19. AALE-Konferenz
Luxemburg, 08.-10. März. 2023

Klassifizierung von Altholz durch FD-FLIM Bilder mit neuronalen Netzen

M. Dietlmeier[1], A. S. Rajan[1], N. Leiter[1], M. Wohlschläger[1], M. Versen[1]

Zusammenfassung

Holz ist eine der wichtigsten erneuerbaren Ressourcen und kann recycelt werden, wobei derzeit nur unbehandeltes Holz in mehreren Lebenszyklen verwendet werden kann. Allerdings werden in Altholzsortieranlagen meist behandeltes und unbehandeltes Altholz gemischt angeliefert und es gibt keine automatische Klassifizierung zur sortenreinen Sortierung. Frühere Studien zeigen, dass behandelte und unbehandelte Holzproben mit Hilfe von bildgebender Fluoreszenzabklingzeit Mikroskopie unterschieden werden können. Um den Prozess der Klassifizierung von Altholz anhand von Fluoreszenzabklingzeitbildern zu automatisieren, werden zwei neuronale Netzwerke (NN) evaluiert. Das erste NN ist ein MLP (Multi Layer Perceptron), das das Altholz anhand von Erwartungswert und Standardabweichung eines Fluoreszenzabklingzeitbildes klassifiziert. Das zweite untersuchte NN ist ein CNN (Convolutional Neural Network), das das Holz direkt anhand der Fluoreszenzabklingzeitbilder klassifiziert. Beide Netzwerke sind in Python implementiert, um die spätere Verwendung in einem automatischen Klassifizierungs- und Sortierprozess zu erleichtern. Die Evaluation der NN zeigt vielversprechende Ergebnisse zur automatisierten Klassifikation von Altholz.

Stichwörter

FD-FLIM, MLP, CNN, Altholzklassen, Erneuerbare Ressourcen

1 Einleitung

Der Klimawandel, eine der größten Herausforderungen, mit denen die Welt derzeit konfrontiert ist, ist die Veränderung der globalen Klimamuster aufgrund der Emission von Treibhausgasen wie Kohlendioxid. Inzwischen gibt es mehrere Ansätze zur Reduktion des CO_2-Ausstoßes und somit zur Entspannung der Klimakrise [1]. Die Verringerung der CO_2-Emissionen erfordert die Wiederverwendung von Materialien, insbesondere Holz, über mehrere Lebenszyklen hinweg, welche ein hohes CO_2-Bindungspotenzial aufweisen. Obwohl die Verbrennung von Holz nach einem Zyklus CO_2-freie Energie liefert, ermöglicht die Verwendung von Altholz in mehreren Kaskaden eine längere Bindung von CO_2 und wirkt sich positiv auf die aktuelle Klimasituation aus. Eine energetische Nutzung von Holz erscheint nur dann sinnvoll, wenn Fremdstoffe oder Verunreinigungen, wie beispielsweise Lacke, Lasuren oder Imprägnierungen im Holz oder auf der Oberfläche vorhanden sind, die bei der stofflichen Verwertung einen Schadstoffeintrag im Recyling-Produkt zur Folge hätten [2].

Zur korrekten Einordnung von Altholz werden vier Klassen nach der deutschen Altholzverordnung betrachtet: A1 (mechanisch bearbeitetes Holz), A2 (nicht halogenorganisch behandelte Hölzer), A3 (halogenorganisch behandelte Hölzer) und A4 (Hölzer mit Holzschutzmittel behandelt oder imprägniert). Derzeit werden die Holzklassen durch eine visuelle Inspektion von Fachleuten sortiert, was zu Fehlklassifizierungen führt, da einige Verunreinigungen visuell kaum erkennbar sind. Darüber hinaus ist die manuelle Sortierung zeitaufwändig und kann nicht automatisiert werden. Um Fehlklassifizierungen zu vermeiden und eine echtzeitnahe Klassifikation und Sortierung anzustreben, sind alternative Methoden zur Sortierung von Altholz erforderlich. Zu den erforschten optischen Methoden gehören die Nah-Infrarot-Spektroskopie (NIRS), die Ionenmobilitätsspektrometrie (IMS) und die Röntgenfluoreszenzspektroskopie (XFL). Allerdings weisen alle Methoden Nachteile zur Sortierung von Altholz auf. Bei der NIR-Spektroskopie erschwert die Feuchtigkeit im Holz die Klassifikation, da Wasser seine Absorptionsbanden überlagert, die zur Charakterisierung von Holz verwendet werden [3]. Mit der IMS-Methode können zwar Konservierungsmittel nachgewiesen werden, aber die Messung einer gemischten

[1] TH Rosenheim

Open Access. © 2023 **M. Dietlmeier**, A. S. Rajan1, N. Leiter1, M. Wohlschläger1, M. Versen1
This work is licensed under the Creative Commons Attribution 4.0 License. +

Holzprobe ist zu kompliziert, um automatisiert zu werden [3]. Mit der XFL-Methode können Materialien mit hoher Ordnungszahl erkannt werden können. Sie eignet sich zum Nachweis von Schwermetallen, aber nur bedingt vom Nachweis von Kunststoffen [3].

Eine praktikable Lösung für die Fragestellung der automatisierten Klassifizierung von Altholz kann die Fluoreszenzabklingzeit Mikroskopie (Frequency-Domain Fluorescence Lifetime Imaging Microscopy; FD-FLIM) sein. Die in [4] gezeigten Untersuchungen zeigen, dass eine Unterscheidung der Althölzer durch deren charakteristische Fluoreszenzabklingzeiten mit einem Entscheidungsbaum möglich ist. Um die Klassifikation von Althölzern weiter zu automatisieren, werden in diesem Beitrag zwei neuronale Netzwerke (NN) untersucht. Das erste NN basiert auf einem MLP, zur Klassifizierung der Althölzern anhand der Erwartungswerte und der Standardabweichungen der Fluoreszenzabklingzeitbilder. Das zweite NN verwendet ein vortrainiertes CNN, dass die Holzkategorie anhand der gegebenen Fluoreszenzabklingzeitbilder klassifiziert.

2 Theorie

2.1 Messung der Fluoreszenzabklingzeit

Die Fluoreszenzabklingzeit (Fluorescence lifetime; FLT) gibt an, wie lange es dauert, bis die Fluoreszenzintensität der angeregten Moleküle auf 1/e (etwa 36,8 %) der maximalen Fluoreszenzintensität abfällt. Die FLT kann im Zeit- oder Frequenzbereich gemessen werden [5]. FLT-Messungen im Zeitbereich werden durch eine zeitkorrelierte Einzelphotonenzählung (Time-correlated single photon counting; TCSPC) realisiert. Durch den integrierenden Prozess ist die Messung der Fluoreszenzabklingzeit im Zeitbereich jedoch nur schwer in Echtzeit umsetzbar. Im Frequenzbereich wird die Anregung durch ein sinusförmig oder rechteckmoduliertes Lichtsignal erreicht. Das resultierende Fluoreszenzsignal hat die gleiche Frequenz wie die Lichtquelle, ist jedoch phasenverschoben, in seiner Amplitude gedämpft und Gleichwert verschoben. Durch die Messung der Phasenverschiebung, der Amplitudendämpfung und der Gleichwertverschiebung können eine modulations- und phasenabhängige Fluoreszenzabklingzeit berechnet werden [6] [7]. Basierend auf der Fluoreszenzabklingzeitmessung im Frequenzbereich hat die Excelitas PCO GmbH eine Kamera entwickelt, die pixelweise die Phasenverschiebung, Amplitudendämpfung und Gleichwertverschiebung bestimmt. Aus den drei pixelweise gemessenen Parametern lässt sich ein Bild aus modulations- und phasenabhängiger FLT sowie der mittleren Fluoreszenzintensität rekonstruieren, aus welchen sich ortsabhängige Veränderungen der FLT ermitteln lassen.

2.2 Neuronale Netzwerke

Es gibt verschiedene Arten von NN, die die gleiche Grundstruktur haben. Jedes NN hat eine Eingabeschicht (Input-Layer), versteckte Schichten (Hidden-Layers) und eine Ausgabeschicht (Output-Layer). Die gemessenen Daten werden an den Input-Layer übergeben, welcher entsprechend der Datengröße konfiguriert werden muss. Wird beispielsweise ein Bild klassifiziert, muss die Größe der Eingabeschicht zur Anzahl der Pixel in der Breite und Länge sowie zu den Kanälen passen. Die Konfiguration erfolgt durch die Definition der Knotenpunkte oder Neuronen. Diese Neuronen sind im Input-Layer, den Hidden-Layers und dem Output-Layer vorhanden und unterscheiden sich primär durch ihre Aktivierungsfunktion und ihre Initialisierung. Die Neuronen erhalten eine Eingabe, die mit einem Gewicht multipliziert und aufsummiert wird. Nach der Summation wird eine Verstärkung (Bias) hinzugefügt, sodass die Neuronen auch dann aktiviert werden, wenn die Eingabe Null ist.

Eine häufig verwendete Aktivierungsfunktion ist ReLU (Rectified Linear Unit). Sie wird bevorzugt verwendet wegen ihrer effizienten Berechnung, der guten Optimierung mit stochastischem Gradientenabstiegsverfahren und der Fähigkeit, einen echten Nullwert auszugeben [6]. Die Initialisierung erfolgt häufig auf Basis einer zufälligen Normalverteilung, jedoch gibt es auch gezielte Intialisierungsmethoden (z.B. [8]).

Die Hidden-Layer sind bei jedem NN unterschiedlich. Bei „tiefen“ NN kann ein Netzwerk über hundert Hidden-Layer haben. Im Allgemeinen führt eine höhere Anzahl an Hidden-Layern zu einer höheren Anzahl extrahierter Merkmale, sodass mehr Gewichte angepasst werden können. Dies führt jedoch nicht nur zur Verbesserung von Netzwerken, denn neben dem erhöhten rechnerischen Zeitaufwand kann es auch zu einem Gradientenabstieg im Lernprozess kommen, da spätere Layer ihre interne Gewichtung in Bezug auf die vorangegangene Layer anpassen. Das bedeutet beispielsweise,

Open Access. © 2023 **M. Dietlmeier**, A. S. Rajan1, N. Leiter1, M. Wohlschläger1, M. Versen1
This work is licensed under the Creative Commons Attribution 4.0 License. +

wenn der Gradient bereits im ersten Layer klein ist, wird er im letzten Layer gegen Null gehen, was wiederum bedeutet, dass das NN die Gewichte nicht aktualisiert und dass es zu einem Fehlschlag im Lernprozess kommt.

Am Ende eines NN steht ein Output-Layer, der die gewünschten Ergebnisse ausgibt. Im Output-Layer ist die Anzahl der Neuronen vom Anwendungsfall abhängig. Bei einer binären Klassifikation können z.B. ein oder zwei Neuronen verwendet werden. Die Neuronen können durch verschiedene Aktivierungsfunktionen aktiviert werden, wobei normalerweise die SoftMax-Funktion verwendet wird, die die Wahrscheinlichkeiten im Output-Layer berechnet. Wichtig ist, dass die Wahrscheinlichkeiten proportional zur relativen Größe der einzelnen Werte aus den Hidden-Layern sind [9 bis 12].

2.2.1 „Multilayer Perceptron“

Das MLP wurde entwickelt, um nichtlineare Zuordnungen zu bewältigen. Es wird die allgemeine Struktur eines NN verwendet, was bedeutet, dass das MLP aus mehr als einem Hidden-Layer besteht, welche mehr als ein Neuron enthalten. Die Struktur, die in MLPs verwendet wird, wird als „Feed-Forward Artificial Neural Network“ (ANN) bezeichnet, da jede Schicht ihre Berechnung an die nächste Schicht weitergibt. Um die Gewichte tatsächlich zu trainieren, ist eine Rückkopplung zwischen den Schichten erforderlich, wodurch eine iterative Anpassung der Gewichte in den Neuronen stattfindet. Die Optimierung erfolgt durch Minimierung der Kosten-/Verlustfunktion, wobei diese z.B. der mittlere quadratische Fehler ist (siehe Gleichung (1)).

$$C(y,\, p) = \sum_{i=1}^{N} (y_i - p_i)^2 \tag{1}$$

Die Aufgabe des „Solvers“ besteht darin, den Gradienten dieser Kostenfunktion zu berechnen, d.h. die partielle Ableitung in Bezug auf die Gewichte und die Verzerrungen. Mit dieser Berechnung macht das NN einen Schritt in Richtung des lokalen/ globalen Minimums, wobei die Größe dieses Schrittes durch die Lernrate definiert wird. Dieser iterative Prozess wird so lange wiederholt, bis der Gradient für jedes Input-Output-Paar gegen einen definierten Schwellwert konvergiert [11].

2.2.2 „Convolutional Neural Networks“

In CNNs umfassen die Hidden-Layer nicht nur Layer mit Neuronen, sondern auch Faltungs- und Pooling-Layer. Das Prinzip einer Faltungs-Layer besteht darin, eine Faltungsmatrix über ein Bild zu bewegen, um Merkmale zu extrahieren und anschließend die Bildgröße zu verringern. Besteht kein Bedarf an einer Verkleinerung, kann eine Schicht mit Nullen um das Originalbild herum eingefügt werden. Die Größen und Werten der Faltungsmatrix vom Ziel der Faltung ab. Sie bewegt sich über das Bild in Abhängigkeit vom Stride-Parameter, der in der Regel einen Wert von 1 hat, was bedeutet, dass sich die Faltungsmatrix immer um ein Pixel nach rechts bewegt, bis sie das Ende einer Zeile erreicht, und sich dann um ein Pixel nach unten und links im Bild verschiebt, um den Vorgang in der nächsten Zeile zu wiederholen, bis dass das gesamte Bild durchlaufen ist. Nach mehreren Faltungs-Layern wird ein Pooling-Layer hinzugefügt, welcher die Größe des gefalteten Bildes reduziert und potenziell dominante Merkmale extrahiert. Beim Pooling wird eine Matrix mit einer vordefinierten Größe, z. B. 2x2, auf das gefaltete Bild angewendet. Je nach Art der Pooling-Layer wird nur der Maximal- oder der Durchschnittswert in einem Pixel gespeichert. Der letzte Schritt vor dem Eintritt in ein Feed-Forward-NN wie das MLP ist eine Flatten-Layer, der die Ausgabe der letzten Pooling-Layer in einen Vektor umwandelt. Für die meisten fortgeschrittenen Bilderkennungsaufgaben wird ein einfaches Netzwerk keine guten Ergebnisse liefern, da es zu wenige Faltungsschichten gibt, um die Merkmale zu extrahieren. Aus diesem Grund müssen die meisten vortrainierten Netzwerke individuell mit Input-, Hidden und/oder Output-Layern angepasst werden [13].

2.3 Konfusionsmatrix und Kennzahlen

Der Standard für die Bewertung von NNs ist eine Konfusionsmatrix (siehe Bild 1). Aus ihr können schnell und einfach Aussagen getroffen werden über richtig und Falsch klassifizierte Proben. Da es in diesem Projekt zwei Klassen (A1 und A2-A4) unterschieden werden, kann eine binäre Konfusionsmatrix verwendet werden. In dieser binären Konfusionsmatrix wird das wahre Label gegen die Vermutung des Netzwerkes für jedes getestete Bild aufgetragen. Demzufolge gibt es vier verschiedene Kategorien: „True Negative“ (TN), „True Positive“ (TP), „False Negative“ (FN) und „False Positive“ (FP). In diesem sowie vielen weiteren Klassifikationsfällen ist die Unterscheidung zwischen (TP) und (TN) willkürlich gewählt, da es keine Wahrheitsunterschiede bei den beiden Klassen gibt. Da die Berechnung der Kennzahlen jedoch auf

Open Access. © 2023 **M. Dietlmeier**, A. S. Rajan1, N. Leiter1, M. Wohlschläger1, M. Versen1
This work is licensed under the Creative Commons Attribution 4.0 License. +

den Inhalten der Konfusionsmatrix basiert, ist eine durchgehend einheitliche Bewertung wichtig. Da die Konfusionsmatrix automatisch von einer Funktion der „sci-kit learn“ Bibliothek [14] erstellt wird, müssen lediglich die Label (A1, A2-4) in der gleichen Reihenfolge an die Funktion übergeben werden.

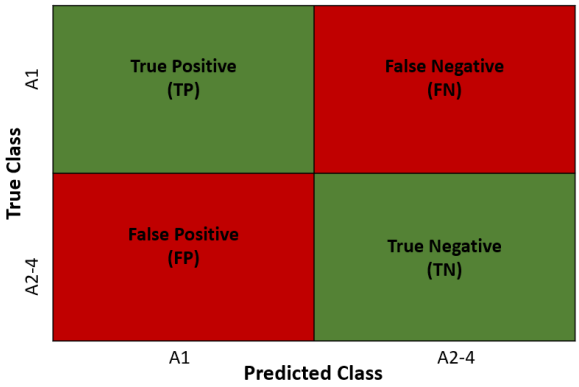

Bild 1: Beispiel einer Konfusionsmatrix.

Die in diesem Projekt verwendeten Kennzahlen sind: „Accuracy“, „Recall“, „Precision“ und „F1-score“ (siehe Gleichungen 2-5). Die „Accuracy“ (Genauigkeit) gibt den Anteil an korrekten Vorhersagen über allen Vorhersagen an. Somit ist diese Metrik ein guter genereller Indikator für die Güte eines Netzwerks, jedoch kann die Kennzahl ohne weitere Informationen auch irreführend sein. Der „Recall“ (Sensitivität) stellt den Anteil von korrekt identifizierten Vorhersagen für die „wahre“ Klasse über allen Proben dieser Klasse dar. Die „Precision“ (Präzision) ist ähnlich zur Sensitivität, spiegelt aber das Verhältnis zwischen korrekten Vorhersagen für die „wahre“ Klasse über allen Vorhersagen für diese Klasse. Da Proben die fälschlicherweise der Klasse A1 zugeordnet wurden ein großes Problem bei der Verwertung darstellen, sollte die Anzahl dieser Fehlklassifizierungen so gering wie möglich sein und demzufolge die Präzision hoch. Der F1-Score ist eine Kombination aus Sensitivität und Präzision. Somit vereint sie zwei wichtige Metriken und wird als eine der wichtigsten Kennzahlen zur Bestimmung der Güte eines NN angesehen [15].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$\mathrm{F1} - score = \frac{2 * (\mathrm{Recall} * \mathrm{Precision})}{Recall + Precision} \tag{5}$$

3 Methodik der Messung und Klassifikation

3.1 Aufbau des Experiments

Eine FD-FLIM-Kamera der Excelitas PCO AG und ein Laser des Herstellers Omicron Laserage GmbH in Kombination mit dem Mikroskop PSM-1000 von CASCADE MICROTECH zur Steuerung des Strahlengangs sind die Hauptkomponenten des experimentellen Aufbaus [5] (siehe

Bild 2 (links)).

Open Access. © 2023 **M. Dietlmeier**, A. S. Rajan1, N. Leiter1, M. Wohlschläger1, M. Versen1
This work is licensed under the Creative Commons Attribution 4.0 License. +

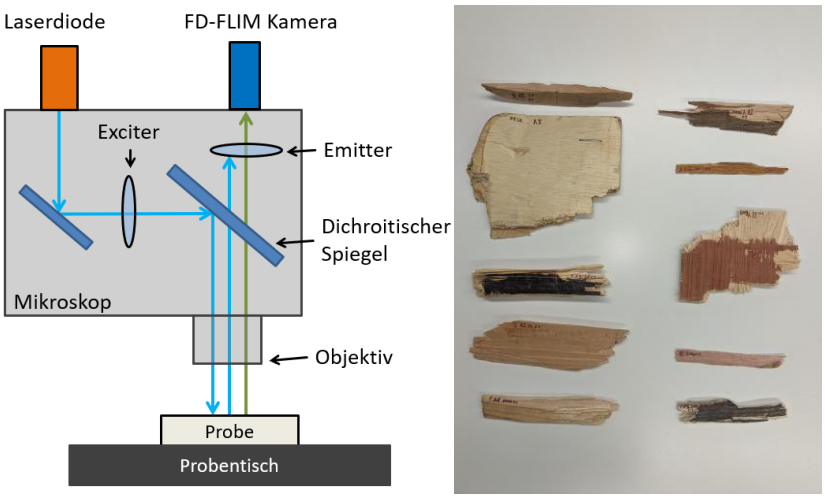

Bild 2: Schematische Darstellung des Messaufbaus bestehend aus Laserdiode, FD-FLIM Kamera und Mikroskop (links) und Beispiele für zu klassifizierende Altholzproben (rechts).

Die in Bild 2 beispielhaft gezeigten Altholzproben werden unter das Mikroskop mit einer 10-fachen Vergrößerung gelegt und mit einer Laserdiode mit einer Wellenlänge von 405nm angeregt. Ein optischer Bandpassfilter wird im Anregungslichtweg verwendet, um die Bandbreite des anregenden Laserlichts zu begrenzen. Zusätzlich wird ein optischer Langpassfilter im Lichtweg der Fluoreszenzemission verwendet, um Streulicht oder Reflektionen von der Probenoberfläche zu blocken. Die FD-FLIM-Kamera detektiert das Fluoreszenzsignal und misst die Phasenverschiebung und Modulation. Die gemessenen Daten werden mit Nikon‘s NIS-ELEMENTS Software verarbeitet und die phasen- und modulationsabhängige FLT werden berechnet. Das Ergebnis wird im ND2-Dateiformat gespeichert. Insgesamt werden 664 Bilder von 160 Proben aufgenommen. Die ND2-Dateien bestehen aus einem Bild mit fünf Kanälen: der Fluoreszenzintensität (I), der Phasenverschiebung (P), dem Modulationsindex (M), der phasenabhängigen FLT (PL) und der modulationsabhängigen FLT (ML), sowie die Metadaten der Messung. Jeder Kanal hat eine Größe von 1008x1004 Pixeln. Darüber hinaus enthalten die Metadaten Informationen über die Kamera, wie z.B. Kameratyp, die Temperatur der Kamera zum Zeitpunkt der der Aufnahme und die Belichtungszeit.

3.2 Verarbeitung der Messdaten

Eine Übersicht über den Datenfluss ist in Bild 3 dargestellt. Zunächst muss der Versuchsaufbau referenziert werden. Vom referenzierten Versuchsaufbau werden die fünf Kanäle erfasst und die Metadaten als ND2-Dateien mit dem Label als Dateinamen auf einer Festplatte gespeichert. Die ersten drei Schritte werden für 664 Messungen wiederholt, wobei die Initialisierung nur notwendig ist, wenn die Kamera neu gestartet wird. Anschließend werden die gesammelten Daten für die beiden unterschiedlichen NN-Typen vorbereitet. Für die MLPs werden verschiedene Kombinationen der fünf Kanäle, für die jeweils der Mittelwert und die Standard Abweichung berechnet werden, als TensorFlow Datensatz gespeichert. Zusätzlich werden die Dateinamen bzw. Label als Text Datei gespeichert. Der Vorteil, jeden Datensatz einzeln zu speichern, trotz der größtenteils redundanten Informationen liegt in dem zeitlichen Aufwand, die ND2 Dateien zu öffnen. Bei den Datensätzen für das CNN werden die Bilder mit der „image_resize“ Funktion aus der TensorFlow Bibliothek verkleinert.

Diese Limitierung der Informationen ist für sehr große Netzwerke notwendig, da das Training ansonsten aufgrund von fehlendem Arbeitsspeicher scheitert. Die „image_resize“ Funktion nimmt das Originalbild, die Form für das verkleinerte Bild und den Modus für die Transformation als Eingabeparameter. Es gibt acht verschiedene Modi für die Funktion image resize(). Nach einer Analyse scheinen die drei Modi Bilinear, Gaussian und Area, die meisten Details in den veränderten Bildern zu liefern. Ein Beispiel für ein Original und in der Größe veränderte Bilder der Intensität sind in Bild 3 (rechts) zu sehen.

Vor dem Training werden die vorgefertigten Datensätze von einer Funktion geladen und aufgeteilt. Die Aufteilung erfolgt mit der „train_test_split“ Funktion aus der „sklearn“ Bibliothek, welche die Prozentzahl der Aufteilung als Parameter für das Teilen der Daten verwendet. Des Weiteren kann die Aufteilung strategisch vorgenommen werden, sodass die Bilder der vier Klassen in etwa gleich verteilt sind über das Test- und Trainingsset.

Open Access. © 2023 **M. Dietlmeier**, A. S. Rajan1, N. Leiter1, M. Wohlschläger1, M. Versen1
This work is licensed under the Creative Commons Attribution 4.0 License. +

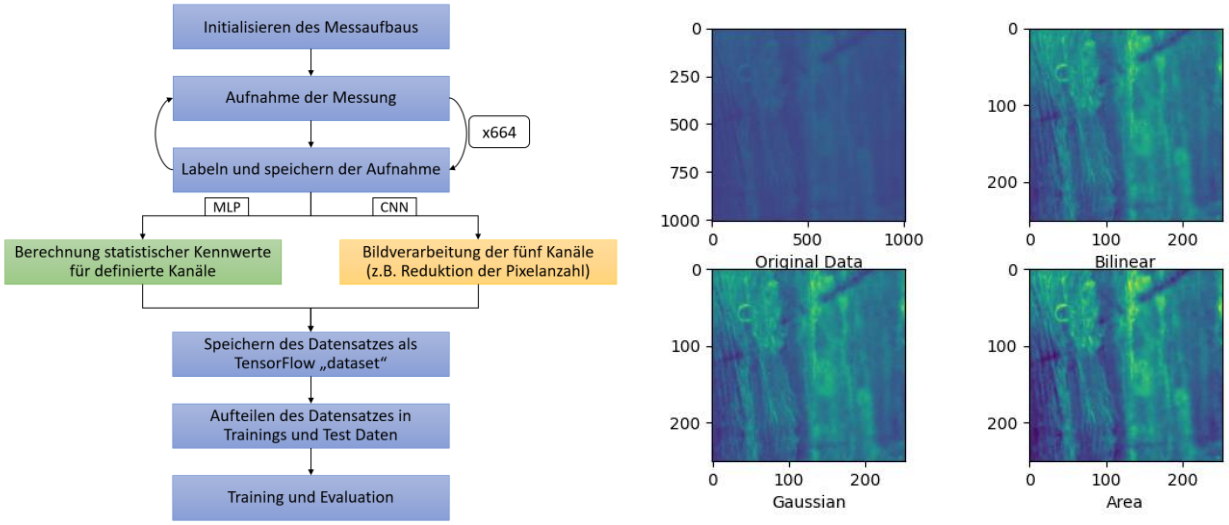

Bild 3: Flussdiagramm der Datenverarbeitung (links) und vergleich von Original und verkleinertem Bild (rechts)

3.3 Aufbau der Netzwerke

3.3.1 Aufbau des MLP

Das MLP wird mit der „sci-kit learn“ Bibliothek für Python erstellt. Die „sci-kit learn“ Bibliothek bietet zwei einfache MLP Modelle, an die mit Parametern genau definiert werden können, wobei hier das MLP Modell für Klassifikationsprobleme verwendet wird. Für diesen „MLPClassifier“ kann mit Parametern die maximale Anzahl der Epochen, die Größe der Hidden-Layer, die Aktivierungsfunktion und die Lernrate definiert werden. Mithilfe der „GridSearch“ Funktion können verschiedene Kombinationen an Hyperparametern getestet werden. Nachdem die optimale Kombination an Hyperparametern gefunden ist, werden die unterschiedlichen Kombination aus den Kanälen (I, P, M, PL, ML) getestet. Bei der Funktion zum Laden der Datensätze wird noch ein zusätzlicher Parameter gesetzt, der die Liste aus Mittelwerten und Standard Abweichungen mithilfe des „StandardScalers“ der „sci-kit learn“ Bibliothek normalisiert. Da verschiedene Kombinationen der Hyperparameter vielsprechend gewesen sind, werden alle möglichen Kombinationen der Kanäle für eine Kombination der Hyperparameter trainiert und gegeneinander verglichen.

3.3.2 Aufbau des CNN

Als Basis für das CNN werden verschiedene vortrainierte Netzwerke verwendet. Ein bekanntes Beispiel ist das Inception v3 Netzwerk [16], dass wie die meisten vortrainierten Netzwerke anhand großer öffentlich zugänglicher Bilddatenbanken trainiert wird. Diese Bilddatenbanken (z.B. ImageNet) beinhalten tausende gelabelte RGB-Bilder aus den verschiedensten Kategorien. Demzufolge sind die CNNs auf Bilder mit drei Kanälen ausgelegt, während die Bilddaten aus den FLIM-Messungen fünf Kanäle haben. Konsequenterweise müssen die Bilddaten der Datensätze mit mehr als drei Kanälen erst auf die korrekte Größe gefaltet werden. Nach der optionalen ersten Faltung auf die korrekte Größe für das folgende vortrainierte Netzwerk folgen am Ende noch ein „Pooling“, ein „Dropout“, ein „Flatten“-Layer und ein weiterer Output-Layer. Die genannten Layer sind notwendig, da die reguläre Ausgabe der vortrainierten Netzwerke auf die Vielzahl der Klassen in den Trainingsdaten konfiguriert ist, während lediglich zwei Klassen voneinander unterschieden werden. Der „Optimizer“ sowie die „Loss“ Funktion sind frei wählbar aus den verfügbaren Optionen der TensorFlow Bibliothek und können der Funktion, die das Model erstellt und kompiliert, als Parameter übergeben werden.

3.4 Training und Bewertung der neuronalen Netzwerke

Das Training von neuronalen Netze ist ein Optimierungsproblem. Folglich müssen verschiedene Netze vergleichbar sein. Für den Vergleich muss bei der Bewertung der Einfluss des Zufalls eliminiert werden, da das Training selbst bei identischer Struktur und Hyperparametern keine identischen Ergebnisse liefert. Eine Methode um den Einfluss von Zufällen beim Training zu eliminieren

Open Access. © 2023 **M. Dietlmeier**, A. S. Rajan1, N. Leiter1, M. Wohlschläger1, M. Versen1
This work is licensed under the Creative Commons Attribution 4.0 License. +

ist die „K-Fold“ Validierung [17]. Diese Validierung über mehrere Faltungen teilt den Datensatz in Test und Trainingsdaten, wobei sich die Testdatensätze in jeder Faltung unterscheiden, wie in

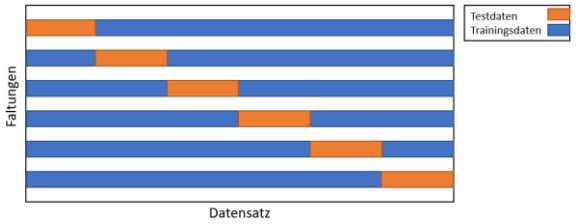

Bild 4 zu sehen ist.

In diesem Projekt wird k=10 verwendet aufgrund der Empfehlung aus [18]. Zudem wurde die strategische Variante der „K-Fold“ Validierung aus der „sci-kit learn“ Bibliothek verwendet. Diese Erweiterung versucht, gleiche Anteile aller Klassen in den Test- und Trainingsdatensätzen zu erhalten.

Da eine schnelle Auswertung ein Zusatzziel ist, wird jedes Netzwerk für alle möglichen Kombinationen der Kanäle getestet. Jede Kombination wurde mit dem 10-Fold Training trainiert und die Bewertung

erfolgt anhand der Kriterien aus Kapitel 2.3. Dabei wird bei den Konfusionsmatrizen die absolute

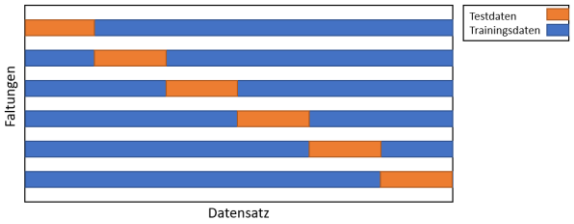

Bild 4: Visualisierung der „K-Fold“ Aufteilung eines Datensatzes

Anzahl der jeweiligen Kategorie in einen Prozentwert umgewandelt. Des Weiteren wird die Konfusionsmatrix als Gesamtdarstellung über die 10-Folds dargestellt, wodurch die Vorhersagen aller Faltungen aufsummiert werden. Die Kennzahlen zur Bewertung werden über alle Faltungen berechnet. Als zusätzliche Information werden die Kennzahlen sowie die Konfusionsmatrix in Textform für das beste Netzwerk aus den 10 Faltungen gespeichert. Diese Zusatzinformation hilft bei der Bewertung des Netzwerkes, da ein großer Unterschied zwischen den gemittelten Kennzahlen und den besten Kennzahlen auf Probleme bei dem Training hinweist.

4 Ergebnisse

4.1 MLP

Bei den Versuchen mit MLP Netzwerken hat sich der Adam [19] “Optimizer” mit dem hyperbolischen Tangens als Aktivierungsfunktion über 1500 Iterationen als Optimum gezeigt. Dabei wurde „early stopping“ aktiviert, sodass ein „Overfitting“ vermieden wird.

Das beste Netzwerk erreicht einen F1-score von 91.7% mit ähnlichen (±0.13%) Werten für die anderen Kennzahlen. Die Konfusionsmatrix (siehe Bild 5 links) spiegelt diese Kennwerte wieder. Insbesondere der niedrige Wert (5.21%) an falsch positiv klassifizierten Proben zeigt das Potential dieser Klassifikation in einer industriellen Anwendung. Des Weiteren sind lediglich vier der fünf möglichen Kanäle (I, P, M, ML) genutzt worden. Ein weiteres MLP Model, das mit den Daten von drei Kanälen (M, PL, ML) trainiert worden ist, erreicht einen F1-Score von 90.79% (siehe Bild 6 rechts).

Open Access. © 2023 **M. Dietlmeier**, A. S. Rajan1, N. Leiter1, M. Wohlschläger1, M. Versen1
This work is licensed under the Creative Commons Attribution 4.0 License. +

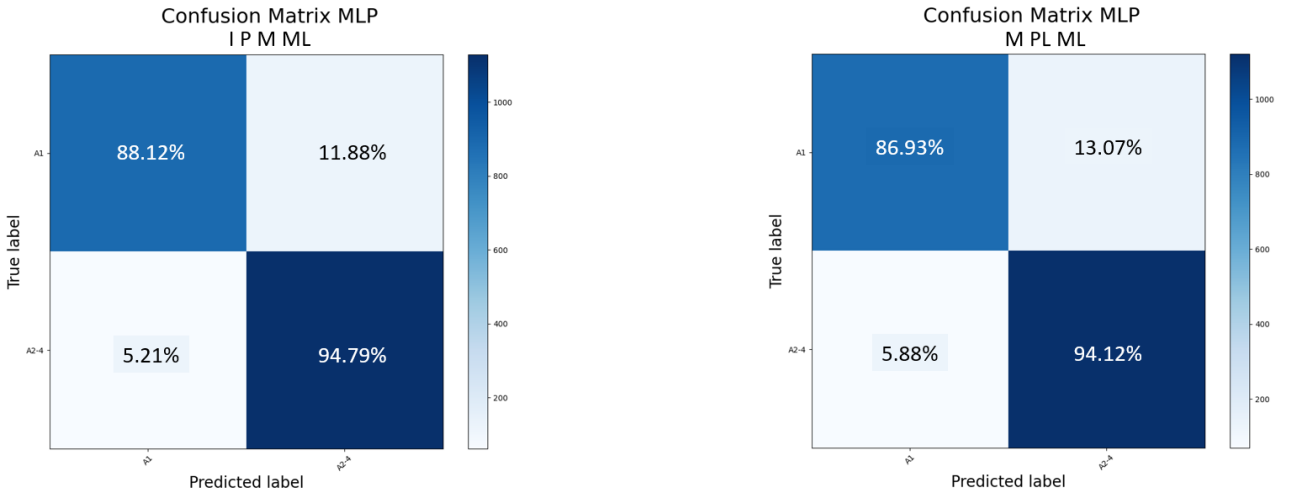

Bild 5: Konfusionsmatrix des MLPs, das die Althölzer anhand von 4 Kanälen klassifiziert (links) und Konfusionsmatrix des MLPs, das die Althölzer anhand von 3 Kanälen klassifiziert (rechts).

4.2 CNN

Das bestes Ergebnis bei den CNN Netzwerken wird mit der Inception v3 Architektur [20] erreicht. Dabei wird der „NAdam-Optimizer“ verwendet, der eine Weiterentwicklung des „Adam-Optimizers“ ist. Als Verlustfunktion wird die binäre Kreuzentropie gewählt und als Aktivierungsfunktion die „SeLU“ Funktion. Das Netzwerk ist mit über 500 Iterationen trainiert worden und auch hier ist „early stopping“ implementiert worden. Damit wird ein F1-Score von 89.07% erreicht mit den anderen Kennwerten in dem gleichen Bereich (±0.1%). Die Konfusionsmatrix ist in Bild 6 zu sehen, wobei auch hier festgestellt werden kann, dass der Wert an falsch positiven Klassifikationen gering ist (8.91%).

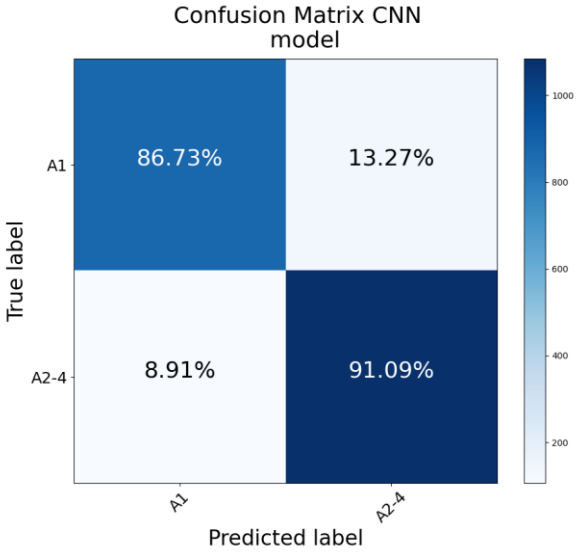

Bild 6: Konfusionsmatrix des CNNs, welches die Althölzer anhand von allen Kanälen klassifiziert

5 Zusammenfassung und Ausblick

FD-FLIM hat sich als geeignete Messmethode zur Erkennung der Altholzklasse herausgestellt. Aus diesem Grund wurden zur automatisierten Klassifikation von FD-FLIM Messungen von Althölzern der Klassen A1-A4 ein MLP und ein CNN getestet und optimiert. Hierzu wurde ein Datensatz an 664 FD-FLIM Messungen von Althölzern der Klassen A1-A4 aufgenommen.

Beide neuronalen Netze erwiesen sich als ausgesprochen gut zur Klassifikation von Althölzern basierend auf FD-FLIM Messungen. So erreichte das MLP einen geringfügig besseren F1-Score von 90,79 % als das CNN mit 89,07 %. Zudem

Open Access. © 2023 **M. Dietlmeier**, A. S. Rajan1, N. Leiter1, M. Wohlschläger1, M. Versen1
This work is licensed under the Creative Commons Attribution 4.0 License. +

stellte sich heraus, dass ist die Datenvorbereitung für das MLP schneller ist als die Datenvorverarbeitung des CNN was wiederum ein weiterer Schritt in Richtung einer echtzeitfähige Klassifikation darstellt. Zusammenfassend zeigten die Untersuchungen, dass sowohl ein MLP als auch ein CNN zur Klassifikation von Althölzern im Bereich des Altholzrecyclings basierend auf FD-FLIM Daten geeignet ist.

Künftig sollen das MLP und das CNN weiter optimiert werden. So können zum Beispiel bei den MLP weitere statistische Werte wie der Median etc. als neue Features hinzugefügt werden, um den F1-Score weiter zu erhöhen. Eine weitere Möglichkeit wäre die Umsetzung des MLP in TensorFlow, welches mehrere selbst spezifizierbare Optionen bei der Erstellung des neuronalen Netzwerks bietet. Bei den CNN Netzwerken könnten große Verbesserungen durch ein selbst erstelltes und trainiertes NN, das auf Basis von FD-FLIM Bilder klassifiziert, erzielt werden, da die bisher verwendeten, vortrainierten CNNs mit RGB Bildern trainiert wurden, welche sich stark von den FD-FLIM Bildern unterscheiden.

6 Literaturangaben

[1] BAV e.V.: Waste wood - Bundesverband der Altholzaufbereiter und -verwerter | German Waste Wood Association - BAV e.V (2019)

[2] Goldhahn, C., Cabane, E. u. Chanana, M.: Sustainability in wood materials science: an opinion about current material development techniques and the end of lifetime perspectives. Philosophical transactions. Series A, Mathematical, physical, and engineering sciences 379 (2021) 2206

[3] Leiter, N.: A significance analysis of the fluorescence decay times of different material combinations, TH Rosenheim Master thesis 2021

[4] N. Leiter, M. Wohlschlaeger, M. Versen u. C. Laforsch: Non-destructive differentiation of treated and non-treated wood using fluorescence lifetime imaging microscopy. Sensors and Measuring Systems; 21th ITG/GMA-Symposium. 2022, S. 1–6

[5] Lakowicz, J. R. (Hrsg.): Principles of fluorescence spectroscopy. New York, NY: Springer 2006

[6] N. Leiter, M. Wohlschläger, V. Auer, M. Versen u. C. Laforsch: A Novel Approach to Identify Wood Species Optically using Fluorescence Lifetime Imaging Microscopy. SMSI 2021 - Sensors and Instrumentation (2021)

[7] Merck: Fluorescence Lifetime Measurement (2022)

[8] Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification, He, K., Zhang, X., Ren, S. u. Sun, J., 2015

[9] McCulloch, W. S. u. Pitts, W.: A logical calculus of the ideas immanent in nervous activity. The Bulletin of Mathematical Biophysics 5 (1943) 4, S. 115–133

[10] Rumelhart, D. E., Hinton, G. E. u. Williams, R. J.: Learning representations by back-propagating errors. Nature 323 (1986) 6088, S. 533–536

[11] Goodfellow, I., Bengio, Y. u. Courville, A.: Deep learning. Adaptive computation and machine learning. Cambridge, Massachusetts: The MIT Press 2016

[12] Glorot, X., Bordes, A. u. Bengio, Y.: Deep Sparse Rectifier Neural Networks. Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research. Fort Lauderdale, FL, USA: PMLR 2011, S. 315–323

[13] He, K., Zhang, X., Ren, S. u. Sun, J.: Deep Residual Learning for Image Recognition. 2015. https://arxiv.org/pdf/1512.03385

[14] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot u. Édouard Duchesnay: Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 12 (2011) 85, S. 2825–2830

[15] Chinchor, N. u. Sundheim, B.: MUC-5 evaluation metrics. Proceedings of the 5th conference on Message understanding - MUC5 '93. Morristown, NJ, USA: Association for Computational Linguistics 1993

[16] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke u. Andrew Rabinovich: Going Deeper With Convolutions

[17] Ron Kohavi: A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. undefined (1995)

[18] Geoffrey J. McLachlan, Kim-Anh Do, Christophe Ambroise: Supervised Classification of Tissue Samples. In: Wiley Series in Probability and Statistics. 2004, S. 214

Open Access. © 2023 **M. Dietlmeier**, A. S. Rajan1, N. Leiter1, M. Wohlschläger1, M. Versen1
This work is licensed under the Creative Commons Attribution 4.0 License. +

[19] Adam: A Method for Stochastic Optimization, Kingma, D. P. u. Ba, J., 2014
[20] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. u. Wojna, Z.: Rethinking the Inception Architecture for Computer Vision. 2015. https://arxiv.org/pdf/1512.00567

Open Access. © 2023 **M. Dietlmeier**, A. S. Rajan1, N. Leiter1, M. Wohlschläger1, M. Versen1
This work is licensed under the Creative Commons Attribution 4.0 License. +