Neuronale Netze in der Videoproduktion

Laura Anger
Technische Hochschule Köln
Institut für Medien- und Phototechnik
laura.anger@th-koeln.de

Vera Brockmeyer
Technische Hochschule Köln
Institut für Medien- und Phototechnik
vera.brockmeyer@smail.th-koeln.de

Zusammenfassung

Vera

Schlüsselwörter

Faltungsnetze, Videoproduktion

Vera

1. Einleitung

Vera

Die Videoproduktion konnte sich im Zuge der Digitalisierung im letzten Jahrzehnt enorm qualitativ verbessern. Jeder der vier Produktionabschnitte (siehe Abbildung 1) konnte verbessert und vor allem erleichtert werden. In der Konzeptionsphase konnten die Arbeitsprozesse mit Hilfe des Internets erleichtert werden durch den beschleunigten weltweiten Austausch von Skripten oder den standort-unabhängigen Zugriff auf cloudbasierte Projektmanagement Systeme.

Während der Produktion des Videomaterials unterstützen moderne digitale Kamerasysteme den Kameramann indem sie den Weißabgleich und die Belichtung automatisch berechnen und einstellen. Selbst geringfügige unruhige Bewegungen werden mit Bildstabilisatoren direkt unterdrückt.

Die darauffolgenden Arbeitsprozesse, wie das Schneiden und Editieren des Videomaterials währen der Postproduktionsphase, wurden in den letzten Jahren vereinfacht oder können teilweise durch zuverlässige Algorithmen automatisch durchgeführt werden. Mittlerweile können realistisch virtuelle Bildinhalte von CGI Experten mit entsprechender Rechenkapazität in das gedrehte Videomaterial nahtlos rendert werden. Dies ermöglicht es Produktionen fast ausschließlich im Studio zu produzieren und sogar aufwendige Fantasywelten oder aufwendige Stunts mit geringeren Kosten umzusetzen.

Doch gerade qualitativ hochwertige Videoproduktionen erfordern immer noch einen sehr hohen Arbeitsaufwand mit einer große Anzahl an professionellen Mitarbeitern und teuren Materialien. Einen großen Anteil daran hat die Postproduktion in der jede Szene separat editiert und an das Gesamtbild angepasst werden muss. Dieses Gesamtbild muss vorab genau festgelegt werden, denn eine spätere Korrektur erfordert eine vollständige Wiederholung der meisten Arbeitsschritte. Aber auch die Generierung von Bildmaterialien für kurze Schnittszenen oder Webvideos ist sehr zeitaufwändig und teuer. Ein mehrköpfiges Team mit dem umfangreichen Equipment muss zum Drehort gebracht werden.

In der Zukunft gilt es diesen Arbeits- und Kostenaufwand weiter zu reduzieren indem die einzelnen Arbeitsschritte automatisiert oder teil-automatisiert werden. Dies erfordert Ansätze die komplexe Zusammenhänge und Erfahrungen wie das menschliche Gehirn vereinen können. Sie sollten im idealen Fall Kreativität umsetzen, Bewegungen und Abläufe voraussagen, bekannte Eigenschaften sinnvoll kombinieren oder erlernte Informationen übertragen und anwenden können. Diese Anforderungen können mit einer Form von künstlichen Intelligenz, den neuronalen Netzen (NN) (siehe Abschnitt 2.3), erfolgreich erfüllt werden, die jenen des menschlichen Gehirn nachempfunden sind. In den letzten

Jahren wurden NN stetig weiterentwickelt und es konnten vor allem im Bereich der Medienproduktion bahnbrechende Erfolge erzielt werden. Die vielversprechendsten Erfolge konnten mit einer besonderen Form der NN erzielt werden. Diese Faltungsnetze (CNN) (siehe Abschnitt 2.3) ermöglichen orts- und skalierungs-unabhängige Operationen und somit ideal für mehrdimensionale digitale Signale.

Zu Beginn wurden CNN zur Objektklassifizierung eingesetzt um unter anderem automatisch Metadaten von Bild- oder Videodaten zu generieren und in Datenbanken oder Suchmaschinen einzupflegen. Ein bekannter Ansatz ist *Clarifai* [13], welcher eine Bibliothek mit konfigurierten CNN anbietet um optimierte Datenbanken anzulegen und zu verwalten. In den letzten Jahren wurden sie auch verstärkt für die Generierung oder Fortsetzung von bekannten Daten oder Signalen eingesetzt. Es konnten klassische Musikstücke sinnvoll beliebig verlängert werden [18] oder bewegte Sequenzen aus einzelnen Bildern generiert werden [19]. Auch in der Postproduktion konnten neue Bilder erstellt werden [10].

In den folgenden Kapiteln wird in den Grundlagen (siehe Kapitel 2) auf den allgemeine Ablauf in der Videoproduktion sowie detailliert die Funktionsweise der NN und CNN beschrieben. Im Anschluss werden in den folgenden Kapiteln verschiedene Entwicklungen von Videoproduktionsmittel vorgestellt, welche verschiedene Formen von NN und im besonderen von CNN nutzen. Drei Ansätze werden detailliert beschrieben und bewertet. Der erste Ansatz [19] beschreibt in Anschnitt 4.2 die Generierung von eine bewegten Bildsequenz aus einem Einzelbild. Die anderen Ansätze [10] [4] erläutert die Übertragung eines Bildstils auf eine andere Videosequenz.

2. Grundlagen

Laura

Um gezielter Ansatzpunkte für den Einsatz von NNs in der Videoproduktion aufzeigen zu können, wird diese in Kapitel 2.1 kurz beschrieben. Der Schwerpunkt dieses Kapitels liegt jedoch auf den Grundlagen von NNs und insbesondere CNNs, sowie deren Training.

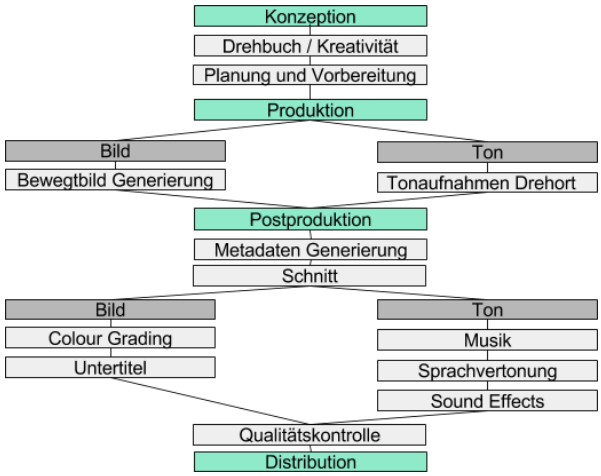

Abbildung 1: Grundlegende Arbeitsschritte einer Videoproduktion.

2.1. Videoproduktion

Laura

Mit der Videoproduktion oder auch Filmproduktion wird die Herstellung sowohl von Kino- als auch von Werbe- und Fernsehfilmen zusammengefasst. In Abbildung 1 ist ein Ablaufplan einer typischen Videoproduktion zu sehen. Da es alleine in Deutschland über 850 Produktionsformen gibt (Stand 2014) [15], kann der Ablaufplan nur einen sehr allgemeinen Überblick über die notwendigen Arbeitsschritte bieten.
Der erste Schritt, die Konzeption soll sowohl die Projektentwicklung, als auch die Vorproduktion zusammenfassen. Die sich anschließende Produktionsphase kann grob, wie im Schaubild zu sehen in Bild und Ton unterteilt werden, wobei diese beiden Bereiche nicht immer getrennt voneinander betrachtet werden sollten. Die Postproduktion besteht aus vielen verschiedenen Arbeitsschritten, deren Schwerpunkt auf dem Schnitt und der digitalen Bildnachbearbeitung liegt. Der Schritt der Distribution ist hier der Vollständigkeit halber erwähnt und beschreibt die Filmverwertung.
Der Aufbau der folgenden Ausführungen orientiert sich an Abbildung 1 (vgl. Kapitel 3, 4 und 5).

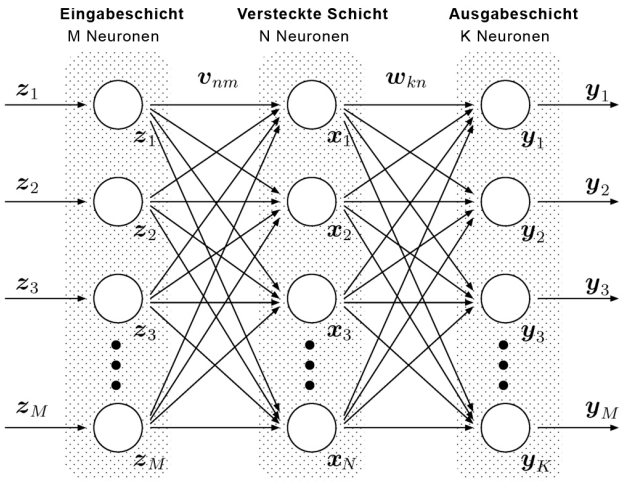

Abbildung 2: Prinzipieller Aufbau MLP nach [7].

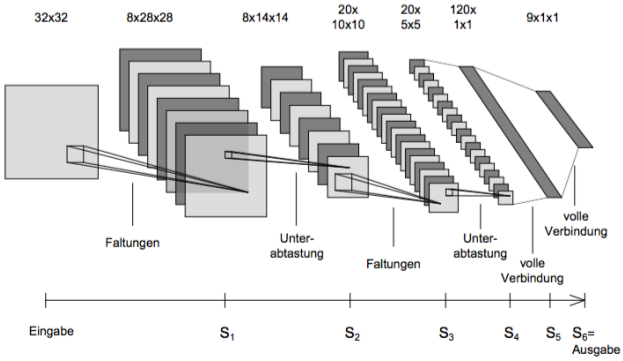

Abbildung 3: Prinzipieller Aufbau Faltungsnetz nach [12].

2.2. Neuronale Netze

Laura

NNs finden unter anderem Anwendung bei der Steuerung von Robotern, Börsenkursanalysen, Medizin oder Fahrzeugsteuerung. In der Bildverarbeitung werden NNs vor allem zur Klassifizierung genutzt.
Sie sind vom menschlichen Gehirn inspiriert, welches laut [6] ein nicht-lineares, komplexes und hoch paralleles System zur Verarbeitung von Informationen darstellt. Ähnlich wie dieses bestehen künstliche NNs aus einer Menge an simulierten Neuronen, die untereinander verbunden sind und in Schichten organisiert sind. Es gibt verschiedene Arten der Vernetzung, die, wie in [6] und [14] beschrieben, in rück- und vorwärts gekoppelte Modelle unterteilt werden können.

Am häufigsten kommen sogenannte *Multilayer Perceptrons* (MLP) [1][11][14] zum Einsatz. Ein generalisierter Aufbau ist beispielhaft in Abbildung 2 zu sehen. Dieses MLP besteht aus einer Eingabe- und Ausgabeschicht mit M bzw. K Neuronen und einer versteckten Schicht mit N Neuronen. Es handelt sich um ein vorwärtsgekoppeltes Modell, bei welchem jedes Neuron einer Schicht mit jedem Neuron der darauffolgenden Schicht verbunden ist. Dies nennt man volle Verbindung.
Die Eingangsschicht dient zum Verteilen der Daten z_m mit $m = 1, ..., M$. Die Ausgabe eines jeden Neurons in der versteckten Schicht, dargestellt durch x_n mit $n = 1, ..., N$, lässt sich durch Formel 1 berechnen. Hierbei steht v_{nm} für die jeweilige Gewichtung der Verbindungen zwischen den Neuronen der Eingabe- und der versteckten Schicht und f für die Aktivierungsfunktion [14][6] des jeweiligen Neurons.

$$x_n = f\left(\sum_{m=1}^{M} v_{nm} z_m\right) \tag{1}$$

Die Ausgangswerte y_k, mit $k = 1, ..., K$, lassen sich äquivalent unter Hereinnahme der Werte x_n und der Gewichte w_{kn}, sowie einer Aktivierungsfunktion g berechnen, und gelten als Vertrauenswerte. Sie müssen gemäß der Aufgabenstellung interpretiert werden.

2.3. Faltungsnetze

Laura

Im Folgenden wird genauer auf CNNs eingegangen, da diese die Grundlage, für die meisten der in den folgenden Kapitel vorgestellten Ansätze, bilden. Vereinfacht ausgedrückt besteht ein CNN aus einer Vernetzung von Faltungsoperationen mit unterschiedlichen Filtermasken. CNNs kommen, bedingt durch ihre Architektur, oft zum Einsatz, wenn große Datenmengen von einem NN verarbeitet werden sollen. Ein schematischer Aufbau ist in Abbildung 3 zu sehen.

Jedes Pixel eines Feldes, das auf der Abbildung zu sehen ist, wird durch ein Neuron repräsentiert. Die Felder sind in Schichten organisiert. Die Eingangsschicht fungiert, vergleichbar mit den MLPs aus Kapitel 2.2, als Verteiler der Information an die Neuronen der nächsten Schicht S_1. Die Besonderheit eines CNNs sind die sich abwechselnd durchgeführte Faltung und anschließende Unterabtastung. Zwischen den Schichten S_4 und S_6 ähnelt das Modell einem MLP, da die Neuronen schichtweise voll verbunden sind.
Im Allgemeinen wird für eine Faltung eine Filtermaske h, also ein endlicher, zweidimensionaler Koeffizientensatz, wie in Formel 2 zu sehen, verwendet. Hier-

bei stehen x und y jeweils für die horizontale bzw. die senkrechte Bildkoordinate. Die Anzahl der Koeffizienten a_{xy}, wird in der Horizontalen mit N_{hx} und im Vertikalen mit N_{hy} bezeichnet.

$$h(x,y) = \begin{cases} 0 \; für \; x < -\lfloor \frac{N_{hx}-1}{2} \rfloor \; \vee \; y < -\lfloor \frac{N_{hy}-1}{2} \rfloor \\ 0 \; für \; x > \lfloor \frac{N_{hx}-1}{2} \rfloor \quad \vee \; y > \lfloor \frac{N_{hy}-1}{2} \rfloor \\ a_{xy} \end{cases} \tag{2}$$

Formel 3 beschreibt die Faltung eines Eingangssignals s mit einer Filtermaske h, wobei I das Ausgangssignal in Abhängigkeit von x und y beschreibt.

$$I(x,y) = (s * h)(x,y) = \sum_{m_x=-\infty}^{\infty} \sum_{m_y=-\infty}^{\infty} s(m_x, m_y) \cdot h(x - m_x, y - m_y) \tag{3}$$

Im Fall eines CNNs wird die Faltung, die wie in Abbildung 3 zu sehen, beispielsweise zwischen der Eingangsschicht und S_1 vollzogen wird, durch die Verbindung zwischen den Neuronen zweier Felder modelliert. Dabei entsprechen die Gewichte der Neuronen genau den Filterkoeffizienten a_{xy}. Für ein jedes Feld sind diese Koeffizienten konstant, was bedeutet, dass alle Neuronen eines Feldes mit nur einem Gewicht auskommen. Dieses Prinzip nennt man geteilte Gewichte.
Im CNN wird nach jeder Faltung eine Unterabtastung durchgeführt um zu gewährleisten, dass die Dimension der Eingangsdaten schrittweise an die Dimension des Ausgangsvektors angepasst wird. Hierzu wird meist eine bilineare Unterabtastung um den Faktor 2 vorgenommen. Allgemeiner betrachtet werden nxn Werte zu einem Wert zusammengefasst.
Wie zu Beginn des Kapitels erwähnt, haben CNNs gegenüber den MLPs den Vorteil, dass sie nahezu beliebig hochskaliert werden können und somit gut geeignet für große Datenmengen sind. Dies liegt vor allem daran, dass die Neuronen nur lokal verbunden sind und sich somit das Prinzip der geteilten Gewichte zu Nutze gemacht werden kann. Ein weiterer Vorteil von CNNs, der vor allem in der Bildverarbeitung genutzt wird, ist das sie translationsinvariant sind.

2.4. Training Faltungsnetze

Laura

Meistens werden CNNs mittels der *back-propagation* Methode trainiert. Bei dieser überwachten Lernmethode bedarf es einer großen Menge an vorher klassifizierten Eingabematerialien [9].
In den Faltungsschichten kann der Fehler der vorangegangen Schicht nach Formel 4 berechnet werden. Dabei steht E für den Fehler in der jeweiligen Schicht l gemacht wird. Während x^l für die Eingabe in die Schicht steht, bezeichnet y^l die Ausgabe der entsprechenden Schicht. Die Größe der Eingabe wird der Einfachheit halber als quadratisch, also mxm-groß angenommen. Die Gewichtung wird mit w bezeichnet. Um die Formel in der Realität anzuwenden, muss die linke und obere Grenze de Eingabeinhaltes, z.B. eines Bildes, mit Nullen ergänzt werden. Ansonsten wäre es nicht möglich den Fehler für Pixel zu berechnen, welche näher als m an den entsprechenden Rändern liegen.

$$\begin{aligned} \frac{\delta E}{\delta y_{ij}^{l-1}} &= \sum_{a=0}^{m-1} \sum_{b=0}^{m-1} \frac{\delta E}{\delta x_{(i-a)(j-b)}^{l}} \frac{\delta x_{(i-a)(j-b)}^{l}}{\delta y_{ij}^{l-1}} \\ &= \sum_{a=0}^{m-1} \sum_{b=0}^{m-1} \frac{\delta E}{\delta x_{(i-a)(j-b)}^{l}} w_{ab} \end{aligned} \tag{4}$$

Die Schichten, in denen die Unterabtastung stattfindet, leisten kaum Beitrag zum eigentlichen Lernprozess des CNNs. Hier wird das Problem allerdings reduziert, da nxn Werte in einem einzigen resultieren.
Weil alle Gewichtungen w mittels des *back-propagation* Algorithmus während des Trainings angepasst werden, können CNNs laut LeCun als Erzeuger ihrer eigenen Merkmalextraktion gesehen werden [8].

3. Konzeption

Vera

Vor der eigentlichen Videoproduktion muss das Vorhaben zunächst konzipiert und detailliert geplant werden. Dies bezieht sich vor allem auf die kreativen Prozesse des Drehbuchschreibens und darauffolgend die gesamte Projektplanung und -vorbereitung. Naturgemäß sind NN weniger sinnvoll für die Planung von Projekten. Doch in den letzten Jahren wurde damit begonnen zu erforschen, ob sich mit Hilfe von NN kreative Prozesse umsetzen lassen und sie eine eigene Kreativität entwickeln können. In diesem Sinne ist Kreativität auch mit sinnvollen selbstständigem Denken oder Entscheiden gleichzusetzen, was im Entste-

hungsprozess von Drehbüchern unumgänglich ist.

3.1. Aktueller Stand

Vera

Es gibt bereits erste Versuche mit Hilfe von NN automatisch sinnvolle Texte oder Dialoge zu erstellen, die auf bekannten Texten und Storylines basieren.

4. Produktion

Laura

muss angepasst werden

In diesem Kapitel werden zunächst Ansätze vorgestellt, die auf Grundlage von NNs Arbeitsschritte bei der Produktion von Videos übernehmen bzw. vereinfachen könnten (vgl. Kapitel 4.1). Dazu ist an zu merken, dass diese Ansätze meist in einem anderen Kontext entwickelt wurden und ggf. eine Anpassung an die Standards einer Produktion stattfinden müsste.

In Kapitel 4.2 wird ein Ansatz zur automatisierten Generierung von Szenendynamiken in Hinsicht auf Funktionsweise und Arbeitserleichterung für die Videoproduktion analysiert.

4.1. Aktueller Stand Musikgenerierung

Laura

Es gibt verschiedene Ansätze NNs zu nutzen, um Musik automatisiert generieren zu lassen. Im Folgenden werden drei Ansätze kurz vorgestellt, welche alle rückgekoppelte Neuronale Netze (RNN) benutzen.
An der *University of Washington* ist im Rahmen einer Projektarbeit ein Musik Generator namens *Algo Rythm* entstanden [16]. Für die Umsetzung habend die Studierenden um Timmerman RNNs geeignet trainiert. Der Quellcode kann auf Github [17] eingesehen werden.
Ein weiterer Ansatz, der in [3] beschrieben wird, arbeitet ebenfalls mit einem RNN, welches mit einem Autokorrelation-basierte Prädiktor kombiniert wird. Dabei soll die Struktur von Musikstücken erlernt werden, indem zunächst die folgende Note einer Tonreihenfolge vorausgesagt werden soll.
Der letzte Ansatz benutzt ebenfalls RNNs um auf Grundlage einer Notensequenz eine Musikstück zu komponieren [2]. Dabei wird die Eingabesequenz zunächst interpretiert. Anschließend sorgen zwei RNN basierte Algorithmen für die Produktion von sowohl Rhythmus, als auch die Vorhersage der nächsten Note.

Alle drei Ansätze weisen hohes Potential auf, wenn es darum geht Musikstücke automatisiert zu generieren. Über das Genre des zugeführten Musikmaterials lässt sich die gewünschte Ausgabe begrenzt steuern. Es wäre durchaus vorstellbar, so Musik für eine Filmproduktion zu generieren.

4.2. Szenendynamik nach Vondrick et al.

Laura

Mit dem Ansatz von Vondrick et al. [19] können aus Einzelbildern ganze Szenen Dynamiken erstellt werden, welche zum einen für die Klassifizierung und zum anderen für die Vorhersage von Bewegung genutzt werden kann. Das Resultat sind kleine Videos mit einer Auflösung von 65x64 Pixeln und 32 Einzelbildern.
Die Grundlage für dieses Verfahren bilden zwei Faltungsnetze, die als *Generative Adversarial Networks* (GAN) [5] fungieren. Die beiden Netze können als Gegenspieler angesehen werden. Während der Generator G für die Generierung von einer Videosequenz auf Grundlage eines Standbildes zuständig ist, soll der Diskriminator D zwischen den so erzeugten Videos und realen Videosequenzen differenzieren können. Um dies zu erreichen werden beide GANs, wie in Formel 1 des Papers [19] zu sehen gegeneinander trainiert. Das Minimierungs- bzw. Maximierungsproblem wird hierzu mittels Gradientenverfahren gelöst.
Das Training der beiden Netze erfolgt unüberwacht. Da sich ähnliche Objekte in der Regel auch ähnliche Bewegungsmuster aufweisen, lassen die Autoren das zur Verfügung stehende Videomaterial in Kategorien (Strand, Babys, Golf und Züge) einteilen. Zudem sorgt ein Stabilisierungsalgorithmus dafür, dass bei dem gesamten Testmaterial von einer statischen Kamera ausgegangen werden kann und die zu erlernende Szenendynamik nicht durch eine Bewegung der Kamera verfälscht wird. Für jede dieser Kategorien wird dann ein eigenes Pärchen bestehend aus Generator und Diskriminator trainiert.
Bevor die beiden Faltungsnetze G und D auf die beschriebene Art trainiert werden können, müssen sie

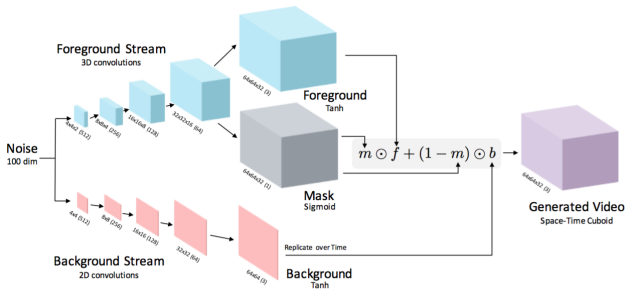

Abbildung 4: Aufbau Generator nach [19].

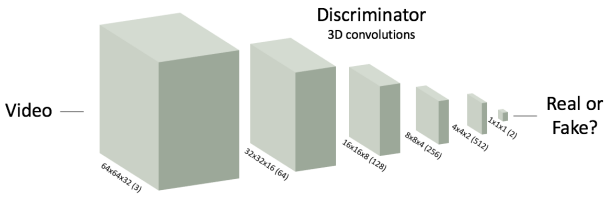

Abbildung 5: Aufbau Diskriminator nach [19].

erst einmal ihren Aufgaben entsprechend aufgebaut werden.

Den Aufbau des Generators, welchen Vondrick et al. genutzt haben, sieht man in Abbildung 4. Im Gegensatz zu dem in Kapitel 2.3 beschriebenen, gewöhnlichen CNN, wird hierbei keine Unterabtastung vorgenommen, sondern die Menge der Daten von Schicht zu Schicht erweitert. Zudem werden die Eingabedaten für die Verarbeitung in zwei Datenströme aufgeteilt. Während in dem eine Strom der statische Hintergrund entsprechend vergrößert wird, wird in dem anderen die Bewegung des sich im Vordergrund befindlichen Objektes vorhergesagt. Beide Teile werden dann am Ausgang nach Formel 5 zusammengefasst.

$$G(z) = m(z) \odot f(z) + (1 - m(z)) \odot b(z) \qquad (5)$$

Dafür ist die binäre Maske m so gewählt, dass sie überall den Wert eins hat, wo der Vordergrund übernommen werden soll und ansonsten nur Nullen enthält.

Der Diskriminator hat einen wesentlich simpleren Aufbau, der in Abbildung 5 zu sehen ist. Dabei muss die Eingabeschicht eben so groß sein, wie die Ausgabeschicht des Generators.

Die Bedeutung für so erzeugte Bewegtbildsequenzen für die Automatisierung der Produktion von Filmen, wird im nächsten Kapitel diskutiert.

4.3. Bewertung des Ansatzes von Vondrick

Laura

Das in Kapitel 4.2 beschriebene Verfahren schafft es automatisiert aus einem Einzelbild eine kurze und niedrig aufgelöste Bewegtbildsequenz zu erstellen. Hierbei wir vor allem die Szenendynamik in den meisten Fällen annähernd realistisch vorhergesagt, wobei die Modellierung des sich bewegenden Bildinhaltes eher mangelhaft ist. Trotz des hohen Bedarfs an Trainingsmaterial und den damit verbundenen Zeitaufwand, kann das System ohne menschliche Hilfe eigenständig Kategorisieren, Trainieren und letztendlich Szenendynamiken generieren. Gegen einen Einsatz in der Videoproduktion sprechen die geringe Auflösung und Zeitspanne die bisher abgedeckt werden kann. Zudem kann in einer Szene nicht immer nur von einem sich bewegenden Objekt ausgegangen werden. Und ein trainieren von Netzen für alle denkbaren Kategorien ist sehr aufwendig. Dennoch könnte der Ansatz als Grundlage für einen ausgereifteren Algorithmus dienen, der beispielsweise die Anweisungen des Drehbuchs mit einbezieht und somit eine Generierung von planbareren Szenen ermöglicht.

Der jetzige Stand des Algorithmus könnte beispielsweise für die Generierung von GIFs oder Simulationen genutzt werden. Zudem könnte der Ansatz, mit etwas Anpassung, anstatt Szenendynamiken in Form von Videos darzustellen, Bewegungsvektoren aus einem Standbild vorhersagen und somit in der Videocodierung Einsatz finden.

5. Postproduktion

Vera

5.1. Aktueller Stand

Vera

6. Zusammenfassung

Vera

Literatur

[1] H. Braun. *Neuronale Netze Optimierung durch Lernen und Evolution*. Springer, 1997.

[2] C. Browne. System and method for automatic music generation using a neural network architecture, Oct. 2 2001. US Patent 6,297,439.

[3] D. Eck and J. Lapalme. Learning musical structure directly from sequences of music. (1300), 2008.

[4] L. A. Gatys, A. S. Ecker, and M. Bethge. A neural algorithm of artistic style. *CoRR*, abs/1508.06576, 2015.

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[6] S. Haykin. *Neural Networks A Comprehensive Foundation*. Macmillan College Publishing Company, 1994.

[7] T. Isokawa, H. Nishimura, and N. Matsui. Quaternionic multilayer perceptron with local analyticity. *Information*, 3(4):756, 2012.

[8] Y. LeCun and Y. Bengio. The handbook of brain theory and neural networks. chapter Convolutional Networks for Images, Speech, and Time Series, pages 255–258. MIT Press, Cambridge, MA, USA, 1998.

[9] Y. LeCun, K. Kavukcuoglu, and C. Farabet. Convolutional networks and applications in vision. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, pages 253–256, May 2010.

[10] A. Mordvintsev. Inceptionism: Going deeper into neural networks. `https://research.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html`. Abgerufen: 16.08.2016.

[11] D. Nauck, F. Klawonn, and R. Kruse. *Neuronale Netze und Fuzzy Systeme*. Vieweg, 1994.

[12] M. Osadchy, Y. LeCun, and M. Miller. Synergistic face detection and pose estimation with energy-based models. *Journal of Machine Learning Research*, 8:1197–1215, 2007.

[13] T. Simonite. A startup's neural network can understand video. `https://www.technologyreview.com/s/534631/a-startups-neural-network-can-understand-video/`. Abgerufen: 16.08.2016.

[14] J. Stanley and E. Bate. *Neuronale Netze Computersimulation biologischer Intelligenz*. Systhema Verlag GmbH, 1991.

[15] statista. Anzahl der aktiven film- und fernsehproduktionsfirmen in deutschland in den jahren 1998 bis 2014. `https://de.statista.com/statistik/daten/studie/243238/umfrage/anzahl-der-film-und-fernsehproduktionsfirmen-in-deutschland/`. Abgerufen: 10.08.2016.

[16] G. Timmermann. Algo rhythm: Music composition using neural networks. `https://medium.com/@granttimmerman/algo-rhythm-music-composition-using-neural-networks-f89897ff2df7`. Abgerufen: 18.08.2016.

[17] G. Timmermann. Algorithmic music composition using artificial neural nets. `https://github.com/grant/algo-rhythm`. Abgerufen: 18.08.2016.

[18] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu. Wavenet: A generative model for raw audio. *CoRR*, abs/1609.03499, 2016.

[19] C. Vondrick, H. Pirsiavash, and A. Torralba. Generating videos with scene dynamics. *CoRR*, abs/1609.02612, 2016.