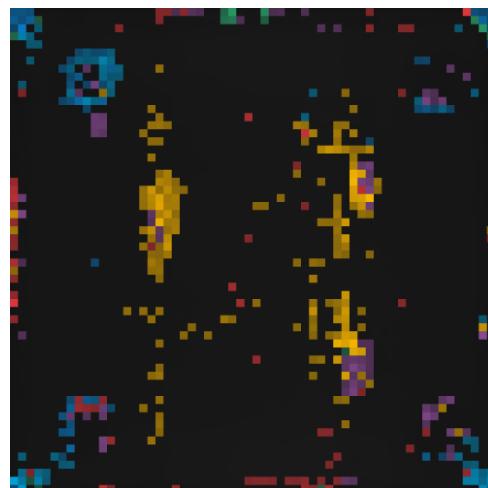


Abbildung 6.6: Segmentation anhand der Aufmerksamkeitskarten aus dem DINO Modell ohne Pretraining

6.2.2 STEGO

STEGO mit DINO als Backbone

Wie in Kapitel ?? beschrieben wurden verschiedene Konfigurationen von STEGO getestet und analysiert. Es wurde ein nachtrainiertes Modell von DINO(vgl. ??) verwendet. Sowohl die Eingabe- als auch die Ausgabegröße ist 480px x 480px. Es wurde insgesamt 100000 Schritte trainiert was mit der verwendeten Datensatzgröße

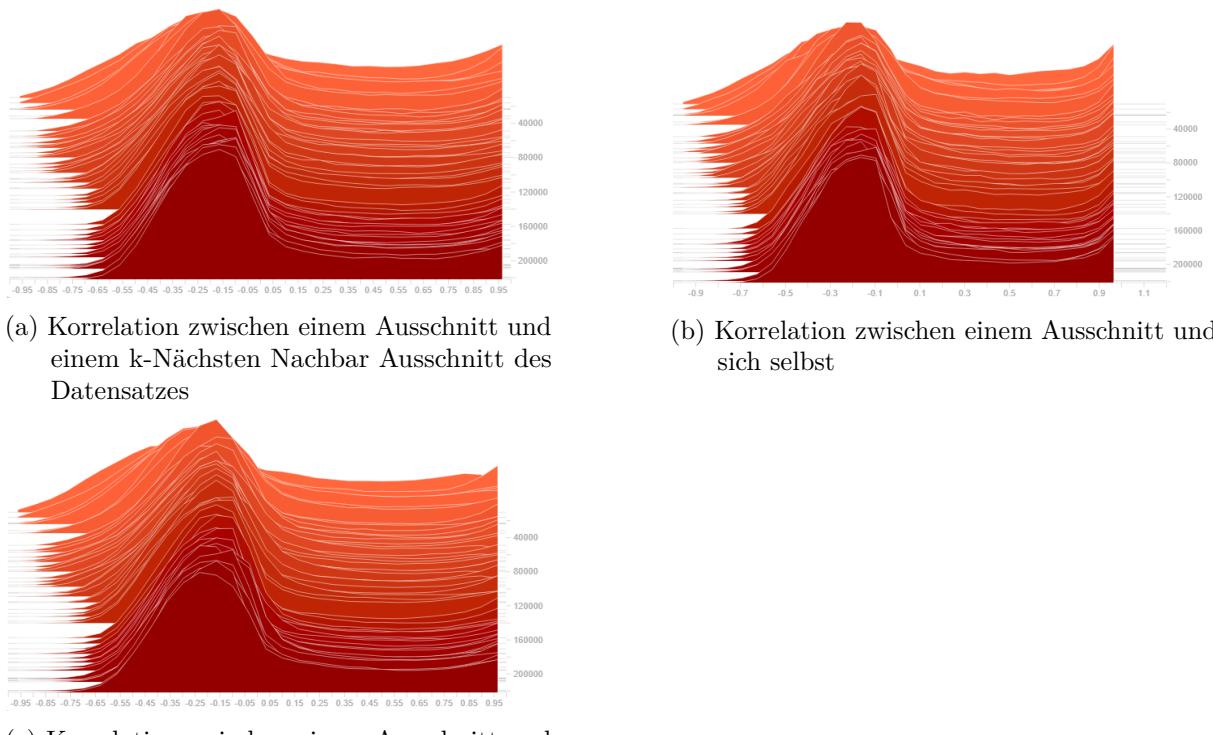


Abbildung 6.7: Vergleich der berechneten Tensorkorrelationen

Vergleichende Analyse der Verlustfunktion Ähnlichkeitsmaß der features -> Negatives Ähnlichkeitsmaß immer schlecht -> parameter haben keinen Einfluss

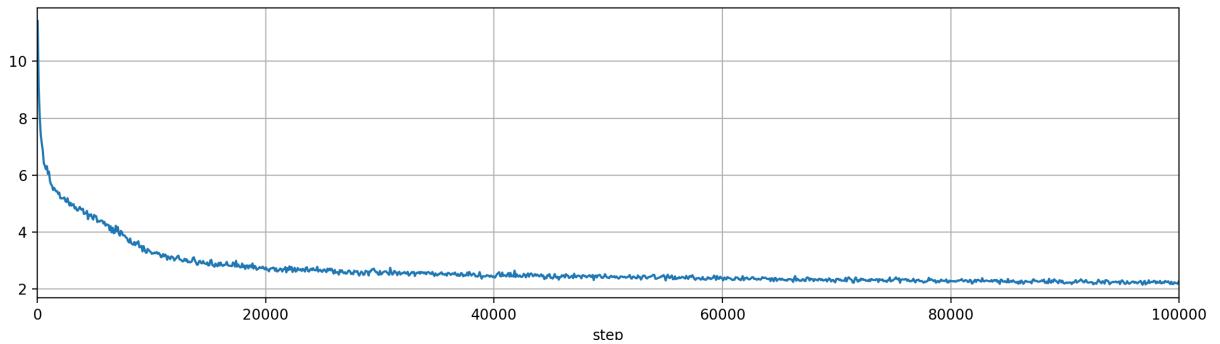


Abbildung 6.8: Verlust des Modells

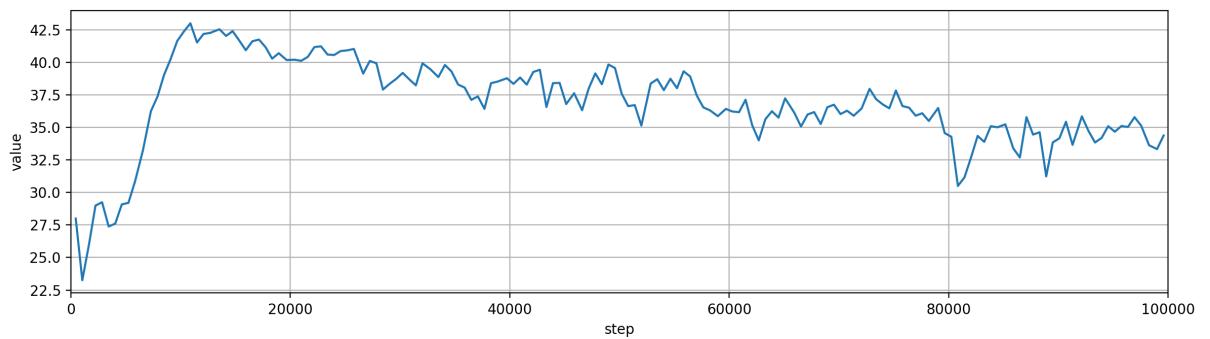


Abbildung 6.9: Genauigkeit

6.3 Vergleich der Modelle

6.3.1 Vergleich Experimente unüberwachtes Lernen

Interpretation der einzelnen Lossgraphen Entgegenwirken der Parameter (Patchgröße, Datensatz anpassung, bias) Geometrische Loss als Gegenpol > Beispiele - STEGO -> hoher Zusammenhang zwischen den patches die nicht ähnlich sind, da mit hoher Wahrscheinlichkeit in zwei verschiedenen Patches ähnliche Features liegen.

6.3.2 DINO

- Beschreibung was genau analysiert wurde
- welche Fragestellung damit abgedeckt werden soll
- Aufschlüsseln was alles miteinander gekoppelt wurde (Transformationen, Daten)
- Graphen und Ergebnisse darstellen

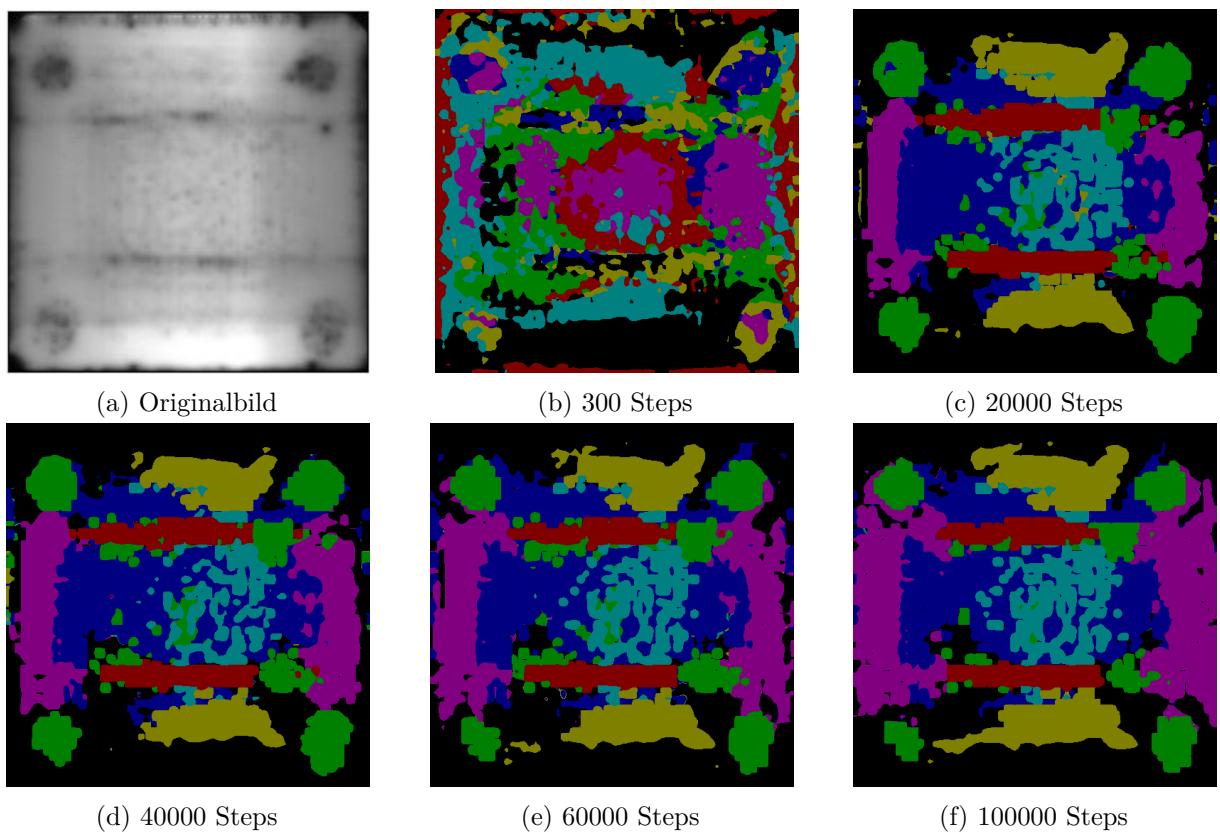


Abbildung 6.10: Veränderung der Vorhersage über Epochen

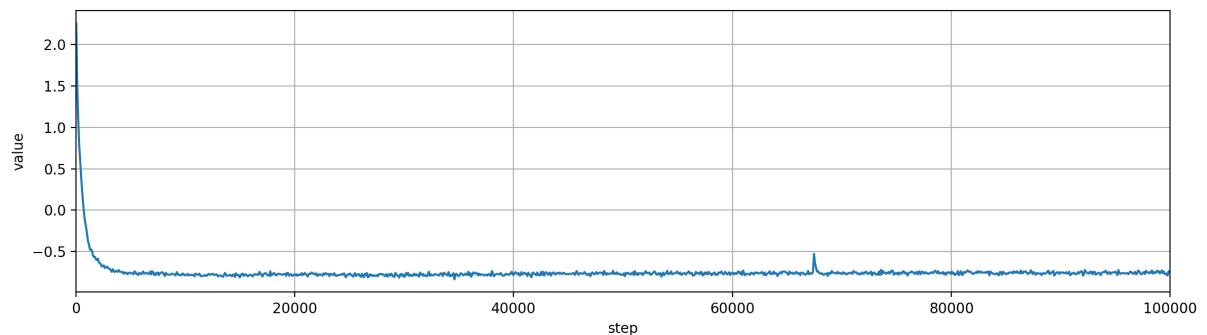


Abbildung 6.11: Verlust des Modells

- Graphen beschreiben
 - Ergebnisse(Bilder) von Prediction beschreiben
 - Beschreiben wo passiert das gut, wo schlecht (Analyse der einzelnen Defekte, Positon, lokalität)
-

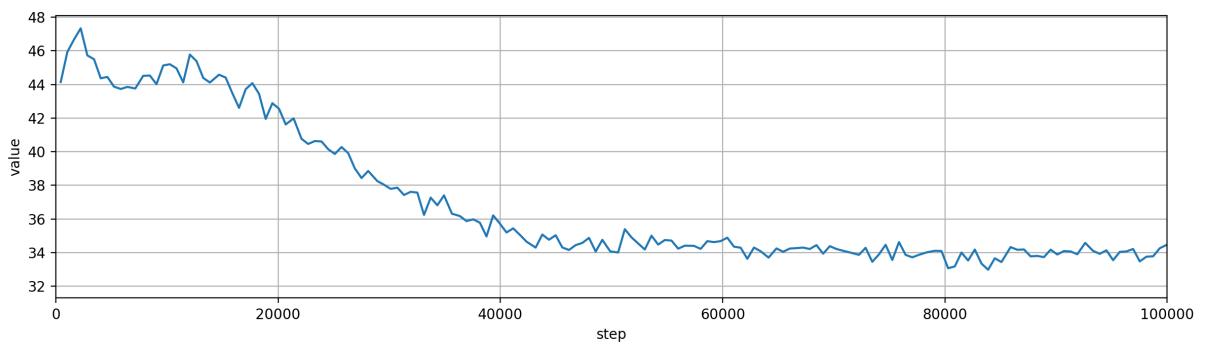


Abbildung 6.12: Genauigkeit

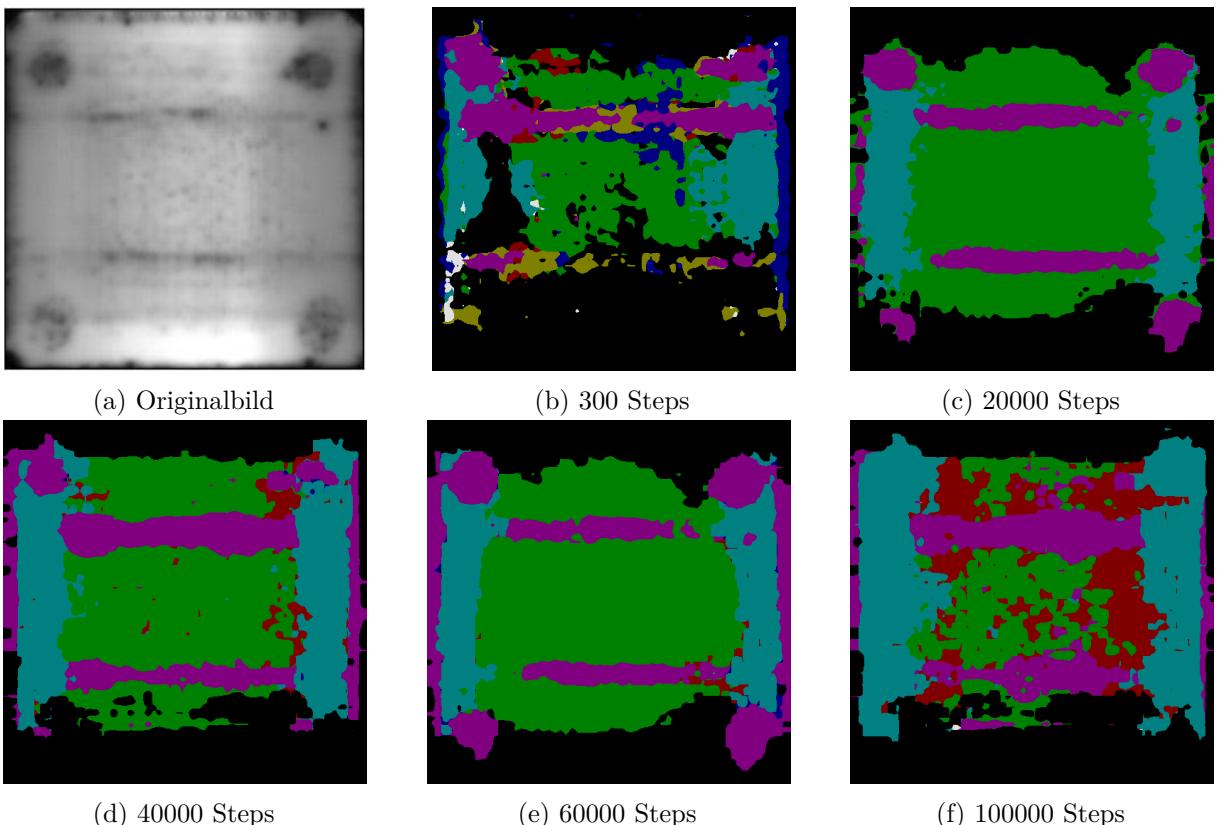


Abbildung 6.13: Veränderung der Vorhersage über Epochen ohne Transformationen des Ausgangsbildes

6.3.3 DINO + STEGO

- Beschreibung was genau analysiert wurde
 - welche Fragestellung damit abgedeckt werden soll

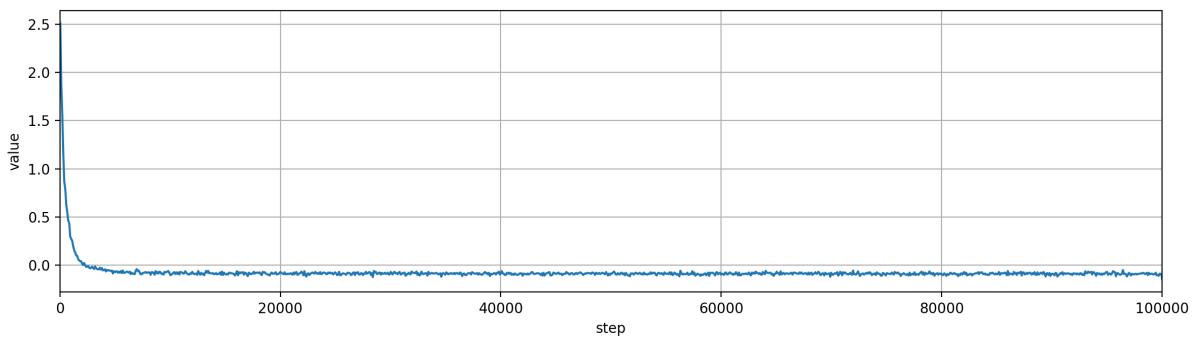


Abbildung 6.14: Verlust des Modells

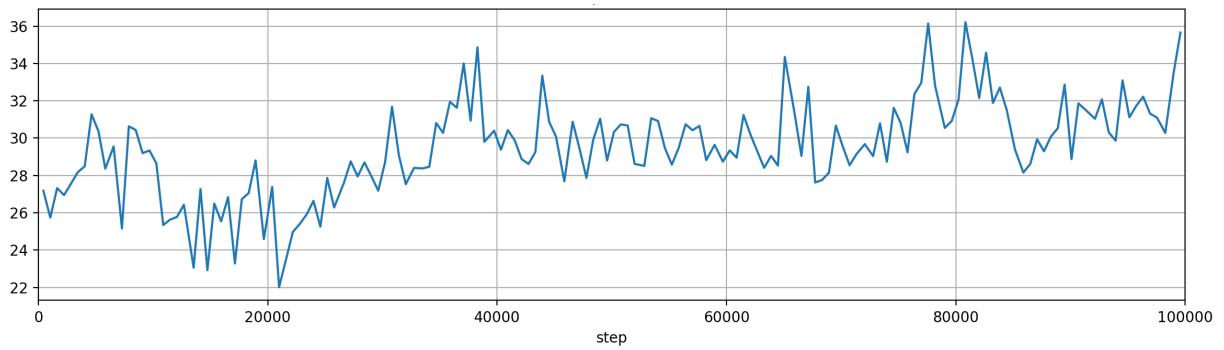


Abbildung 6.15: Genauigkeit des Modells

- Aufschlüsseln was alles miteinander gekoppelt wurde (Transformationen, Daten)
- Graphen und Ergebnisse Darstellen
- Graphen beschreiben
- Ergebnisse(Bilder) von Prediction beschreiben
- Beschreiben wo passiert das gut, wo schlecht (Analyse der einzelnen Defekte, Position, Lokalität)

6.3.4 Vergleich Modelle

- SAM vs UNet
- Dino vs DINO + STEGO
- unsupervised vs supervised
- vgl einzelne Modelle

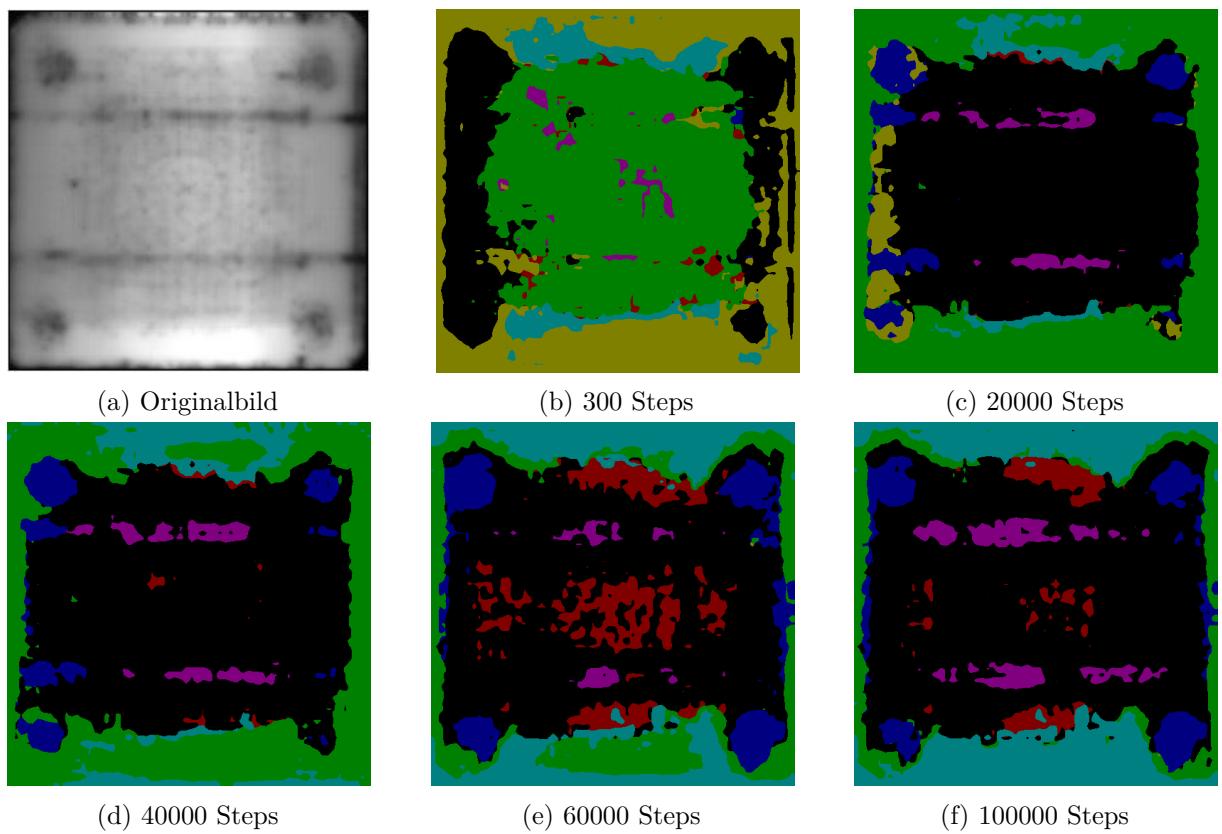


Abbildung 6.16: Veränderung der Vorhersage über Epochen mittel geometrischer Verlustfunktion

7 Diskussion

- Modelle wurden erfolgreich implementiert und verfeinern (anpassung auf daten) - Diskussion ist eine Bewertung der Ergebnisse

Das Ziel dieser Arbeit war die Segmentierung von Defekten in Photolumineszenz Messungen von HJT-Solarzellen mittels Deep-Learning Methoden. Bei den zu detektierenden Defekten handelt es sich ausschließlich um Strukturen, die durch Transportvorrichtungen innerhalb der Solarzellenprozessierung eingebracht werden. Darüber hinaus wurde untersucht, ob unüberwachte Lernmethoden eine mögliche Alternative zu überwachten Lernmethoden, für diesen Anwendungszweck mit entsprechenden Daten, darstellen um eine einfachere Adaption auf weitere Zellen zu erreichen und die **Kosteneffizienz (besser erklären)** zu erhöhen. Die Defektsegmentierung dieser Zellen ist Teil der Charakterisierung von HJT Solarzellen.

Die im Kapitel 6 vorgestellten Ergebnisse zeigen, dass die Defekte mit Hilfe von Photolumineszenz Messungen erfolgreich segmentiert werden können.

Die Experimente der Überwachten Lernmethoden wurden auf dem händisch erstellten Datensatz mit Datenpaaren aus Messung und Label auf Pixelebene trainiert und evaluiert. Die trainierten Modelle des Überwachten Lernens zeigten in der Qualitativen Auswertung sehr gute Ergebnisse, welche

- Reproduzierbarkeit im vgl zu supervised Learning nicht vorhanden (Vergleich zu aktuellen State of the Art (iou,...)) <- Mathias fragen -> unterschied zu standard Datensätze-> Farbbilder, Anzahl der Klassen
- STEGO 1. Backbone 2. verschiedene Vorfaktoren (in ein plot) 2.1 Kleinere Gewichte erzeugen kleineren Loss höhere Acc , allerdings umgekehrt in den visuellen Ergebnissen -> niedrigere Losses erzeugen schlechtere visuelle Bilder 2.2 Cluster Loss starker

einfluss auf Total loss Wenn Gewichte kleiner gewählt, dann hat Cluster loss stärkerer Einfluss 3. im unsupervised fall keine gewichtung des Hintergrundes -> vllt deswegen Metriken keinen aufschluss auf die quantitative Qualtiät der Bilder !!! Metriken (Acc, IoU) spiegeln nicht umbedingt die visuelle qualität der Daten wieder -> Anpassung des Losses Angepasster Loss mit vergleichbaren Gewichtungen -> geo <-> nicht geo -> im bezug auf Losses werden Ergebnisse besser, allerdings visuell nicht -> für self04 b04 knn02 b04 rnd04 b02 stimmen Losses mit visuellen Ergebnissen überein -> niedrigerer Loss -> bessere visuell Auswerung -> niedrigere neg Loss Werte aber höhere Streuung in negativ loss STEGO sehr fragil, kleine änderungen in der Parameter -> große Auswirkungen

8 Zusammenfassung und Ausblick

Das Ziel dieser Arbeit ist es Defekte auf HJT-Wafern welche durch den Prozessierungs Prozess entstehen zu detektieren. Diese Defekte werden aktuell nicht charakterisiert. Diese Arbeit gibt einen Ansatz diese Defekte mittels Deep Learning Methoden zu detektieren, welche für eine umfangreiche Charakterisierung der Wafer sehr wichtig ist. Es wird weiter untersucht ob, die Detektion mittels Unüberwachten Lernmethoden erfolgt werden kann um eine noch schnellere und Kosten effektivere Charakterisierung der Wafer vorzunehmen.

Da für die Vorliegenden Photolumineszenz Daten keine Annotationen vorliegen, macht dies, das trainieren und evaluieren von Überwachten Modellen schwer. Aus diesen Grund wurde ein gepaarter Datensatz für das Training und die Evaluation dieser Messungen erstellt.

Es wurden drei verschiedene Deep Learning Modelle implementiert und evaluiert. Die Modelle lassen sich in Unüberwachte Lernverfahren und Überwachte Lernverfahren einteilen. Die Unüberwachten Modelle wurden dabei auf den gesamten Daten trainiert während die Überwachten Modelle lediglich auf einem Teil dieser Daten, welche die erstellten Bildannotationen enthalten, trainiert und evaluiert werden. Es wurden verschiedene Experimente mit diesen Modellen ausgeführt und im Zuge der Unüberwachten Modelle eine neue Verlustfunktion, welche auf den Eigenschaften der verwendeten Daten basiert, entwickelt. Die Modelle wurden mit verschiedenen Konfigurationen der Verlustfunktionen für die Optimierung ausgeführt.

In in dieser Arbeit implementierten Modelle sind in der Lage Defekte auf photolumineszenz Messungen zu detektieren und segmentierten. Das Überwachte Modell hatte Probleme mit der Ungleichgewichtung der Klassen, welche sich über eine Gewichtung des Verlustes ausgleichen ließ. Die Unüberwachten Modelle hatten das Problem, dass kein ausreichender Gegenpol zu den positiven Korrelationen der Verlustfunktion gefunden wurden. Um diesem Problem entgegenzuwirken, wurde eine Verlustfunktion entwickelt, welche die Position des verwerteten Bildausschnitts mit beachtet. Sowohl die Überwachten als auch die Unüberwachten Modelle zeigen gute Ergebnisse. Allerdings in der Qualitativen Auswertung eine höhere Güte in den Ergebnissen der Überwachten Modelle zu sehen.

Basierend auf der Evaluation der Ergebnisse, hat diese Arbeit weiterhin großes Potenzial für Verbesserungen und Erweiterungen. Der Datensatz könnte um weitere Zelltypen und Prozessierte Chargen erweitert werden um eine breitere Anwendung auf verschiedene Zelltypen zu gewährleisten. Ebenso ist eine Erweiterung der Unüberwachten Modelle mittels Conditional Random Fields, für eine bessere räumliche Auflösung, denkbar.

