doi.org/10.37544/1436–4980–2024–05–82

GEFÖRDERT VOM

Bundesministerium
für Bildung
und Forschung

Machine Learning Anwendung beim Laserschneiden

Generalisierung eines maschinellen Lernmodells für das Laserschneiden auf verschiedenen Blechdicken

K. Leiner, D. Wall, J. Kühlem, M. Huber

ZUSAMMENFASSUNG In diesem Beitrag wird ein Machine-Learning-Modell auf Basis eines Autoencoders trainiert. Das Ziel des Modells ist es, Fehlschnitte beim Laserschneiden zu erkennen, da fehlerhafte Schnitte zu hohen Ausschussraten führen. Die Literatur zeigt, dass es möglich ist, mit Machine Learning Fehler beim Laserschneiden zu erkennen. Eine noch nicht vollständig gelöste Problemstellung ist die Anwendung eines Modells auf verschiedene Prozessparameter. In diesem Beitrag wird ein Ansatz vorgestellt, wie sich ein trainiertes Modell auf verschiedenen Blechdicken anwenden lässt. Zu diesem Zweck wird der Autoencoder mit einer erweiterten Verlustfunktion trainiert. Das Modell ist dann in der Lage, einen Fehlschnitt generalisiert über mehrere Blechdicken zu erkennen.

STICHWÖRTER

Machine Learning Anwendung, Laserschneiden, Autoencoder

Generalising a machine learning model for laser cutting on various sheet thicknesses

ABSTRACT In this work, a machine learning model is trained on the basis of an autoencoder. The aim of the model is to recognise faulty cuts during laser cutting, as faulty cuts lead to high reject rates. The literature shows that it is possible to recognise errors in laser cutting using machine learning. One problem that has not yet been fully solved is the application of a model to various process parameters. In this paper, an approach is presented on how a trained model can be applied to different sheet thicknesses. The autoencoder is trained with an extended loss function. The model is then able to recognise an incorrect cut generalised over several sheet thicknesses.

1 Einleitung

Seit den 1970er Jahren steigt die Nachfrage nach Laserschneidanlagen in der Blechbearbeitung [1]. Die Maschinen- und Anlagenbauer müssen die Werkzeugmaschinen ständig weiterentwickeln, um den sich ändernden Marktbedingungen und den steigenden technologischen Anforderungen zu begegnen [2]. Die Nachfrage nach Automatisierungslösungen und den dafür benötigten Sensorsystemen nimmt zu.

Die vorgestellten Versuche wurden an einer Flachbett-Laserschneidmaschine „TruLaser5030“ der Firma Trumpf durchgeführt. Die Maschine ist in **Bild 1** dargestellt. Die Maschine besteht aus zwei Hauptkomponenten. In der Mitte von Bild 1 ist die Werkzeugmaschine zu sehen, auf der der Schneidvorgang stattfindet. Links daneben befindet sich die Strahlquelle, in der der Laserstrahl erzeugt wird. Über einen Lichtleitkabel wird der Laserstrahl zum Schneidkopf geführt, der sich in der Werkzeugmaschine befindet. Hinter der Maschine befindet sich der Palettenwechsler. Hier belädt der Maschinenbediener die Maschine mit den zu schneidenden Blechen. Nachdem die Bleche auf die Palette geladen und in das Maschinengehäuse eingefahren wurden, erfolgt der Schnitt automatisch. Aus Sicherheitsgründen kann der Maschinenbediener während des Schneidvorgangs nicht mehr in die Maschine eingreifen. Daraus ergibt sich die Notwendigkeit, den Schneidprozess zu überwachen, da Fehler beim Schneiden für den Maschinenbediener erst nachträglich ersichtlich sind. Ein häufiger Fehler beim Laserschneiden ist der Fehlschnitt [4]. Ein Fehlschnitt ist eine unvollständige Trennung des Materials und führt zu Ausschussteilen.

In der Vergangenheit wurden bereits Sensorsysteme entwickelt, um den Laserschneidprozess zu automatisieren, wie zum Beispiel die Sensorkomponente Active Speed Control (ASC) [3]. Mithilfe der ASC ist es möglich den Schneidprozess optisch zu überwachen. Dafür werden mit einer Kamera Bilder des Schnittspaltes aufgezeichnet und konventionell analysiert.

Das Ziel dieser Arbeit ist es, ein Machine Learning (ML)-Modell auf den Bildern der ASC zu trainieren, das Fehlschnitte auf einer Laserschneidmaschine erkennt. Das Modell soll für verschiedene Blechdicken anwendbar sein, die auf einer Maschine geschnitten werden können.

Die Arbeit gliedert sich wie folgt: In Abschnitt 2 wird der Stand der Technik näher erläutert. In Abschnitt 3 wird die

https://doi.org/10.37544/1436-4980-2024-05-82 https://www.inlibra.com/de/page/agb - Open Access - CC BY

Bild 1. Die „TruLaser 5030", eine Laserschneidmaschine. *Grafik: Trumpf [3]*

Problembeschreibung definiert. Es folgt ein Überblick über den Versuchsaufbau in Abschnitt 4 und die Ergebnisse der Versuche in Abschnitt 5. Der Beitrag endet mit einem Ausblick auf Folgearbeiten in Abschnitt 6.

2 Stand der Technik

Wie im vorhergehenden Abschnitt 1 beschrieben, liegt der Schwerpunkt dieser Arbeit auf der Erkennung von Fehlschnitten beim Laserschneiden, wobei in diesem Abschnitt auf einschlägige Literatur hierzu Bezug genommen wird. Der zunehmende Einsatz von ML-Methoden zur Prozessüberwachung beim Laserschneiden wird durch [5] belegt. Hier werden mehr als 20 Quellen zitiert, die sich auf verschiedene Anwendungen des maschinellen Lernens in der Laserbearbeitung beziehen. Diese unterscheiden sich jedoch in der Anwendung und der zugrunde liegenden Lasertechnologie.

Die Arbeiten [6–9] zeigen, dass eine Fehlschnitterkennung mit ML umgesetzt werden kann. Dazu werden in [6–8] visuelle Sensorsignale (Photodioden oder Kamerabilder) verwendet, welche die Eingabedaten für das Trainieren Neuronaler Netze bereitstellen. Die Laserschneidmaschine ist eine vielseitige Werkzeugmaschine [10]. Das bedeutet, dass auf einer Maschine verschiedene Prozesse geschnitten werden können. Die Prozesse unterscheiden sich dabei hinsichtlich der verwendeten Materialien, Blechdicken, Schneidgase und vieles mehr. Die Prozessvarianz kann zu Problemen bei der Verwendung von ML-Modellen führen, da diese nicht ohne weitere Anpassungen über mehrere Prozesse hinweg verallgemeinern können [11]. Für unsere Arbeit ist die Übertragung verschiedener Blechdicken relevant. Dieses Transfer-Problem wird auch in den Arbeiten [6, 8] aufgegriffen, jedoch nicht gänzlich gelöst. *Peghini et al.* [8] zeigt, dass eine Übertragung von einem Datensatz mit einer Prozessvariante auf einen anderen nicht möglich ist. Die Lösung des Autors besteht darin, den Datensatz mit den Prozessvarianten zu mischen, eine Lösung, die auch von den anderen Autoren vorgestellt wurde. Dieser Lösungsansatz führt zu einer Verringerung der Modellgenauigkeit im Vergleich zu den Einzelmodellen und wird daher als suboptimal eingestuft.

Zur Lösung der Übertragungsproblem von ML-Modellen wurde das Themenfeld Transferlernen eingeführt. In diesem Kontext werden die Begriffe Ursprungs- und Zieldomäne (engl. Source und Target) eingeführt [12, 13]. Im vorliegenden Problemfall wäre also eine Blechdicke von zum Beispiel 4 mm die Ursprungsdomäne und bei der Übertragung des Modells auf eine andere Blechdicke mit beispielsweise 6 mm die Zieldomäne.

Pan und *Yang* [13] unterteilen das Transferlernen in drei Kategorien: induktives Transferlernen, transduktives Transferlernen und unüberwachtes Transferlernen. Für jede Kategorie gibt es verschiedene Ansätze zur Bewältigung des Transferproblems. Die oben erwähnten Arbeiten [6, 8] fallen in die Kategorie des transduktiven Transferlernens, da hier die Modellanforderung in der Ursprungs- und Zieldomäne bestehen bleibt. Diese ist: die Klassifikation in gute und schlechte Schnitte. Der von [6, 8] gewählte Ansatz zum transduktiven Transferlernen ist der Ansatz des Instanztransfers. Nach [13] werden beim Instanztransfer die Gewichte des ursprünglichen Modells so verändert, dass die Quelle und das Ziel gleichermaßen abgebildet werden. Im Falle von [6, 8] geschieht dies durch das Zusammenfuhren des Datensatzes, also vor dem Training des Modells. Grundsätzlich können im Rahmen des Instanztransfers auch nachträglich die Gewichte des Modells auf die Zieldomäne angepasst werden.

Netzer et. al. [12] beschreibt den Einsatzbereich von Transferlernen im industriellen Anlagenbau. Der Fokus des industriellen Transferlearnings liegt jedoch auf der einmaligen Übertragung von Modellen auf verschiedene Maschinen oder Maschinenkomponenten. Unsere Arbeit unterscheidet sich dahingehend, dass der Transfer von Prozessen an jeder Maschine und zu jeder Zeit stattfinden muss.

In [13] wird ein alternativer Ansatz des Instanztransfers vorgeschlagen: der Feature-Repräsentations-Transfer. Dieser eignet sich besonders für Regressionsmodelle, bei denen nach einer Merkmalsraumrepräsentation gesucht wird, die sowohl für die Ursprungs- als auch für die Zieldomäne geeignet ist. Die Autoren *Zhiyi et al.* [14] zeigen einen Feature-Repräsentations-Transfer-

https://doi.org/10.37544/1436-4980-2024-05-82 https://www.inlibra.com/de/page/agb - Open Access - CC BY

Ansatz für die Adaption eines Autoencoders bei der Fehlerdiagnose von Lagern. Hierbei besteht der Transfer in der Übertragung eines Modells auf zwei verschiedene Maschinen. Zu diesem Zweck wird in *Zhiyi et al.* die Aktivierungsfunktion durch eine skalierte exponentielle Lineareinheit ersetzt und die Verlustfunktion (engl. loss function) durch einen nicht-negativen Nebenbedingungsterm modifiziert. Die Verwendung von Autoencoder Modellen ist eine verbreitete Wahl, um Anomalien zu detektieren und findet auch im Kontext der Werkzeugmaschinen Anwendung, wie zum Beispiel in [15].

3 Problembeschreibung

Nach der im vorherigen Kapitel erfolgten Analyse relevanter Methoden, wird in dieser Arbeit der Feature-Repräsentations-Transfer-Ansatz zur Problemlösung gewählt. Als Modellarchitektur wird ein Autoencoder gewählt, wofür ein Regressionsproblem definiert wird. Bei der Regression klassifiziert das Modell die Eingangsdaten nicht zu diskreten Labeln, sondern sagt eine kontinuierliche Variable voraus. Diese soll in unserem Fall (Schneid-) Qualitätslabel genannt werden. Je größer der Wert desto eher liegt ein Fehlschnitt vor.

Ziel der Arbeit ist es, mit dem trainierten Autoencoder eine Repräsentation der Daten zu finden, die alle Blechdicken abdeckt und in erster Linie die Aufgabe der Fehlschnitterkennung löst. Dem Autoencoder werden hierfür Bilddaten mit bekannten Qualitätslabeln zur Verfügung gestellt. Die Bilder bekommen ein zweites zusätzliches Label, welches die Dicke der geschnittenen Bleche angibt. Es werden vier verschiedene Baustahldicken betrachtet, diese sind: 4 mm, 6 mm, 8 mm und 10 mm.

Somit steht für diese Arbeit ein Datensatz $D = \{x_i, Y_i\}$, $i = 1...n$, für jede Blechdicke zur Verfügung. Durch das Multi-Label-Problem ist Y_i eine Menge von Labeln $Y_i = \{y_{1i}, y_{2i}\}$, wobei $y_1 \in \{0, 0.5, 1\}$ und $y_2 \in \{4, 6, 8, 10\}$. Die genaue Ermittlung von $y_1 \in \{0, 0.5, 1\}$ wird in Abschnitt 4 genauer erläutert.

Für jede Blechdicke existiert ein eigener Datensatz, wobei der Index für die Blechdickenzuordnung steht: D_4, D_6, D_8, D_{10}. Zu jedem Datensatz existiert ein unabhängiger Testdatensatz: T_4, T_6, T_8, T_{10}. Unabhängig bedeutet in diesem Fall, dass jeder Testdatensatz aus eigenständigen Schnittversuchen gewonnen wurde und nicht, wie häufig üblich zufällig, aus einem gemeinsamen Trainingsdatensatz D extrahiert wurde. Auf den Datensätzen $D \supseteq D_4, D_6, D_8, D_{10}$ wird das Modell trainiert. Dabei ist die Ausgabe eine Menge von Labeln, $\hat{Y}=\{\hat{y}_1, \hat{y}_2\}$. Die Ausgabe $\hat{y}_1$ wird für die Fehlschnitterkennung, beziehungsweise das Qualitätslabel verwendet und $\hat{y}_2$ für die Blechdickenzugehörigkeit.

4 Versuchsaufbau

Dieser Abschnitt beginnt mit einer Beschreibung der Datenerfassung und -verarbeitung. Es folgt eine Erläuterung des verwendeten Autoencoders.

4.1 Datenaufnahme und Labeling

Die Versuche wurden an der „TruLaser5030“ der Firma Trumpf im Brennschnitt durchgeführt. Der Schneidkopf ist mit dem ASC-Sensor ausgestattet. Dies ermöglicht die Aufnahme von Prozesslichtbildern beim Laserschneiden. Diese Bilder zeigen optische Emissionen eines bestimmten Wellenlängenbereichs, die

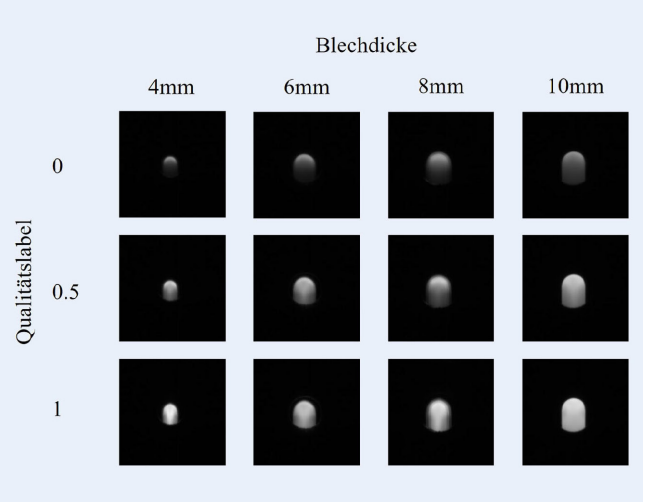

Bild 2. Einfluss von Qualitätswert und Blechdicke auf die Trainingsbilder. *Grafik: Trumpf*

beim Schneiden auftreten. Das **Bild 2** zeigt Bilder der Trainingsdaten. Vergleicht man die dargestellten Bilder, wird der Einfluss von Blechdicke und Qualitätswert deutlich. Wird einem Bild das Qualitätslabel 0 zugeordnet, handelt es sich um einen Gutschnitt. Wird dem Bild ein Qualitätslabel von 1 zugeordnet handelt es sich um einen Fehlschnitt. Das Qualitätslabel 0.5 zeigt einen Schnitt der nicht ideal, aber auch kein Fehlschnitt ist. Es muss mit einer Qualitätsminderung der Schneidproben gerechnet werden. Die dargestellten Bilder sind gemittelte Bilder eines Qualitätswerts und Dicke aus D. Es ist zu erkennen, dass die Bilder heller erscheinen, wenn ein Fehlschnitt auftritt. Es ist auch zu erkennen, dass das Prozesslicht in den Bildern mit zunehmender Blechdicke großflächiger erscheint.

Für die Datenaufnahme wurden Linearschnitte durchgeführt. Die Geschwindigkeit der Schnitte wurde kontinuierlich erhöht, wodurch ab einer gewissen Geschwindigkeit ein Fehlschnitt herbeigeführt wird. Dies liegt daran, dass nicht mehr genügend Energie in den Prozess eingebunden wird und das geschmolzene Material im Schnittspalt nicht mehr herausgeblasen werden kann.

Der weitestgehend automatisierte Labelprozess zur Generierung von wird in **Bild 3** gezeigt. Der obere Teil von Bild 3 zeigt die Prozesslichtkurve, das heißt die Bildintensität der während des Schneidens aufgenommenen Bilder. Darunter ist das geschnittene Blech zu sehen. Befindet sich Material in der Schnittfuge, werden die Bilder in diesem Bereich mit dem Qualitätslabel 1 (blau) gekennzeichnet. Bilder die sich im Bereich davor befinden und einen guten Schnitt darstellen werden mit dem Qualitätslabel 0, grün, bezeichnet. Da der Schnitt kontinuierlich schlechter wird, wurde im Übergangsbereich das Qualitätslabel 0.5, gelb, eingeführt. Zur Ermittlung des Übergangsbereichs wurde ein weiteres Merkmal zum Labeln eingeführt, welches in Bild 3 als hellblaue Kurve dargestellt ist. Die Tests haben gezeigt, dass die Intensität der Bilder über den Schnittverlauf im Gutschnittbereich schwanken. Die Schwankungen werden über die hellblaue Linie visualisiert. Der Übergangsbereich wird an dem Wendepunkt der hellblauen Linie, der dem Schnittabriss am nächsten ist, definiert (hier circa bei Bild 600).

Für jede Blechdicke wurden 2,000 Bilder pro Qualitätswert aufgenommen. Dies ergibt eine Gesamtgröße von 24,000 Bildern für den Datensatz. Für den Testdatensatz T wurden 200 Bilder

https://doi.org/10.37544/1436-4980-2024-05-82 https://www.inlibra.com/de/page/agb - Open Access - (cc) BY

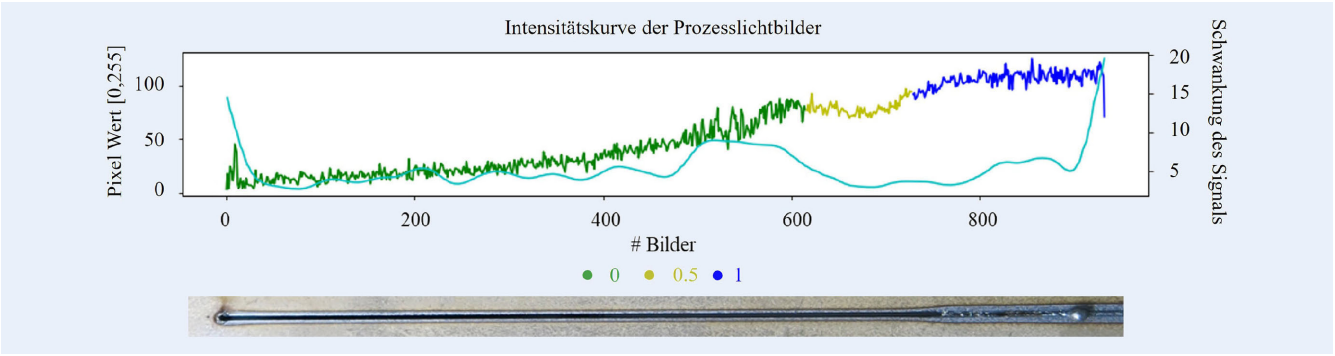

Bild 3. Darstellung des Label Verfahrens, oben: Prozesslicht der Bilder über den Verlauf eines Schnittes. Darstellung gefilterter Schwankung der Bildintensitäten in hellblau. Unten: Bild eines geschnittenen Blechs mit Schlackenaustritt. *Grafik: Trumpf*

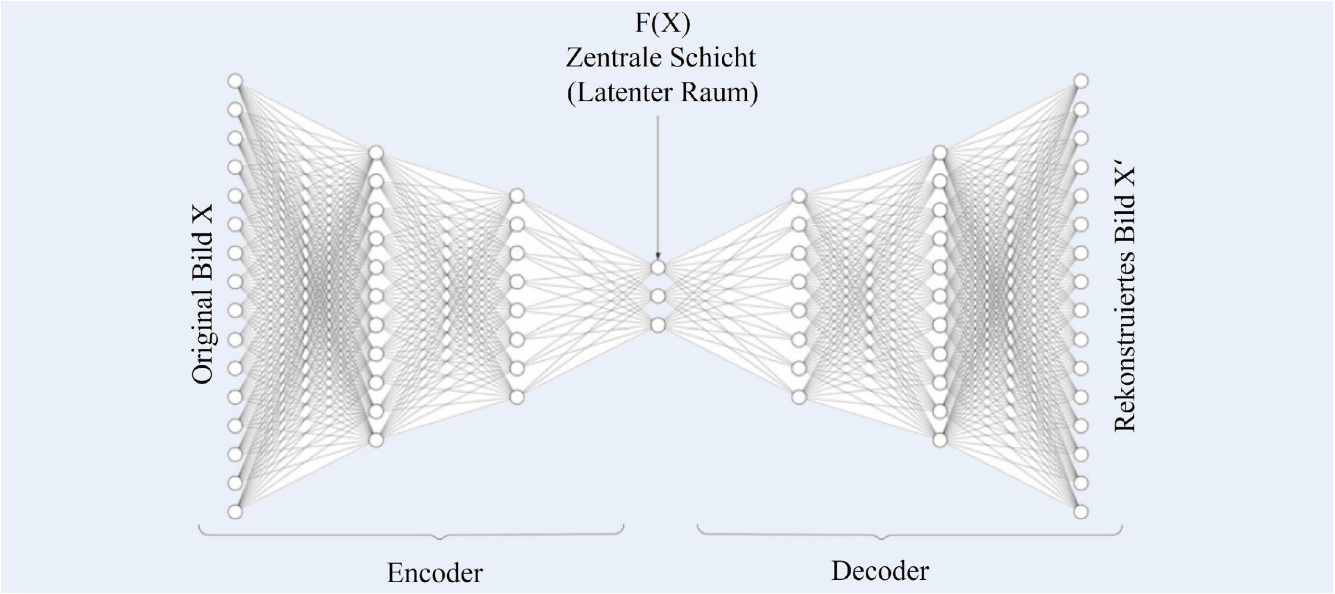

Bild 4. Die Architektur eines Autoencoders. *Grafik: Lanquillon und Schacht [16], Text angepasst*

pro Qualitätswert und Blechdicke aufgenommen, was eine Gesamtgröße von 2,400 Bildern für T ergibt.

4.2 Der Autoencoder

Ein Autoencoder basiert auf dem Prinzip der Faltungsnetze (engl. convolutional neural networks) und wird auch als Encoder-Decoder-Architektur bezeichnet [16]. Der Name leitet sich von der in **Bild 4** dargestellten Struktur ab. Die Hauptkomponenten des Autoencoders sind:

- Der Encoder ist die erste Komponente des Autoencoders. Hier wird die Eingangsvariable X in eine komprimierte Darstellung $F(X)$ konvertiert. $F(X)$ wird auch als latenter Raum bezeichnet. Diese Repräsentation dient als Eingangsvariable für den Decoder. [16, 17]
- Die zentrale Schicht, beziehungsweise der latente Raum $F(X)$ des Autoencoders, ist durch eine Schicht gekennzeichnet, die deutlich kleiner als die Eingangs- oder Ausgangsschicht ist [16]. Dies bewirkt, dass der Autoencoder gezwungen ist möglichst aussagekräftige latente Merkmale zu lernen, um die Daten neu zu repräsentieren [17].
- Der Decoder verwendet die Repräsentation $F(X)$ des Encoders und wandelt sie in die Ausgabe $\hat{y}$ um. Die Ausgabe $\hat{y}$ hängt von der zu lösenden Aufgabe ab, kann aber zum Beispiel das möglichst originalgetreu rekonstruierte Bild X' des Eingangsbildes X sein. Die Architektur des Decoders entspricht üblicherweise der des Encoders [16].

Je nach Aufgabenstellung kann nur der Encoder-Teil eines Autoencoders verwendet werden, zum Beispiel zur Merkmalsreduktion. Ebenfalls kann nur der Decoder verwendet werden, zum Beispiel zur synthetischen Datengenerierung [16].

In dem hier vorgestellten Fall wird der gesamte Autoencoder trainiert, aber nur der Encoder wird im weiteren Verlauf verwendet. Eine weitere Besonderheit des hier verwendeten Autoencoders ist die angepasste Verlustfunktion, welche es ermöglicht, die Anordnung im latenten Raum $F(X)$ vorzugeben. Bei dem hier vorgestellten Problemfall handelt es sich somit um ein überwachtes Lernproblem, im Gegensatz zu den üblichen Autoencodern, die dem halb- oder unüberwachten Lernen zugerechnet werden [16]. Die Verlustfunktion wird für die Modellanpassung während des Trainingsprozesses benötigt und basiert in diesem Fall auf dem mittleren quadratischen Fehler (engl. Mean Squared Error) [16]. Sie lässt sich in die folgenden drei Terme aufteilen:

$$L_1 = \frac{1}{n}\sum_{i=1}^{n}(x_i - x_i')^2, \qquad (1)$$

https://doi.org/10.37544/1436-4980-2024-05-82 https://www.inlibra.com/de/page/agb - Open Access - (cc) BY

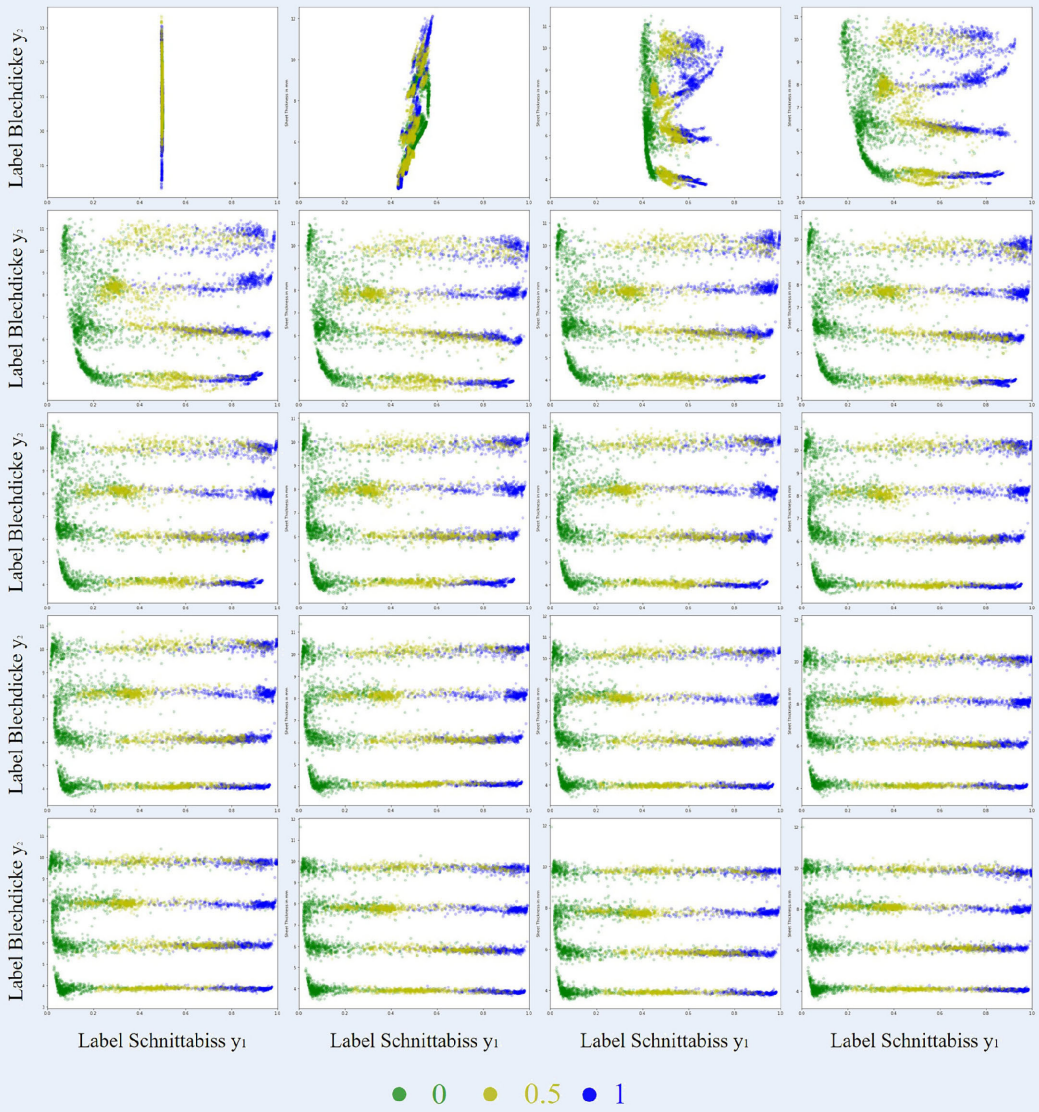

Bild 5. Entwicklung des Latenten Feature Raumes über 20 Trainings Epochen des trainierten Encoders. Das Modell lernt die Blechdicken auf der y-Achse zu trennen und skaliert die Schneidqualität auf der x-Achse. *Grafik: Trumpf*

$$L_2 = \frac{1}{n}\sum_{i=1}^{n}\left(y_{1i} - \hat{y}_{1i}\right)^2, \tag{2}$$

$$L_3 = \frac{1}{n}\sum_{i=1}^{n}\left(y_{2i} - \hat{y}_{2i}\right)^2. \tag{3}$$

Die Verlustfunktion L1 repräsentiert den Rekonstruktionsfehler der Bilder, die Verlustfunktion L2 den Fehler der Qualitäts-Vorhersage, und L3 den der Blechdicken-Vorhersage. Die Verlustwerte L1, L2, L3 aus den Gleichungen (1)-(3) werden mit einem Faktor gewichtet und zu einer Verlustfunktion aufaddiert:

$$L = a \cdot L_1 + b \cdot L_2 + c \cdot L_3. \tag{4}$$

Die nicht-negativen Gewichtungsfaktoren a, b und c aus Gleichung (4) werden experimentell ermittelt.

5 Ergebnisse

Für das verwendete Modell wurde, wie in Kapitel 4 beschrieben, ein Autoencoder implementiert und trainiert. Der Encoder besteht aus drei Faltungsschichten mit einer Filteranzahl von vier, acht und 16. Der Decoder besteht aus zwei Faltungsschichten mit einer Filteranzahl von vier und eins. Die Filtergröße beträgt jeweils drei. Die zentrale Schicht besteht aus zwei Neuronen, ein Neuron für die Kodierung der Blechdicke und ein Neuron für die Kodierung der Qualität. Für das Training wird ein Adam Optimizer mit der Standard-Lernrate von 0.001 verwendet. Die Verlustfunktion wird, durch die in Gleichung (4) beschriebene angepasste Verlustfunktion ersetzt, dabei sind die gewählten Gewichtungsfaktoren a = 1, b = 0.02 und c = 0.1. Das Modell wird über 20 Epochen trainiert und eine Batch-Größe von 32 wurde gewählt. Die Parameter des Autoencoders wurden mithilfe eines Parameter-Tunings, also der automatisierten Suche performanter Parameter, ermittelt.

https://doi.org/10.37544/1436-4980-2024-05-82 https://www.inlibra.com/de/page/agb - Open Access - (cc) BY

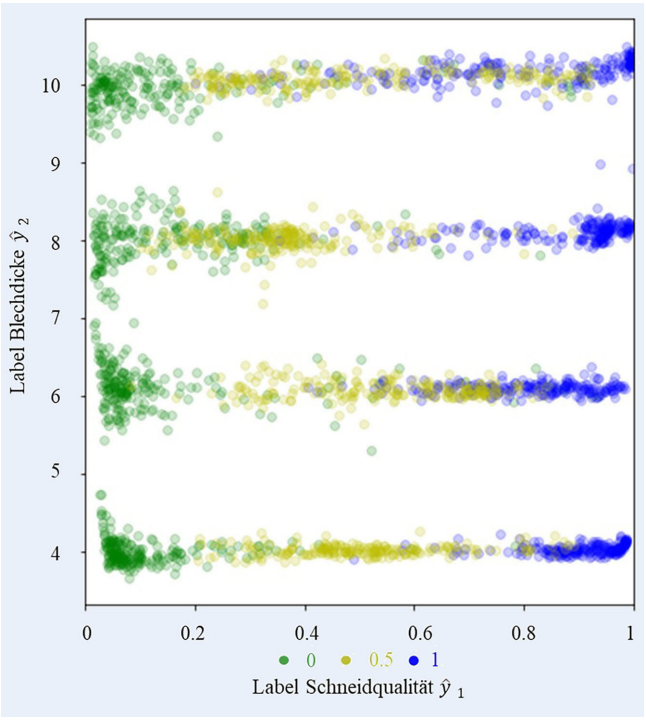

Bild 6. Visualisierung der Modelvorhersage auf einem unabhängigen Testdatensatz *T*. Das Modell ist in der Lage, die Blechdicke und die Schneidqualität im Latenten Raum zu trennen. *Grafik: Trumpf*

Das **Bild 5** zeigt die Entwicklung des latenten Raums $F(X)$ des Encoders über den Trainingsprozess. Es sind insgesamt zwanzig Einzelbilder zu sehen. Jedes Bild gehört zu einer Epoche des Trainingsprozesses. Die Bilder wurden anschließend mit den bekannten Qualitätslabeln, aus dem Labelverfahren aus Abschnitt 4, eingefärbt. Dies zeigt, dass der Autoencoder eine Merkmalsraumrepräsentation $F(X)$ erstellen kann, in der sowohl das Qualitätslabel als auch die Blechdicke erkannt wird. Obwohl die Blechdickenerkennung kein Ziel dieser Arbeit ist, ist es sinnvoll diese durch das Multi-Klassen Problem visualisieren zu können. Dadurch kann sichergestellt werden, dass der Parameter Blechdicke vom Modell sinnvoll verwertet wird.

Der Encoder wurde anhand des unabhängigen Testdatensatzes getestet, und die Ergebnisse sind in **Bild 6** dargestellt. Auf der y-Achse ist die vorhergesagte Blechdicke $\hat{y}_2 \in \mathrm{R}$ aufgetragen, auf der x-Achse die das Qualitätslabel $\hat{y}_1 \in [0, 1]$. Die Bilder werden zur Überprüfung entsprechend ihrer bekannten Qualitätswertbezeichnung aus dem Labelprozess (vgl. Abschnitt 4) eingefärbt.

Für die Verwendung auf der Maschine muss für die Fehlschnitterkennung ein Schwellwert eingeführt werden. Auf Basis der Erkenntnisse von Bild 6 ist ein Schwellwert für den Fehlschnitt von 0.6 sinnvoll.

Es fällt auf, dass die vorhergesagte Blechdicke leicht nach oben und unten schwankt. Zum Beispiel kann ein Bild mit einer Blechdicke von 4 mm auch für eine Blechdicke von 4.5 mm vorhergesagt werden. Außerdem ist auffällig, dass mit zunehmender Blechdicke das Qualitätslabel für den Fehlschnitt nach rechts driftet, also bei niedrigeren y_1-Werten auftritt. Dieser Effekt konnte auch bei der Datenaufnahme festgestellt werden. So findet beim Schneiden von dünneren Blechen ein abrupter Schnittabriss statt. Bei dickerem Blech entspricht diese eher einer zunehmenden Verschlackung des Schnittspaltes, also einem schleichendem Schnittabrisses. Dieses Verhalten erklärt auch, warum die Übertragung unterschiedlicher Blechdicken für ein ML-Modell mit Problemen verbunden ist.

6 Zusammenfassung und Ausblick

Im Gegensatz zum Stand der Technik wurde ein Ansatz vorgestellt, bei dem anstelle eines Klassifikationsproblems, eine Regression genutzt wurde, um Fehlschnitte beim Laserschneiden zu ermitteln. Zur Problemlösung wurde ein Autoencoder verwendet. Besonders hervorzuheben ist, dass es gelungen ist, ein Modell für verschiedene Blechdicken zu generalisieren. Der gewählte Autoencoder zeigt eine korrekte Einteilung der Qualitätslabel im latenten Raum des Encoders, sodass dieser für die Anwendung auf der Maschine zum Einsatz kommen kann. Zur Erkennung des Fehlschnitts können Schwellwerte eingeführt werden.

Die vorgestellte Methode kann auch für andere Prozessänderungen verwendet werden, zum Beispiel für das Schneiden mit unterschiedlichen Gasarten oder Düsen. Eine Untersuchung des Kompromisses zwischen der Genauigkeit und der Anzahl der Prozessänderungen wäre interessant. Es sollte untersucht werden, ob es besser ist, so viele Prozessänderungen wie möglich mit einem Modell abzudecken oder mehrere Modelle für gebündelte Prozessänderungen zu verwenden. Denkbar wäre, dem Modell die Parameter beziehungsweise die Prozessinformationen als Input zur Verfügung zu stellen. Diese müssen aber nicht zwangsweise auch als Ausgabe zur Verfügung stehen.

DANKSAGUNG

Das Forschungs- und Entwicklungsprojekt „Selbstlernende Werkzeugmaschinen für eine hocheffiziente Produktion (AutoLern)“ wurde durch das Bundesministerium für Bildung und Forschung (BMBF) im Programm „Zukunft der Wertschöpfung – Forschung zu Produktion, Dienstleistung und Arbeit“ (Förderkennzeichen 02P20A020 bis 02P20A026) gefördert und vom Projektträger Karlsruhe (PTKA) betreut. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den Autoren. Die Autoren möchten allen unterstützenden Personen danken. Für den aufgeführten Text, wurde DeepL Write eingesetzt, um den Schreibstil zu verbessern.

Literatur

[1] Buchfink, G.; Leibinger-Kammuller, N.: Faszination Blech, 2021.
[2] Institut für Produktionstechnik wbk. Maschinen, Anlagen und Prozessautomatisierung (MAP). Internet: https://www.wbk.kit.edu/maschinen-anlagen-und-prozessautomatisierung.php. Zugriff am 02.03.2023
[3] Trumpf. Active speed control: Durchschaut den prozess, regelt den schnitt. Internet: https://www.trumpf.com/deDE/produkte/maschinen systeme/2d-laserschneidmaschinen/webspecial-autonomes-laser-schneiden/webspecialautonomeslaserschneiden/produkte-und-funktionen/. Zugriff am 29.12.2023
[4] Pro-Lean. Häufige Fehler beim Laserschneidprozess. Internet: https://proleantech.com/de/thecommon-defects-in-the-laser-cutting-process/. Zugriff am 02.03.2023
[5] Mills, B.; Grant–Jacob, J. A.: Lasers that learn: The interface of laser machining and machine learning, 2020, https://doi.org/10.1049/ote2.12039

https://doi.org/10.37544/1436-4980-2024-05-82 https://www.inlibra.com/de/page/agb - Open Access - (cc) BY

[6] Adelmann, B.; Schleier, M.; Hellmann R.: Laser cut interruption detection from small images by using convolutional neural network, Sensors (Basel, Switzerland), vol. 21, no. 2, 2021, https://doi.org/10.3390/s21020655
[7] Adelmann, B.; Hellmann R.: Simultaneous burr and cut interruption detection during laser cutting with neural networks., https://doi.org/10.3390/s21175831
[8] Peghini, N.; et al.: Real-time cross-dataset quality production assessment in industrial laser cutting machines, ICPR International Workshops and Challenges, vol. 12664, pp. 490–505, https://doi.org/10.1007/978–3–030–68799–1 36
[9] Leiner, K.; et al.: Cut interruption detection in the laser cutting process using ROCKET on audio signals, in 2023 IEEE 21st International Conference on Industrial Informatics (INDIN). IEEE, Jul. 2023,https://doi.org/10.1109/INDIN51400.2023.10218267
[10] Karanfil, U.; Yalcin, U.: Real-time monitoring of high-power fibre-laser cutting for different types of materials, Ukrainian Journal of Physical Optics, vol. 20, no. 2, pp. 60–72, 2019, http://dx.doi.org/10.3116/16091833/20/2/72/2019
[11] Netzer, M.; Alexander, P.; Gönnheimer, P.; Fleischer, J.: Industrielles Transferlernen: Der Schlüssel zur Übertragung von Künstlicher Intelligenz. Zeitschrift Für Wirtschaftlichen Fabrikbetrieb. 117, 597–599 (2022,9), https://doi.org/10.1515/zwf-2022–1109
[12] Netzer, D.; Michelberger, J.; Puchta, A., Verl, P. & Fleischer, P. : Introduction of an industrial transfer learning use case systematization for machine tools. Procedia CIRP. 120 pp. 398–403 (2023), https://doi.org/10.1016/j.procir.2023.09.009
[13] Pan, S. J.; Yang, Q.: A survey on transfer learning, IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345–1359, 2010, https://doi.org/10.1109/TKDE.2009.191
[14] Zhiyi, H.; et al.: Transfer fault diagnosis of bearing installed in different machines using enhanced deep autoencoder, Measurement, vol. 152, no. 3, p. 107393, 202, https:// doi.org/10.1016/j.measurement.2019.107393
[15] Netzer, M.; Palenga, Y.; Fleischer, J.: Machine tool process monitoring by segmented timeseries anomaly detection using subprocess-specific thresholds. Production Engineering. 16 pp. 597–606 (2022), https://link.springer.com/article/10.1007/s11740-022-01120-3
[16] Lanquillon, C.; Schacht, S.: Knowledge Science – Grundlagen. Wiesbaden: Springer Fachmedien Wiesbaden, 2023, https://doi.org/10.1007/978–3–658–41689–8
[17] Ye, A.; Wang, Z.: Modern deep learning for tabular data: Novel approaches to common modeling problems. Berkeley: Apress, 2023, https://doi.org/10.1007/978-1-4842-8692-0

Kathrin Leiner
kathrin.leiner@trumpf.com
Foto: Autorin

Trumpf Werkzeugmaschinen SE + Co. KG
Johann-Maus-Str. 2, 71254 Ditzingen
www.trumpf.com

Daniel Wall
dwall@point-8.de

Jan Kühlem
jkuehlem@point-8.de

Point 8 GmbH
Rheinlanddamm 201, 44139 Dortmund
https://point-8.de

Prof. Dr.-Ing. Marco Huber
marco.huber@ipa.fraunhofer.de

Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA
Nobelstr. 12, 70569 Stuttgart
www.ipa.fraunhofer.de

LIZENZ

Dieser Fachaufsatz steht unter der Lizenz Creative Commons Namensnennung 4.0 International (CC BY 4.0)

https://doi.org/10.37544/1436-4980-2024-05-82 https://www.inlibra.com/de/page/agb - Open Access - (cc) BY