

Johannes Anastasiadis*, Philipp Benzing und Fernando Puente León

Erzeugung künstlicher Datensätze zum Training konvolutionaler neuronaler Netze für die spektrale Entmischung

Generation of artificial data sets to train convolutional neural networks for spectral unmixing

<https://doi.org/10.1515/teme-2020-0008>

Eingang 3. März 2020; angenommen 26. Mai 2020

Zusammenfassung: In diesem Beitrag wird eine Methode vorgestellt, mit der Daten zum Training von künstlichen neuronalen Netzen für die spektrale Entmischung erzeugt werden. Dies hat den Vorteil, dass nur Spektren der beteiligten Reinstoffe und, je nach verwendetem Modell, wenige Mischspektren zur Bestimmung der Parameter an realen Daten zur Verfügung stehen müssen. Daraus können mit Hilfe von Mischmodellen, die auch direkt zur Entmischung herangezogen werden können, große Mengen an Spektren zum Trainieren erzeugt werden. Im Gegensatz zum direkten Einsatz der Mischmodelle, wo von einem Spektrum pro Reinstoff ausgegangen wird, wird hier die Spektrenvariabilität berücksichtigt, indem unterschiedliche Spektren desselben Reinstoffs genutzt werden. Dabei wird die Eigenschaft künstlicher neuronaler Netze ausgenutzt, aussagekräftige Merkmale auf Basis großer Datenmengen lernen zu können.

Schlagwörter: Spektrale Entmischung, nichtlineare Mischmodelle, Spektrenvariabilität, künstliche neuronale Netze.

Abstract: This paper presents a method to generate training data for artificial neural networks for spectral unmixing. Therefore, only the spectra of the pure substances involved and, depending on the model used, a few spectra of mixed substances to determine the parameters are needed. With mixing models, which can also be used directly for spectral unmixing, large quantities of spectra can be generated for training. In contrast to the direct use of mixing models, where a spectrum per pure substance is used, this approach takes into account the spectrum variability by using different spectra of each pure substance.

***Korrespondenzautor:** Johannes Anastasiadis, Karlsruher Institut für Technologie, Institut für Industrielle Informationstechnik, Karlsruhe, Germany, E-Mail: anastasiadis@kit.edu

Philipp Benzing, Fernando Puente León, Karlsruher Institut für Technologie, Institut für Industrielle Informationstechnik, Karlsruhe, Germany

The property of artificial neural networks to learn significant features based on large amounts of data is exploited here.

Keywords: Spectral unmixing, nonlinear mixing models, spectral variability, artificial neural networks.

1 Einleitung

Optische Messverfahren eignen sich besonders zur Überwachung industrieller Prozesse, da sie berührungslos und nicht-destruktiv sind. Die Sicherstellung der Zusammensetzung des Endprodukts ist dabei eine wichtige Aufgabe. Wenn es sich bei dem Produkt um eine Mischung aus feinen Pulvern handelt, ist die örtliche Auflösung von Kameras zu gering, um an jedem Pixel nur ein Material bzw. einen Stoff zu erfassen. Stattdessen werden Stoffgemische aufgenommen, woraus hinterher die Komponenten und deren relative Anteile bestimmt werden müssen. Mit herkömmlichen Farbkameras ist dies nicht zu bewältigen. Grund dafür ist die Beschränkung auf nur drei Farbkanaäle (Rot, Grün, Blau) pro Pixel, wohingegen hyperspektrale Kameras ein Spektrum an jedem Pixel aufnehmen können [10]. In den Spektren sind Informationen über die Stoffeigenschaften enthalten, womit Rückschlüsse auf die Zusammensetzung gezogen werden können [1]. Die Schätzung der relativen Anteile der beteiligten Stoffe mit Hilfe des Spektrums wird als spektrale Entmischung bezeichnet (vgl. Abschnitt 2).

Künstliche neuronale Netze (KNN) wurden in den vergangenen Jahren sehr erfolgreich in der Bildverarbeitung eingesetzt. Auch für die Klassifikation hyperspektraler Bilder eignen sie sich [5, 27]. Darüber hinaus können Regressionsaufgaben mit reellwertigen Ausgangsgrößen, wie eine spektrale Entmischung (z. B. [28]), mit neuronalen Netzen gelöst werden. Aufgrund der hohen Anzahl an Trainingsparametern wird jedoch eine große Menge an Daten zum Trainieren des Netzes benötigt. Dieses Problem wird in [26] durch unüberwachtes Training umgangen. Ei-

ne weitere Möglichkeit, diesem Problem zu begegnen, sind konvolutionale neuronale Netze (CNN) [22].

In diesem Beitrag werden CNN zur spektralen Entmischung genutzt. Darüber hinaus wird mit Hilfe von Mischmodellen eine große Anzahl an Trainingsdaten generiert, die reale Reinspektren als Grundlage haben. Dieser Ansatz hat den Vorteil, dass, je nach Mischmodell, keine Mischspektren bzw. nur eine kleine Menge an realen Daten von Mischspektren benötigt werden, um die Parameter bei nichtlinearen Mischmodellen zu bestimmen. In einem industriellen Umfeld sollten die beteiligten Reinstoffe bekannt und deren Spektren verfügbar sein. Bei der Erstellung der künstlichen Mischspektren wird die Spektrenvariabilität ausgenutzt, um viele variierende Spektren eines Mischverhältnisses erzeugen zu können. Da KNN, bei entsprechender Größe, beliebige Funktionen approximieren können, eignen sie sich besonders, um diese Varianz in den Spektren zu berücksichtigen und eine entsprechende Invarianz zu lernen.

Dieser Beitrag gliedert sich wie folgt: In Abschnitt 2 wird zunächst die spektrale Entmischung eingeführt und im weiteren Verlauf vor allem auf Mischmodelle und die Spektrenvariabilität eingegangen. Anschließend werden in Abschnitt 3 die Grundlagen zu KNN bzw. CNN gelegt, bevor näher auf die Struktur des CNN eingegangen wird, das zur Untersuchung der in dieser Arbeit erzeugten Datensätze genutzt wird. Aufbauend auf den Abschnitten 2 und 3 wird in Abschnitt 4 die Vorgehensweise zur Erzeugung der künstlichen Mischspektren vorgestellt. Die damit erzeugten Datensätze werden in Abschnitt 5 untersucht und bewertet. Der Beitrag wird in Abschnitt 6 mit einer Zusammenfassung geschlossen.

2 Spektrale Entmischung

Die Schätzung von Materialanteilen in einem Stoffgemisch aufgrund seines Spektrums wird als spektrale Entmischung bezeichnet [16]. Dabei wird zwischen überwachter und nicht überwachter spektraler Entmischung unterschieden, wobei bei letzterer die Spektren der Reinstoffe mitgeschätzt werden. In diesem Beitrag werden die Reinspektren als bekannt vorausgesetzt, sodass eine überwachte spektrale Entmischung erfolgen kann. Bei der Schätzung $\hat{\mathbf{a}} = [\hat{a}_1, \dots, \hat{a}_P]^T \in \mathbb{R}^P$ der relativen Anteile $\mathbf{a} = [a_1, \dots, a_P]^T \in \mathbb{R}^P$ der bis zu P beteiligten Reinstoffe müssen Nebenbedingungen eingehalten werden, um die physikalische Interpretierbarkeit zu wahren. Das sind die Nichtnegativitätsbedingung

$$\hat{a}_p \geq 0 \quad \forall p \quad (1)$$

und die Normierungsbedingung

$$\sum_{p=1}^P \hat{a}_p = 1. \quad (2)$$

2.1 Hyperspektralbild

In diesem Beitrag wird die spektrale Entmischung von Hyperspektralbildern durchgeführt. Diese besitzen, im Gegensatz zu Farbbildern mit drei Farbkanälen (Rot, Grün, Blau), eine hohe Anzahl an fein abgetasteten Wellenlängenkanälen. Je nach Anwendung können mehrere hundert Kanäle zum Einsatz kommen, die vom Ultravioletten bis in den Infrarotbereich reichen können. Eine gängige Darstellung für Hyperspektralbilder ist ein Würfel, bei dem zwei Kanten den örtlichen Dimensionen entsprechen und die dritte Kante der spektralen Dimension entspricht. Für jedes Pixel liegt ein Spektrum vor, das mit den Methoden der Spektroskopie verarbeitet werden kann, und für jede Wellenlänge ein Grauwertbild, bei dem Methoden der Bildverarbeitung angewendet werden können. Hyperspektralbilder erlauben es zudem, alle Dimensionen gleichzeitig zu untersuchen (siehe bspw. [2, 4]).

Es existieren verschiedene Verfahren zur Aufnahme von Hyperspektralbildern. Neben dem Schnappschussverfahren, welches das vollständige Bild mit einer Aufnahme aufzeichnet, existieren scannende Verfahren, die das Hyperspektralbild zeilenweise oder kanalweise aufnehmen. Die Verfahren stellen einen Kompromiss zwischen (zeitlichem) Aufwand und Auflösung dar. In dieser Arbeit sollen Hyperspektralbilder mit Hilfe von spektraler Entmischung analysiert werden.

2.2 Mischmodelle

Viele Verfahren zur spektralen Entmischung sind modellbasiert, wobei zwischen physikalisch motivierten Modellen und Modellen, die die Mischspektren mit einer parametrisierten Funktion approximieren, unterschieden wird. Ein Beispiel für erstere ist das Hapke-Modell [12], welches aus der Theorie des Strahlungstransports entstanden ist. Da für physikalisch motivierte Mischmodelle viel Wissen über die Messbedingungen und die Mischungen vorhanden sein muss, was meist nicht vorliegt, werden diese hier nicht genutzt.

Das am häufigsten verwendete Mischmodell ist das lineare Mischmodell (LMM), welches für viele Anwendun-

gen eine gute Näherung darstellt [4, 16–19]:

$$\mathbf{y} = \sum_{p=1}^P \mathbf{m}_p a_p + \boldsymbol{\omega} = \mathbf{M} \mathbf{a} + \boldsymbol{\omega}. \quad (3)$$

Dabei ist $\mathbf{y} \in \mathbb{R}^A$ ein an A Wellenlängenbereichen abgetastetes Reflektanzspektrum eines Stoffgemischs und entspricht einem Pixel eines hyperspektralen Bildes. Die Matrix $\mathbf{M} = [\mathbf{m}_1, \dots, \mathbf{m}_P] \in \mathbb{R}^{A \times P}$ beinhaltet die Reinspektren der bis zu P am Gemisch beteiligten Stoffe. In $\boldsymbol{\omega} \in \mathbb{R}^A$ sind alle Abweichungen zwischen \mathbf{y} und dem modellierten Spektrum enthalten. Das LMM beruht auf den Annahmen, dass die Mischung in einem makroskopischen Maßstab stattfindet und Photonen nur mit einem Material interagieren, bevor sie auf den Photosensor treffen [6, 24].

Es existieren darüber hinaus nichtlineare Mischmodelle, die in [7] zusammengefasst sind. In diesem Beitrag werden das generalisierte bilineare Mischmodell (GBM) [11]

$$\mathbf{y} = \sum_{p=1}^P \mathbf{m}_p a_p + \sum_{p=1}^{P-1} \sum_{q=p+1}^P \gamma_{pq} a_p a_q \mathbf{m}_p \odot \mathbf{m}_q + \boldsymbol{\omega} \quad (4)$$

und das linear-quadratische Mischmodell (LQM) [23]

$$\mathbf{y} = \sum_{p=1}^P \mathbf{m}_p a_p + \sum_{p=1}^P \sum_{q=1}^P b_{pq} \mathbf{m}_p \odot \mathbf{m}_q + \boldsymbol{\omega} \quad (5)$$

verwendet. Dabei sind γ_{pq} und b_{pq} die Nichtlinearitätskoeffizienten und \odot ist das elementweise Produkt. Für $\gamma_{pq} = 1 \forall p, q$ entspricht das GBM dem Fan-Modell (FM) [9]. Diese Mischmodelle berücksichtigen auch Licht, das vor dem Auftreffen auf den Photosensor mit bis zu zwei Stoffen interagiert hat. Dabei fließen beim GBM die Anteile auch in die Gewichtung der Summanden, die eine Mischung aus zwei Spektren berücksichtigen, mit ein, während diese Gewichte beim LQM frei sind.

2.3 Anteilsschätzung

Mit Hilfe der Mischmodelle kann eine Schätzung der Parameter $\hat{\boldsymbol{\theta}} \in \mathbb{R}^X$ durchgeführt werden:

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta}} \|\mathbf{y} - \boldsymbol{\mu}(\mathbf{M}, \boldsymbol{\theta})\|_2^2. \quad (6)$$

Dabei steht die Funktion $\boldsymbol{\mu} : \mathbb{R}^{P \times X} \rightarrow \mathbb{R}^A$ stellvertretend für die jeweiligen Mischmodelle ohne $\boldsymbol{\omega}$. Der Parametervektor $\boldsymbol{\theta}$ besteht beim LMM und FM nur aus den Anteilen \mathbf{a} und damit gilt $X = P$. Beim GBM bzw. LQM kommen noch die Nichtlinearitätskoeffizienten dazu, somit ist

hier $X = \frac{P}{2}(P+1)$ bzw. $X = P^2 + P$. Beim LMM ist (6) konvex, da es sich um eine quadratische Funktion handelt und die Hesse-Matrix $\mathbf{M}^T \mathbf{M}$ positiv definit ist. Daher findet beispielsweise ein *Active-Set-Verfahren* (AS), welches die Nebenbedingungen (1) und (2) miteinbezieht, immer das globale Minimum. Aufgrund von $\boldsymbol{\omega}$ ist dies im Allgemeinen jedoch keine optimale Schätzung der Anteile \mathbf{a} . Bei der Optimierung von (6) basierend auf einem nichtlinearen Mischmodell handelt es sich im Allgemeinen nicht um ein konvexes Optimierungsproblem, wodurch hier das Finden des globalen Optimums nicht mehr garantiert ist.

Ein gängiges Verfahren, welches auf dem LMM basiert und die Nebenbedingungen (1) und (2) miteinbezieht, ist das *Fully-Constrained-Least-Squares-Verfahren* (FCLS) [13]. Dabei wird die Lagrange-Funktion $L : \mathbb{R}^{P+1} \rightarrow \mathbb{R}$ mit

$$L(\mathbf{a}, l) = \|\mathbf{y} - \mathbf{M} \mathbf{a}\|_2^2 - l \left(\sum_{p=1}^P a_p - 1 \right) \quad (7)$$

und dem Lagrange-Multiplikator $l \in \mathbb{R}$ optimiert. Es ist ein iteratives Verfahren, bei dem negative \hat{a}_p und die korrespondierenden Reinspektren aus \mathbf{M} entfernt werden.

Für die auf nichtlinearen Mischmodellen basierenden Schätzungen können gradientenbasierte Liniensuchverfahren verwendet werden. In [11] wird dies für das GBM vorgeschlagen. Darüber hinaus existiert für das FM der *Fan-FCLS-Algorithmus* (FFCLS), der auf der Linearisierung des Mischmodells durch Abbrechen der Taylorreihe basiert [9]. Der FFCLS wird in [11] in abgewandelter Form für das GBM verwendet.

2.4 Spektrenvariabilität

Alle Mischmodelle und Verfahren zur spektralen Entmischung, die bisher vorgestellt wurden, gehen von der Annahme aus, dass jeder Reinstoff durch ein einziges Spektrum repräsentiert werden kann. In der Realität unterscheiden sich jedoch die Spektren eines Reinstoffes bei mehreren Aufnahmen [25]. Diese Abweichungen werden als Spektrenvariabilität bezeichnet (siehe Abb. 1). Diese kann viele Ursachen haben, wie z. B. die Oberflächenstruktur, Umgebungsbedingungen und nicht zuletzt die Beleuchtung. Es gibt Verfahren, die die Spektrenvariabilität berücksichtigen. Hier wären das *Extended Linear Mixing Model* (ELMM) und das *Generalized Linear Mixing Model* (GLMM) zu nennen [8, 14]. Das ELMM erweitert das LMM zu

$$\mathbf{y} = \mathbf{M} \Phi \mathbf{a} + \boldsymbol{\omega}. \quad (8)$$

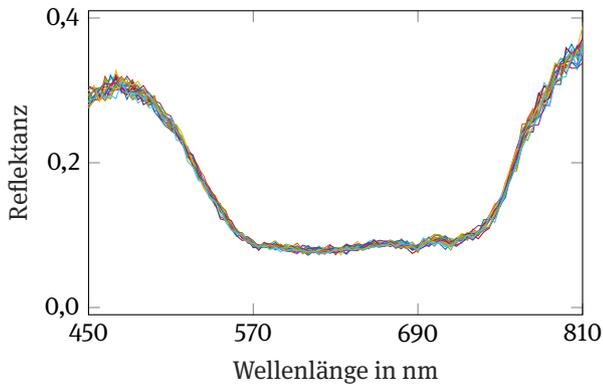


Abb. 1: Beispiele für Spektren des reinen blauen Quarzsandes des später vorgestellten Datensatzes.

Dabei ist $\Phi \in \mathbb{R}^{P \times P}$ eine Diagonalmatrix. Die Diagonalelemente wirken als Skalierungsfaktoren der einzelnen Reinspektren. Bei der Optimierung der Form (6) sind diese Skalierungsfaktoren Teil des zu optimierenden Parametervektors θ und es gilt $X = 2 \cdot P$.

Das GLMM führt wesentlich mehr Parameter ein und erweitert das LMM zu

$$\mathbf{y} = (\mathbf{M} \odot \Psi) \mathbf{a} + \boldsymbol{\omega}. \quad (9)$$

Die Matrix $\Psi \in \mathbb{R}^{A \times P}$ enthält nun je einen Skalierungsfaktor für jeden Wellenlängenkanal eines jeden Reinspektrums. Bei der Optimierung der Form (6) sind diese Skalierungsfaktoren Teil des zu optimierenden Parametervektors θ und es gilt $X = (A+1)P$. Sowohl beim ELMM also auch beim GLMM werden die Nebenbedingungen (1) und (2) eingehalten [8, 14]. Ein weiteres Verfahren, welches iterativ die Abweichungen zu den Bedingungen (1) und (2) minimiert, in dem die Spektrenvariabilität berücksichtigt wird, wird in [20] vorgestellt. Dabei steht eine Menge an Reinspektren zur Verfügung, die verwendet wird, um die Matrix \mathbf{M} in (3) zu modifizieren.

In diesem Beitrag wird die Spektrenvariabilität berücksichtigt, indem ein Trainingsdatensatz erzeugt wird, der möglichst variable Spektren enthält. Dazu werden im nächsten Abschnitt neuronale Netze vorgestellt, bevor näher auf die Datensätze eingegangen wird.

3 Künstliche neuronale Netze

Künstliche neuronale Netze können, bei richtiger Dimensionierung, Zusammenhänge auf Basis von Daten lernen. Mit einem Datensatz aus vielen gemischten Spektren als

Datenpunkte kann ein neuronales Netz zur spektralen Entmischung trainiert werden [2]. Weisen die Spektren im Datensatz eine Spektrenvariabilität auf, so wird auch diese berücksichtigt.

3.1 Grundlagen

Künstliche neuronale Netze sind inspiriert von biologischen Neuronen und besitzen die Fähigkeit, Zusammenhänge aus einem Datensatz zu lernen. Dabei ist es wichtig, dass der Datensatz möglichst groß ist und viele Beispiele enthält. Gleichzeitig muss das Netz an die Größe des Datensatzes angepasst sein. Ziel ist es, eine möglichst gute Generalisierung zu erreichen, sodass das Netz auch unbekannte Daten richtig verarbeitet. Bei einem zu großen Netz oder einem zu kleinen Trainingsdatensatz besteht die Gefahr von Überanpassung, also dem Auswendiglernen des Trainingsdatensatzes. Ist das Netz zu klein, kann das eine Unteranpassung zur Folge haben, bei der das Netz nicht in der Lage ist, die Zusammenhänge im Datensatz abzubilden.

Aufgebaut sind KNN aus künstlichen Neuronen. Allgemein können die Neuronen beliebige Verbindungen haben. Die meisten Netze sind jedoch so verbunden, dass der Signalfluss nur in Vorwärtsrichtung möglich ist (*Feedforward*-Struktur) und die Neuronen in Schichten angeordnet sind. Ist jedes Neuron einer Schicht mit jedem Neuron der Folgeschicht verbunden, wird von vollständig verbundenen Schichten gesprochen. Eine solche Schicht lässt sich mathematisch mit

$$\mathbf{h} = \boldsymbol{\varphi}(\mathbf{W}\mathbf{x} + \mathbf{b}) \quad (10)$$

beschreiben, wobei $\mathbf{h} \in \mathbb{R}^K$ der Ausgang und $\mathbf{x} \in \mathbb{R}^J$ der Eingang einer Schicht mit J Eingangs- und K Ausgangsgrößen ist. Die zu lernenden Parameter sind die Gewichte $\mathbf{W} \in \mathbb{R}^{K \times J}$ und $\mathbf{b} \in \mathbb{R}^K$. Dies geschieht in den meisten Fällen mit Hilfe eines gradientenbasierten Liniensuchverfahrens. Die Aktivierungsfunktion $\boldsymbol{\varphi} : \mathbb{R}^K \rightarrow \mathbb{R}^K$ wird elementweise angewendet und muss nichtlinear sein, da das neuronale Netz ansonsten nur lineare Zusammenhänge abbilden könnte.

Bei CNN, die in dieser Arbeit verwendet werden, sind die Schichten nicht vollständig verbunden. Die Verbindungen, deren Gewichte sich wiederholen, sind hier lokal. Auch diese Schichten lassen sich mit (10) beschreiben, jedoch besteht die Matrix \mathbf{W} einer Faltungsschicht zu einem großen Teil aus Nullen. Die Parameter in \mathbf{W} bestehen aus dem Faltungskern, der sich wiederholt. Das bringt, vor allem in der Signalverarbeitung, Vorteile mit sich. Die Operation ist ortsinvariant, da der Faltungskern verschoben

wird. Außerdem gehen die örtlichen Beziehungen nicht verloren. Darüber hinaus müssen wesentlich weniger Parameter trainiert werden, was einer Überanpassung entgegenwirkt.

In der Praxis werden mehrere Faltungskerne je Schicht genutzt. Daraus resultieren mehrere Ausgangssignale, die Aktivierungskarten genannt werden. In der nächsten Schicht werden diese ebenfalls mit mehreren Faltungskernen gefaltet (siehe Abb. 2).

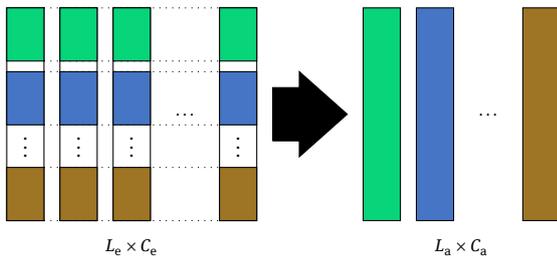


Abb. 2: Prinzip einer Faltungsschicht (1D-Beispiel): Die Eingangsdaten bestehen aus C_e Aktivierungskarten zu je L_e Werten. Sie werden mit C_a Gruppen an Filtern gefaltet, wobei jede Gruppe C_e Filter besitzt. Die C_a neuen Aktivierungskarten sind jeweils die Summe aller Faltungen einer Gruppe und besitzen jeweils L_a Werte.

Im Zusammenhang mit CNN werden außerdem *Pooling*-Schichten verwendet, die eine Nachbarschaft an Werten in einem einzelnen Wert zusammenfassen [21]. Meist wird sog. *Max-Pooling* verwendet, welches das Maximum einer Nachbarschaft an die nächste Schicht weitergibt. *Pooling* hat den Vorteil, dass das Netz in den hinteren Schichten kleiner wird, wodurch der Rechen- und Speicherbedarf sinkt. Darüber hinaus werden so bessere Generalisierungseigenschaften erreicht.

Häufig wird in KNN eine *Batch*-Normalisierung verwendet [15]. Diese gleicht zum einen die unterschiedlichen Verteilungen der Eingangsdaten aus, wenn nicht in jeder Iteration mit dem gesamten Datensatz trainiert wird. Zum anderen werden unterschiedliche Verteilungen der Eingangsdaten tieferer Schichten ausgeglichen, die dadurch entstehen, dass sich die Parameter der Schichten davor ändern. Die *Batch*-Normalisierung erfolgt vor der Aktivierungsfunktion und führt folgende Transformation eines Wertes $o_e \in \mathbb{R}$ durch:

$$o_a = \delta \frac{o_e - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} + \beta. \quad (11)$$

Hier sind μ_B und σ_B^2 Stichprobenmittelwert und -varianz von o_e bezogen auf einen *Batch*, also die Gruppe an Da-

tenpunkten, die in der aktuellen Iteration betrachtet wird. Die Parameter $\beta \in \mathbb{R}$ und $\delta \in \mathbb{R}$ sind trainierbar und $\epsilon \in \mathbb{R}$ ist eine kleine Zahl, die eine Division durch 0 verhindert. Bei Faltungsschichten teilen sich alle Werte einer Aktivierungskarte ein Parameterpaar β und δ , während bei vollständig verbundenen Schichten jeder Ausgangswert sein eigenes Paar bekommt. Durch β und δ können die Werte auf einen Bereich der Aktivierungsfunktion verschoben werden, der eine besonders hohe Ableitung besitzt, sodass schnellere Trainingsvorgänge möglich sind. Als weitere Vorteile werden in [15] höhere mögliche Lernraten und ein regularisierender Effekt genannt.

Im nächsten Abschnitt wird ein CNN vorgestellt, das aus diesen elementaren Komponenten besteht.

3.2 Verwendete Netzstruktur

Das in dieser Arbeit verwendete CNN entspricht weitestgehend dem in [2], was in [3] in abgewandelter Form zur Materialdetektion verwendet wird, und wird deshalb nur knapp vorgestellt.

Die Eingangsdaten des CNN sind hyperspektrale Datenwürfel und die Ausgangsdaten jeweils P Intensitätsbilder. Diese geben für jedes Pixel die relative Menge der erhaltenen Reinstoffe an. Das neuronale Netz kann in zwei Teile eingeteilt werden, wobei der erste Teil Merkmale extrahiert und der zweite Teil diese zur spektralen Entmischung nutzt.

Der erste Teil besteht aus dreidimensionalen Faltungsschichten (siehe Abb. 3), die Merkmale extrahieren, und *Max-Pooling*-Schichten in spektraler Richtung. Dadurch bleibt die örtliche Auflösung der hyperspektralen Bilder erhalten. Am Ausgang des ersten Teils liegen dreidimensionale Aktivierungskarten vor.

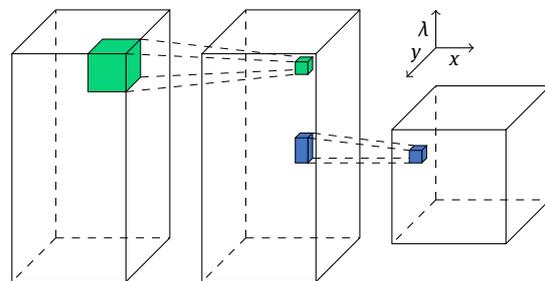


Abb. 3: Eine Schicht des ersten Teils: Eine dreidimensionale Faltung (grün) kombiniert mit *Max-Pooling* entlang der spektralen Dimension (blau). Für eine übersichtlichere Darstellung werden nur eine Aktivierungskarte und ein Faltungskern gezeigt [2].

Diese Aktivierungskarten werden am Eingang des zweiten Teils entlang der spektralen Dimension hintereinander gehängt, sodass ein Datenwürfel entsteht, der aus vielen Aktivierungskarten besteht, die die örtliche Auflösung des Hyperspektralbilds besitzen. Darauf folgen vollständig verbundene Schichten entlang der spektralen Dimension, die auch als 1×1 -Faltungen bezeichnet werden können (siehe Abb. 4).

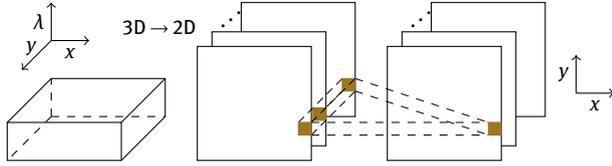


Abb. 4: Datentransformation zwischen erstem und zweitem Teil des neuronalen Netzes ($3D \rightarrow 2D$) und eine Schicht des zweiten Teils: Eine vollständig verbundene Schicht entlang der spektralen Dimension (braun) [2].

Die Sigmoidfunktion $\sigma : \mathbb{R} \rightarrow \mathbb{R}$ wird in jeder Schicht als Aktivierungsfunktion

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad (12)$$

verwendet. Am Ausgang findet eine Normierung mit

$$\hat{a}_p = \frac{a_p^*}{\sum_{\tilde{p}=1}^P a_{\tilde{p}}^*} \quad \text{für } p = 1, 2, \dots, P \quad (13)$$

statt. In Kombination mit der Sigmoidfunktion werden so die Nebenbedingungen (1) und (2) durch die Netzstruktur erzwungen. In [2] wurde gezeigt, dass dies zu besseren Ergebnissen führt als die Verwendung einer *Softmax*-Schicht am Ausgang. Das vorgestellte KNN soll mit einem Datensatz aus künstlich gemischten Reinspektren trainiert werden, der im nächsten Abschnitt vorgestellt wird.

4 Künstliche Mischspektren

Die zugrundeliegenden Daten für die Erzeugung der künstlichen Mischspektren sind echte Spektren der Reinstoffe. Da die beteiligten Stoffe bekannt sind, können die Spektren im Vorfeld mit wenig Aufwand aufgenommen werden. So existiert für jeden der P Reinstoffe eine Menge $\mathcal{M}_p = \{\mathbf{m}_{p1} \dots \mathbf{m}_{pI}\}$ an I Reinspektren, die der Spektrenvariabilität unterliegen.

Für die künstlichen Mischspektren werden nun zufällig Spektren aus den Mengen gezogen, wobei jedes Spektrum einer Menge mit der gleichen Wahrscheinlichkeit gezogen wird. Diese werden dann mit Hilfe der Mischmodelle LMM (3), FM, GBM (4) und LQM (5) gemischt. Es wird bei der Mischung eine Normierung mit der Summe aller Vorfaktoren der Spektren durchgeführt. Ein mit dem GBM erzeugtes Mischspektrum wird demnach mit

$$\left(1 + \sum_{p=1}^{P-1} \sum_{q=p+1}^P \gamma_{pq} a_p a_q \right)^{-1} \quad (14)$$

multipliziert; ein Mischspektrum, das mit dem LQM erzeugt wurde, mit

$$\left(1 + \sum_{p=1}^P \sum_{q=1}^P b_{pq} \right)^{-1}. \quad (15)$$

Dies ist bei den nichtlinearen Mischmodellen erforderlich, da sich die Vorfaktoren im Gegensatz zum LMM nicht zu 1 addieren. Da jedoch das Licht, welches zum linearen Anteil beiträgt, nicht zusätzlich zum quadratischen Anteil beitragen kann, ist diese Normierung sinnvoll. Damit ergeben auch Werte für γ_{pq} und b_{pq} Sinn, die größer als 1 sind.

Für die Mischmodelle GBM und LQM müssen geeignete Nichtlinearitätskoeffizienten gefunden werden. In diesem Beitrag wird dafür ein fester Wert verwendet, wodurch für alle p und q gilt: $\gamma_{pq} = \gamma$ und $b_{pq} = b$. Dies ist eine Einschränkung, besonders beim LQM, jedoch können damit bessere Ergebnisse erzielt werden als bei den Mischmodellen ohne zusätzliche Parameter. Um einen geeigneten Wert zu ermitteln, ist ein kleiner Validierungsdatsatz mit einigen Mischungen erforderlich.

Die Erzeugung der künstlichen Mischspektren erfolgt so, dass eine vorgegebene Anzahl an Realisierungen eines Mischverhältnisses gleichzeitig erstellt wird, die zu einem hyperspektralen Bild zusammengesetzt werden. Während des Trainings werden dem neuronalen Netz in jeder Iteration eine bestimmte Anzahl dieser Bilder gleichzeitig zugeführt. So kann auch nur ein einzelnes Spektrum je Mischverhältnis erzeugt werden.

Bei KNN wird das Training in sogenannte Epochen eingeteilt, wobei in jeder Epoche dem Netz jeder Datenpunkt des Trainingsdatensatzes einmal zugeführt wird. Weil hier die Daten zufällig generiert werden, endet eine Epoche, nachdem aus allen vorgegebenen Mischverhältnissen einmal Spektren erzeugt wurden. Da über viele Epochen trainiert wird, werden dem KNN viele Realisierungen der Mischverhältnisse zugeführt.

Im nächsten Abschnitt werden diese Vorgehensweisen anhand von Beispieldatensätzen untersucht.

5 Ergebnisse

Nach Vorstellung der verwendeten Datensätze wird untersucht, welches Mischmodell sich für den jeweiligen Datensatz am besten eignet und wie gut die Vorgehensweise im Vergleich zu den Verfahren aus Abschnitt 2 abschneidet. Bei nichtlinearen Mischmodellen wird zusätzlich der feste Nichtlinearitätskoeffizient (vgl. Abschnitt 4) bestimmt.

5.1 Datensätze

Die Datensätze zur Auswertung wurden im eigenen Bildverarbeitungslabor erstellt. Dazu wurden Pulver in verschiedenen Mischverhältnissen gemischt und in Setzkästen gefüllt. Es handelt sich bei den Pulvern im ersten Datensatz um farbige Quarzsande und beim zweiten um fein gemahlene Gewürze. Für alle erstellten Aufnahmen wurde ein Weißbildabgleich mit Hilfe eines Reflexionsnormals durchgeführt. Darüber hinaus wurden keine Vorverarbeitungen durchgeführt. Es werden nur Pixel der Proben verwendet, keine Pixel der Ränder der Setzkästen. Es wurden bei allen Aufnahmen 91 Wellenlängenkanäle mit einer Bandbreite von ca. 4 nm verwendet, die sich über den Wellenlängenbereich von 450 nm bis 810 nm erstrecken.

5.1.1 Quarzsande

Der Quarzsande-Datensatz besteht aus drei verschiedenen farbigen Quarzsanden. Von jedem Reinstoff gibt es 400 Spektren, wobei je 300 für die künstlichen Mischungen im Trainingsdatensatz verwendet werden. Die verbleibenden 100 Spektren sind Teil des Testdatensatzes, der Spektren echter Mischungen beinhaltet. Für diese Mischungen wurden die Materialanteile in Stufen von 0,125 variiert, womit sich insgesamt 45 Mischungen ergeben (siehe Abb. 5). Von jeder Mischung sind 100 Spektren im Testdatensatz. Der Testdatensatz ist als einzelnes Hyperspektralbild realisiert, das alle Mischungen enthält (siehe Abb. 6).

5.1.2 Gewürzmischungen

Der Gewürzmischungen-Datensatz besteht aus elf verschiedenen Gewürzen. Von jedem Reinstoff gibt es 576 Spektren, wobei je 384 für die künstlichen Mischungen im Trainingsdatensatz verwendet werden. Die verbleibenden 192 Spektren sind Teil des Testdatensatzes, der ebenfalls Spektren echter Mischungen beinhaltet. Für diese Mischungen wurden die Materialanteile von bis zu vier Rein-



Abb. 5: RGB-Bild der Setzkästen. Links: Quarzsande. Rechts: Gewürzmischungen (Ausschnitt).

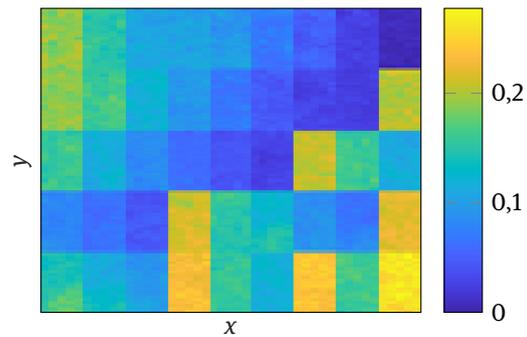


Abb. 6: Reflektanz des Testdatensatzes der Quarzsande bei 506 nm.

stoffen in Stufen von 0,125 variiert, wobei aufgrund der großen Anzahl an Kombinationsmöglichkeiten nur 155 Mischungen realisiert wurden (siehe Abb. 5). Darunter sind zusätzlich Mischungen von je drei Reinstoffen zu gleichen Anteilen. Auch hier ist der Testdatensatz als einzelnes Hyperspektralbild realisiert.

5.2 Vergleich der Entmischungsverfahren

Die Testdaten der beiden Datensätze sollen nun spektral entmischt werden. Dabei wird das mit den künstlichen Mischspektren aus Abschnitt 4 trainierte CNN aus Abschnitt 3 mit Methoden aus Abschnitt 2 verglichen. Als Gütemaß wird der Effektivwert des Fehlers (engl. *root-mean-square error*)

$$\Delta_{\text{RMSE}} = \sqrt{\frac{1}{N} \sum_{n=1}^N \frac{1}{P} \sum_{p=1}^P (\hat{a}_{pn} - a_{pn})^2}, \quad (16)$$

bezogen auf alle N Spektren eines Testdatensatzes, verwendet. Die Nebenbedingungen (1) und (2) werden von allen Verfahren, die hier verglichen werden, eingehalten.

An Verfahren, die auf dem LMM basieren, werden das AS-Verfahren und der FCLS-Algorithmus im Vergleich be-

trachtet. Zur Entmischung mit Hilfe von den nichtlinearen Mischmodellen GBM und LQM wurde ein gradientenbasiertes Liniensuchverfahren verwendet. Als Initialisierung wurde in beiden Fällen das mit dem AS-Verfahren gefundene Optimum verwendet, wie es in [11] vorgeschlagen wird. Für die auf dem ELMM und GLMM basierenden Entmischungsverfahren wurde als Initialisierung die mit dem FCLS-Algorithmus gefundene Lösung verwendet. Für all diese Verfahren werden in der Reinspektrenmatrix \mathbf{M} die mittleren Spektren aller aufgenommenen Reinspektren der jeweiligen Reinstoffe genutzt.

Das hier verwendete CNN besteht aus einem ersten Teil mit drei Schichten, wie sie in Abb. 3 zu sehen sind. Auf diese folgt ein zweiter Teil bestehend aus zwei Schichten nach Abb. 4. In Tab. 1 ist die Anzahl der Ausgangsaktivierungskarten der Schichten zu sehen.

Tab. 1: Anzahl der Ausgangsaktivierungskarten: Diese entspricht der Anzahl der Eingangsaktivierungskarten der nächsten Schicht mit Ausnahme der Schicht 4 (Anzahl der Eingangsaktivierungskarten in Klammern, vgl. Abschnitt 3).

Faltungsschicht	1	2	3	4	5
Quarzsande	32	64	128	(1408) 64	3
Gewürzmischungen	32	64	128	(1408) 66	11

Für die dreidimensionalen Filterkerne in den ersten Schichten erwies sich eine spektrale Filtergröße von $w_\lambda = 9$ als gut. Die örtliche Filtergröße wurde bei $w_{xy} = 1$ belassen, da beim Erzeugen der künstlichen Mischspektren keine Nachbarschaftsrelationen beachtet werden, weshalb der Einbezug der örtlichen Nachbarschaft, wie er in [2] verwendet wird, nicht sinnvoll ist.

Die künstlichen Mischspektren werden, wie in Abschnitt 4 beschrieben, während des Trainings des CNN erzeugt. Dabei werden in jedem Trainingsschritt alle Mischverhältnisse, die vorher definiert wurden, berücksichtigt, indem zu jedem Mischverhältnis mindestens ein zufälliges Spektrum erzeugt wird. Für die hier gezeigten Ergebnisse wurden die Anteile in Stufen von 0,125 variiert, wobei bei den Gewürzmischungen maximal vier Komponenten verwendet wurden. Als Mischmodelle werden das LMM (CNN-LMM), das FM (CNN-FM), das GBM (CNN-GBM) und das LQM (CNN-LQM) verwendet. Für die Auswertung wird das CNN jeweils fünfmal trainiert, da die Spektren zufällig erzeugt werden.

Da beim CNN-GBM und CNN-LQM noch die Nichtlinearitätskoeffizienten γ und b (vgl. Abschnitt 4) bestimmt werden müssen, wird ein Teil des Testdatensatzes dafür verwendet, diese zu bestimmen (Validierung). Bei den Quarz-

sanden werden alle Mischungen, bei denen keine Komponente mit einem Anteil kleiner oder gleich 0,125 enthalten ist, zur Bestimmung verwendet, die restlichen für die Auswertung. Das ergibt 24 Mischungen im Validierungs- und 21 Mischungen zu je 100 Spektren im Testdatensatz. Bei den Gewürzmischungen werden 80 Mischungen zu je 192 Spektren zur Bestimmung der Nichtlinearitätskoeffizienten genutzt und 75 für die Auswertung. Zur besseren Vergleichbarkeit werden alle Verfahren mit diesem verkleinerten Teil des Testdatensatzes ausgewertet (Abb. 7 und Abb. 8).

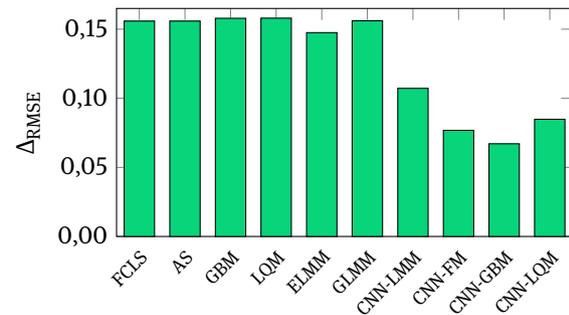


Abb. 7: Vergleich der verschiedenen Verfahren für die Quarzsande. Für die Entmischung der Quarzsande wurden $\gamma = 1,25$ und $b = 0,042$ ermittelt.

In Abb. 7 sind die Δ_{RMSE} dargestellt, die durch die zuvor beschriebenen Verfahren bei der spektralen Entmischung der Quarzsande erreicht werden. Ausgehend von den auf dem LMM basierten Verfahren FCLS und AS kann die Verwendung von nichtlinearen Mischmodellen keine Verbesserung herbeiführen. Das liegt zum einen daran, dass diese Modelle zu einem nicht-konvexen Optimierungsproblem führen und zum anderen, dass auch hier ein ähnlicher Modellfehler vorliegt. Bei den auf dem LMM basierenden Verfahren, die die Spektrenvariabilität miteinbeziehen, kann vor allem das ELMM eine leichte Verbesserung erzielen. Mit dem GLMM hingegen gelingt das nicht, was an der großen Anzahl an Freiheitsgraden liegt.

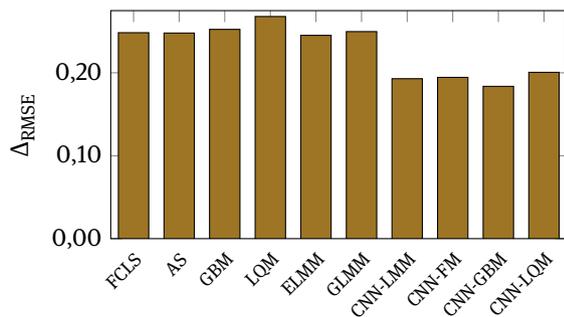
Das Erzeugen künstlicher Spektren und anschließende Training des CNN führt in allen Fällen zu einer Verbesserung. Dies gilt auch bei der Verwendung von LMM und FM, bei denen kein Parameter festzulegen ist. Die Verwendung des GBM führt hier noch einmal zu einer Verbesserung. Das LQM wird vom FM und GBM übertroffen. Dies liegt daran, dass ein fester Nichtlinearitätskoeffizient für alle Mischungen gewählt wurde (vgl. Abschnitt 4), was beim LQM zur Folge hat, dass alle quadratischen Anteile denselben Koeffizienten bekommen, während beim GBM und FM die Anteile in die Koeffizienten einfließen. In Ta-

Tab. 2: Effektivwert des Fehlers, getrennt nach Validierungs- und Testdatensatz für CNN-GBM und CNN-LQM.

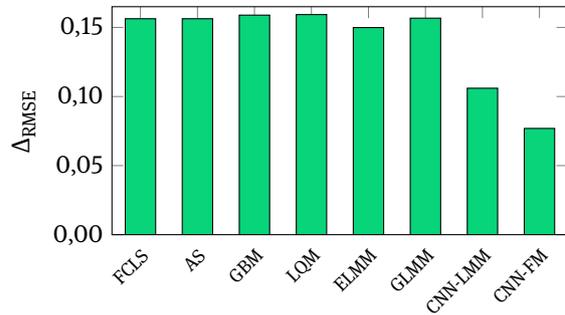
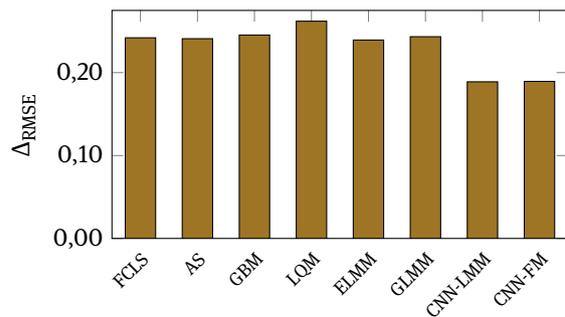
CNN	Quarzsande		Gewürzmischungen	
	-GBM	-LQM	-GBM	-LQM
Validierung	0,0675	0,0721	0,1761	0,1937
Test	0,0671	0,0848	0,1837	0,2006

belle 2 wird dies bestätigt. Beim GBM liefern die Auswertungen von Test- und Validierungsdatensatz ähnlich gute Werte, während beim LQM eine Verschlechterung beim Auswerten des Testdatensatzes zu erkennen ist.

Bei den Ergebnissen zu den Gewürzmischungen (Abb. 8) sind ähnliche Beobachtungen zu machen. Jedoch sind hier die Ergebnisse insgesamt schlechter, was an der schwereren Problemstellung zu liegen scheint, da sich im Vergleich zu den Quarzsanden alle Verfahren verschlechtert haben. Darüber hinaus ist die relative Verbesserung durch die Verwendung künstlicher Mischspektren in Verbindung mit CNN schwächer ausgeprägt. Trotzdem ist auch hier, selbst bei der Verwendung des LMM, eine Verbesserung erkennbar. Eine weitere Verbesserung durch Verwendung eines nichtlinearen Mischmodells ist hier nur beim GBM gegeben und auch geringer ausgeprägt. Bei einem Datensatz mit einer großen Anzahl an Kombinationsmöglichkeiten fällt die starke Vereinfachung, nur einen Nichtlinearitätskoeffizienten zu wählen, stärker ins Gewicht und führt bei Verwendung des LQM sogar zu einer Verschlechterung.

**Abb. 8:** Vergleich der verschiedenen Verfahren für die Gewürzmischungen. Für die Entmischung der Gewürzmischungen wurden $\gamma = 2,50$ und $b = 0,002$ ermittelt.

Ergänzend werden in den Abbildungen 9 und 10 für die Verfahren, die ohne vorherige Bestimmung eines Nichtlinearitätskoeffizienten auskommen, die Ergebnisse für den gesamten Testdatensatz (45 bzw. 155 Mischungen) gezeigt. Hierbei wird deutlich, dass die Verfahren für wei-

**Abb. 9:** Vergleich der verschiedenen Verfahren ohne Nichtlinearitätskoeffizient für die Quarzsande bei Verwendung des vollständigen Testdatensatzes.**Abb. 10:** Vergleich der verschiedenen Verfahren ohne Nichtlinearitätskoeffizient für die Gewürzmischungen bei Verwendung des vollständigen Testdatensatzes.

tere Daten nahezu identische Werte liefern und daher ähnlich gut abschneiden.

6 Zusammenfassung

In dieser Arbeit wird ein Verfahren vorgestellt, bei dem künstliche Mischspektren aus einer Menge an Reinspektren erzeugt werden. Diese Mischspektren können dann zum Training eines CNN für die spektrale Entmischung verwendet werden. Dabei wird durch das Verwenden von zufälligen Kombinationen aus Mengen an Reinspektren die Spektrenvariabilität berücksichtigt. Im Vergleich zu Verfahren aus der Literatur, die zum Teil ebenfalls die Spektrenvariabilität miteinbeziehen, können bei allen verwendeten Mischmodellen bessere Ergebnisse bezüglich des Fehlers der geschätzten Materialanteile erzielt werden. Die Voraussetzung ist das Vorhandensein mehrerer Spektren eines jeden Reinstoffes, weshalb sich das Verfahren vor allem zur Verwendung im industriellen Umfeld, wo die beteiligten Stoffe bekannt sind, eignet. Die besten Ergeb-

nisse konnten unter Verwendung des GBM erzielt werden, wobei hier einige Spektren von Stoffgemischen zur Ermittlung eines geeigneten Nichtlinearitätskoeffizienten benötigt werden. Liegen diese nicht vor, kann das LMM herangezogen werden, welches keinen Nichtlinearitätskoeffizienten besitzt. Alternativ kann das FM verwendet werden, wo die Nichtlinearitätskoeffizienten ausschließlich von den Materialanteilen abhängen.

Zukünftig könnte dieser Ansatz dadurch erweitert werden, dass statt vorgegebener Mischmodelle künstliche neuronale Netze zur Erzeugung der Mischspektren eingesetzt werden. Dafür sind vor allem *Variational Autoencoder* und bedingte *Variational Autoencoder* interessant. Bei letzteren könnten unterschiedliche Mischverhältnisse per Bedingung eingestellt werden. Zum Trainieren dieser Netze wäre jedoch ein umfangreicherer Trainingsdatensatz nötig. Es wäre daher zu prüfen, ob eine Erstellung weiterer Trainingsdaten für die spektrale Entmischung zu einer Verbesserung führt.

Literatur

1. J. Anastasiadis und F. Puente León. Detektion von Stoffen in Lebensmitteln mit Hilfe von 3D-Faltungsautoencodern. *tm – Technisches Messen*, 85 (s1): 38–44, 2018.
2. J. Anastasiadis und F. Puente León. Ortsaufgelöste spektrale Entmischung mit Hilfe von konvolutionalen neuronalen Netzen. *tm – Technisches Messen*, 86 (S1): 122–126, 2019. 10.1515/teme-2019-0062.
3. J. Anastasiadis, W. Krippner und F. Puente León. Spatially resolved ingredient detection in spice mixes using 3D convolutional neural networks. In *OCM 2019 – Optical Characterization of Materials: Conference Proceedings*, S. 35. KIT Scientific Publishing, 2019.
4. S. Bauer, J. Stefan und F. Puente León. Hyperspectral image unmixing involving spatial information by extending the alternating least-squares algorithm. *tm – Technisches Messen*, 82 (4): 174–186, 2015.
5. C. Chen, F. Jiang, C. Yang, S. Rho, W. Shen, S. Liu und Z. Liu. Hyperspectral classification based on spectral-spatial convolutional neural networks. *Engineering Applications of Artificial Intelligence*, 68: 165–171, 2018.
6. R. N. Clark und T. L. Roush. Reflectance spectroscopy: Quantitative analysis techniques for remote sensing applications. *Journal of Geophysical Research: Solid Earth*, 89 (B7): 6329–6340, 1984.
7. N. Dobigeon, Y. Altmann, N. Brun und S. Moussaoui. Linear and nonlinear unmixing in hyperspectral imaging. In *Data Handling in Science and Technology*, Band 30, S. 185–224. Elsevier, 2016.
8. L. Drumetz, M.-A. Veganzones, S. Henrot, R. Phlypo, J. Chanussot und C. Jutten. Blind hyperspectral unmixing using an extended linear mixing model to address spectral variability. *IEEE Transactions on Image Processing*, 25 (8): 3890–3905, 2016.
9. W. Fan, B. Hu, J. Miller und M. Li. Comparative study between a new nonlinear model and common linear model for analysing laboratory simulated-forest hyperspectral data. *International Journal of Remote Sensing*, 30 (11): 2951–2962, 2009.
10. A. Gowen, C. O'Donnell, P. Cullen, G. Downey und J. Frias. Hyperspectral imaging – an emerging process analytical tool for food quality and safety control. *Trends in Food Science & Technology*, 18 (12): 590–598, 2007.
11. A. Halimi, Y. Altmann, N. Dobigeon und J.-Y. Tourneret. Nonlinear unmixing of hyperspectral images using a generalized bilinear model. *IEEE Transactions on Geoscience and Remote Sensing*, 49 (11): 4153–4162, 2011.
12. B. Hapke. Bidirectional reflectance spectroscopy: 1. Theory. *Journal of Geophysical Research: Solid Earth*, 86 (B4): 3039–3054, 1981.
13. D. Heinz, C.-I. Chang und M. L. Althouse. Fully constrained least-squares based linear unmixing. In *IEEE 1999 International Geoscience and Remote Sensing Symposium*, Band 2, S. 1401–1403. IEEE, 1999.
14. T. Imbiriba, R. A. Borsoi und J. C. M. Bermudez. Generalized linear mixing model accounting for endmember variability. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, S. 1862–1866. IEEE, 2018.
15. S. Ioffe und C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.
16. N. Keshava und J. F. Mustard. Spectral unmixing. *IEEE signal processing magazine*, 19 (1): 44–57, 2002.
17. W. Krippner und F. Puente León. Bandbereichswahl und Materialanteilsschätzung mithilfe von Spektralfiltern. *tm – Technisches Messen*, 85 (6): 454–467, 2018.
18. W. Krippner, S. Bauer und F. Puente León. Ortsaufgelöste optische Bestimmung von Materialanteilen in Mischungen. *tm – Technisches Messen*, 84 (3): 207–215, 2017.
19. W. Krippner, S. Bauer und F. Puente León. Considering spectral variability for optical material abundance estimation. *tm – Technisches Messen*, 85 (3): 149–158, 2018.
20. W. Krippner, J. Anastasiadis und F. Puente León. Robust iterative estimation of material abundances based on spectral filters exploiting the svd. In *Algorithms, Technologies, and Applications for Multispectral and Hyperspectral Imagery XXV*, Band 10986, S. 109861T. International Society for Optics and Photonics, 2019.
21. A. Krizhevsky, I. Sutskever und G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, S. 1097–1105, 2012.
22. Y. LeCun, Y. Bengio et al. Convolutional networks for images, speech, and time series. In *The handbook of brain theory and neural networks*, 3361 (10): 1995, 1995.
23. I. Meganem, P. Déliot, X. Briottet, Y. Deville und S. Hosseini. Linear–quadratic mixing model for reflectances in urban environments. *IEEE Transactions on Geoscience and Remote Sensing*, 52 (1): 544–558, 2013.
24. R. B. Singer und T. B. McCord. Mars-large scale mixing of bright and dark surface materials and implications for analysis of spectral reflectance. In *Lunar and Planetary Science Conference Proceedings*, Band 10, S. 1835–1848, 1979.

25. M. O. Smith, J. B. Adams and D. E. Sabol. Spectral mixture analysis-new strategies for the analysis of multispectral data. In *Imaging spectrometry – a tool for environmental observations*, S. 125–143. Springer, 1994.
26. M. Wang, M. Zhao, J. Chen and S. Rahardja. Nonlinear unmixing of hyperspectral data via deep autoencoder networks. *IEEE Geoscience and Remote Sensing Letters*, 1–5, 2019.
27. S. Yu, S. Jia and C. Xu. Convolutional neural networks for hyperspectral image classification. *Neurocomputing*, 219: 88–98, 2017.
28. X. Zhang, Y. Sun, J. Zhang, P. Wu and L. Jiao. Hyperspectral unmixing via deep convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, 15 (11): 1755–1759, 2018.



Philipp Benzing

Karlsruher Institut für Technologie, Institut für Industrielle Informationstechnik, Karlsruhe, Germany
philippbenzing@gmail.com

Philipp Benzing studiert am Karlsruher Institut für Technologie (KIT) im Masterstudiengang Mechatronik und Informationstechnik. Bis April 2018 schrieb er seine Bachelorarbeit am Institut für Industrielle Informationstechnik (IIIT) des KIT. Seit August 2018 arbeitet er als wissenschaftliche Hilfskraft am IIIT.

Autoreninformationen



Johannes Anastasiadis

Karlsruher Institut für Technologie, Institut für Industrielle Informationstechnik, Karlsruhe, Germany
anastasiadis@kit.edu

Johannes Anastasiadis hat am Karlsruher Institut für Technologie (KIT) Elektro- und Informationstechnik studiert. Seit 2017 arbeitet er als wissenschaftlicher Mitarbeiter mit dem Ziel der Promotion am Institut für Industrielle Informationstechnik (IIIT) des KIT. Seine aktuelle Forschung beschäftigt sich mit künstlichen neuronalen Netzen und deren Anwendung zur Verarbeitung hyperspektraler Bilder.



Fernando Puente León

Karlsruher Institut für Technologie, Institut für Industrielle Informationstechnik, Karlsruhe, Germany
fernando.puente-leon@kit.edu

Fernando Puente León leitet das Institut für Industrielle Informationstechnik (IIIT) am Karlsruher Institut für Technologie (KIT). Hauptarbeitsgebiete: Mess- und Automatisierungstechnik, Signalverarbeitung, automatische Sichtprüfung und Bildverarbeitung, Mustererkennung und Informationsfusion.