

# Musik-Visualisierung

## Wie sieht Musik aus?

MATURAARBEIT  
ANDRÉ RÖSTI (15GB)  
AUGUST 2014

PRODUKT MIT WERKBERICHT  
BETREUT DURCH TOBIAS SCHROPP  
GYMNASIUM THUN-SCHADAU

© 2014 André Rösti  
[www.andreroesti.com](http://www.andreroesti.com)

## Abstract

Thema dieser Maturaarbeit ist *Musik-Visualisierung*, die grafische Darstellung von Musik. Musik-Visualisierungen erweitern das Hören der Musik visuell. *Visual-Jockeys*<sup>1</sup> (VJs) und Programme wie *iTunes* nutzen diese unter anderem. Ziel der Arbeit war es, ein Computerprogramm zu erstellen, welches ästhetisch ansprechende Animationen auf Basis von Musik generieren kann. Änderungen in Lautstärke und Tonhöhe werden in der Animation als eine Änderung in Form oder Farbe dargestellt. Das Programm visualisiert die Musik und liefert damit eine mögliche Antwort auf die Leitfrage „Wie sieht Musik aus?“

Bei dieser Arbeit handelt es sich um ein Produkt mit Werkbericht. Das Programm, welches die Musik-Visualisierung generiert, ist das Produkt. Der vorliegende Werkbericht erklärt den Erarbeitungsprozess und die Funktionsweise des Programms und enthält weitere Informationen zum Thema.

Eine Herausforderung dieser Arbeit war die Übertragung physikalischer Eigenschaften wie Frequenz und Amplitude einer Schallwelle auf musikalische Eigenschaften wie Tonhöhe und Lautstärke. Diese war nur mit Einschränkungen möglich, unter anderem, da es sich bei vielen musikalischen Eigenschaften um subjektive Wahrnehmungen handelt, die sich von Person zu Person unterscheiden. Diese Einschränkungen machten auch die kreative Umsetzung von zuvor skizzierten Visualisierungen schwieriger. Weitere Aufgaben auf dem Weg zum Endprodukt waren die Umsetzung von Mechanismen zum Auslesen von digitalen Audio-Dateien (*RIFF WAVE*), die Umsetzung eines Algorithmus zum Berechnen der *Fourier-Transformation*<sup>2</sup>, die Wahl der richtigen *Grafikbibliothek*<sup>3</sup> zum Generieren des Videos und andere.

Das Endprodukt der Arbeit funktioniert und ist in der Lage Musik grafisch darzustellen. Zuvor skizzierte Ideen für Musik-Visualisierungen liessen sich umsetzen.

---

<sup>1</sup> Visual Jockeys (VJs): Videokünstler, die live bei Musikveranstaltungen auftreten

<sup>2</sup> Fourier-Transformation: Mathematische Transformation von der Zeit- in die Frequenzdomäne

<sup>3</sup> Grafikbibliothek: Ansammlung von Funktionen die das Erzeugen von Grafiken vereinfachen

## Vorwort

Das Thema „*Musik-Visualisierung*“ habe ich gewählt, da es zwei meiner Interessen verbindet; Zum einen betätige ich mich gerne kreativ, zum Beispiel mit Fotografie, und ich lege Wert auf gute Gestaltung. Zum anderen interessiert mich die Technik und Physik.

Mit dem Erstellen eines Visualisierungsprogramms konnte ich mich bei der Gestaltung der Visualisierungen kreativ betätigen, und mein Wissen in der Physik und dem Programmieren erweitern.

Ich hoffe, dass das Endprodukt - die Visualisierungen, die das Programm generiert - einige Leute zu begeistern vermag, auch wenn die technischen Hintergründe sehr theoretisch erscheinen mögen.

Mein Dank geht an Tobias Schropp, der diese Maturaarbeit betreut hat, Michael Spahr, der bereitwillig meine Fragen im Interview zur Video-Jockey-Kultur beantwortet hat, und meine Eltern, die diese Arbeit korrekturgelesen haben.

# Inhaltsverzeichnis

1	Einleitung	1
1.1	Musik-Visualisierung	1
2	Geschichte der Musik-Visualisierung	3
2.1	VJing	6
3	Inspiration	8
3.1	Michal Levy: Giant Steps	8
3.2	Esteban Diácono: Ljósið	9
3.3	Matthias Dittrich: Narratives 2.0	10
4	Erarbeitungsprozess	11
4.1	Übersicht über das Thema	11
4.2	Schaffen der technischen Grundlagen	11
4.3	Skizzieren und Umsetzen einer Visualisierung	13
5	Funktionsweise des Visualisierungsprogramms	14
5.1	Auslesen der Audio-Datei	16
5.1.1	Wahl des Dateiformates	16
5.1.2	Das RIFF-WAVE-Dateiformat	17
5.2	Analyse der Schallwelle	20
5.2.1	Fourier-Transformation	20
5.2.2	Fensterfunktion	23
5.2.3	Eigenschaften von Musik	24
5.3	Zeichnen der Visualisierung	26
5.3.1	Wahl einer Grafikbibliothek	26
5.3.2	Zeichnen mit Cairo	27
5.4	Einschränkungen	28
6	Ergebnisse	29
7	Quellen	31
7.1	Literatur	31
7.2	Internet	31
8	Eidesstattliche Erklärung	34
	Anhang A: Interview mit Michael Spahr	35

# 1 Einleitung

Wie sieht Musik aus? Diese Frage erscheint zunächst paradox - Musik hört man, aber sie ist meist unsichtbar. Sitzt man zu Hause vor der Stereoanlage oder geht man mit Kopfhörern durch die Stadt, so hat das, was man sieht, keinerlei Zusammenhang mit dem, was man hört.

Musik und Bild sind zwei Kunstformen, die Menschen stark emotional bewegen können. Was also, wenn wir diese zwei Medien kombinieren, und die Töne und Rhythmen von Musikstücken in bewegten Bildern grafisch darstellen? Was wäre das Resultat dieser Erweiterung des Hörsinns durch den Sehsinn?

## 1.1 Musik-Visualisierung

Schon Sir Isaac Newton hatte die Idee, verschiedenen Tonhöhen verschiedene Farben zuzuordnen.<sup>4</sup> Das war eine erste einfache Form der *Musik-Visualisierung*; Bei der Musik-Visualisierung werden Eigenschaften der Musik wie Tonhöhen und Lautstärke auf Farben, Formen und andere Eigenschaften einer Darstellung (Bild oder Animation) übertragen. Änderungen im Lied führen zu Änderungen der Darstellung.<sup>5</sup>

Ein Beispiel: Wird die Tonhöhe auf die Farbe eines Kreises im Bild übertragen, so könnte ein tiefer Ton dazu führen, dass der Kreis blau gefärbt wird, während ein hoher Ton den Kreis rot färben würde. Ein mittelhoher Ton, würde dazu führen, dass der Kreis in der Farbe eingefärbt wird, die zwischen rot und blau liegt - also violett. In diesem Beispiel wird die Tonhöhe grafisch als Farbe dargestellt. Dem Künstler bleibt die Entscheidung welche Farben zur Darstellung der Tonhöhen genutzt werden.

Dieses Konzept lässt sich kreativ vielseitig einsetzen; Der Künstler kann frei wählen, wie die musikalischen Eigenschaften auf die Grafik übertragen werden und kann damit auch die Wahrnehmung des Musikstückes durch seine gestalterischen Entscheidungen beeinflussen.

---

<sup>4</sup> URL: [www.hps.cam.ac.uk/library/universalharmony/newton.html](http://www.hps.cam.ac.uk/library/universalharmony/newton.html) (Stand: 07.08.2014)

<sup>5</sup> URL: [www.wisegeek.com/what-is-music-visualization.htm](http://www.wisegeek.com/what-is-music-visualization.htm) (Stand: 12.08.2014)

Dank Computertechnik ist es heute einfach, auf Basis von Musik automatisch Animationen zu generieren. Ein weit bekanntes Beispiel dafür ist die Visualisierung im Medienplayer iTunes.<sup>6</sup>

Ziel dieser Maturaarbeit war es, selbst ein Programm zu erstellen, welches Musik-Visualisierungen abhängig von der Musik automatisch generieren kann. Das Endprodukt sollte ästhetisch ansprechend sein und sich von bestehenden Visualisierungen abheben.

---

<sup>6</sup> URL: [de.wikipedia.org/wiki/Audiovisualisierung#Audiovisualisierung\\_als\\_Unterhaltungsmedium](https://de.wikipedia.org/wiki/Audiovisualisierung#Audiovisualisierung_als_Unterhaltungsmedium) (Stand: 12.08.2014)

## 2 Geschichte der Musik-Visualisierung

Es liegt in der Natur des Menschen, zu versuchen die Eindrücke verschiedener Sinne miteinander zu kombinieren. Das zeigt das neurologische Phänomen der *Synästheise*, das bei schätzungsweise 4% der Menschen auftritt. Bei diesem Phänomen tritt eine starke Kopplung zweier Sinneseindrücke auf; Die Personen nehmen beim Hören verschiedener Töne zusätzlich auch Farben wahr. Jeder Ton löst bei ihnen die Wahrnehmung einer anderen Farbe aus.<sup>7</sup> Gewissermaßen „sehen“ sie die Musik. Es ist also selbstverständlich, dass die Verbindung von Musik und Bild Künstler schon seit langer Zeit fasziniert.

Eine der ersten Beschreibungen eines Konzepts zur Musik-Visualisierung stammt vom berühmten Physiker Sir Isaac Newton. Nachdem er durch Experimente mit Prismen das Farbspektrum entdeckte, fielen ihm Gemeinsamkeiten zur Notenskala auf. Er ordnete in seiner 1704 veröffentlichten Arbeit „*Opticks*“ zum ersten Mal den Tonhöhen Farben zu. Zwar wurden die Gemeinsamkeiten, die Newton zu sehen glaubte, später widerlegt, aber die Zuordnung der Tonhöhen zu Farben blieb bestehen.<sup>8</sup> Andere Erfinder und Künstler nutzten die Idee und entwickelten die Zuordnung weiter.<sup>9</sup>

26 Jahre danach, im Jahre 1730, entwickelte Louis Bertrand Castel ein sogenanntes *Farbenklavier*. Beim Spielen dieses Klaviers wurde durch Drücken der Tasten auf der Klaviatur ein Mechanismus ausgelöst, welcher eine Farbe zeigte. Den Tönen C, D, E, F, G, A, H wies der Erfinder die Farben blau, grün, gelb, fauve, rot, violett, grau zu.<sup>10</sup>

In der jüngeren Geschichte ist Oskar Fischinger zu erwähnen. Fischinger war Filmpionier und produzierte im Rahmen seiner 1920er-Experimente abstrakte Filme mit Visualisierungen zu Musik („*Raumlichtkunst*“). Der Deutsche tat dies Dekaden vor den ersten Computergrafiken und Musikvideos. Der Film „*Motion Painting No. 1*“ ist eines seiner bekanntesten Werke. Es handelt sich dabei um eine Animation zu Musik von

---

<sup>7</sup> URL: [de.wikipedia.org/wiki/Synästhesie#Nonverbal\\_akustisch](https://de.wikipedia.org/wiki/Synästhesie#Nonverbal_akustisch) (Stand: 08.08.2014)

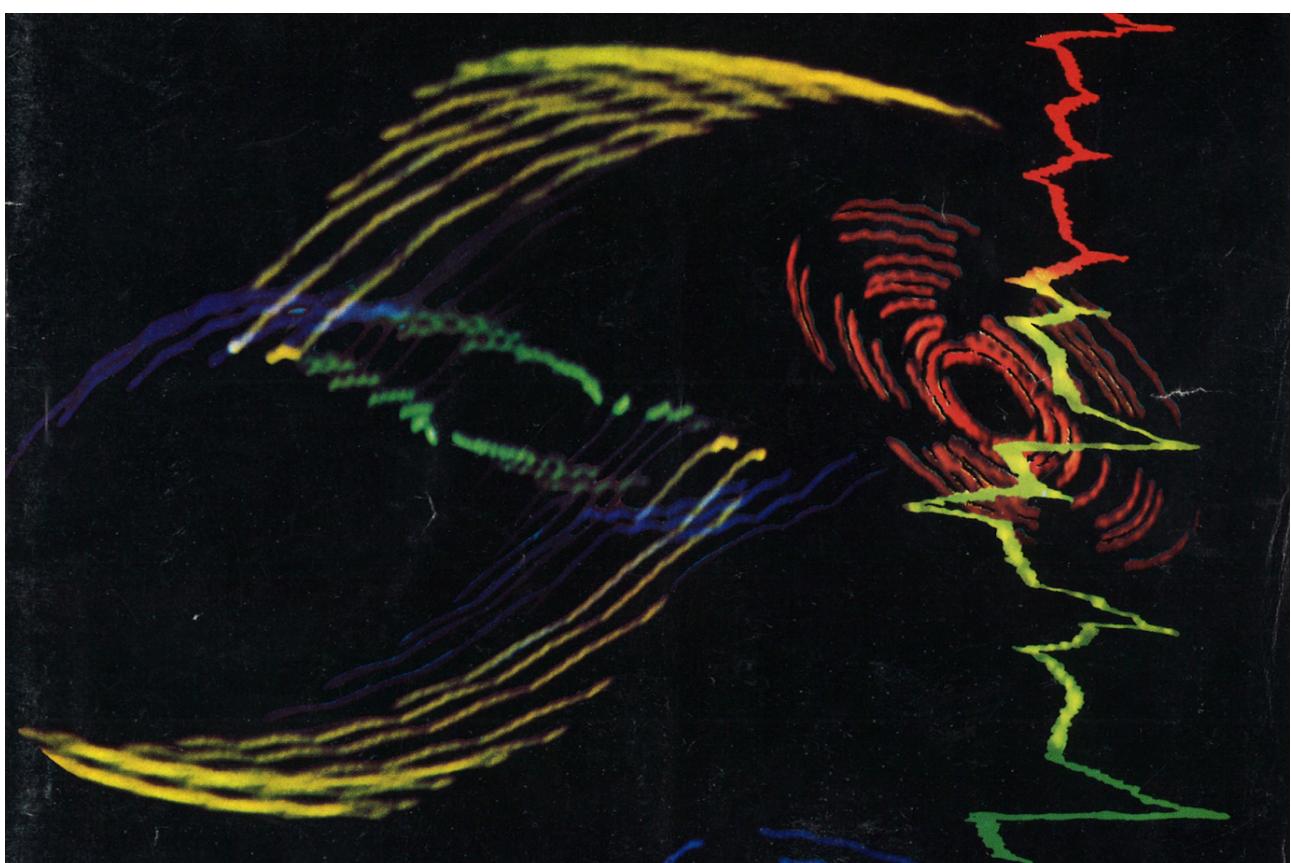
<sup>8</sup> URL: [www.hps.cam.ac.uk/library/universalharmony/newton.html](http://www.hps.cam.ac.uk/library/universalharmony/newton.html) (Stand: 07.08.2014)

<sup>9</sup> Matthew Neil Bain, „Real Time Music Visualization: A Study in the Visual Extension of Music“ (Ohio State University 2008)

<sup>10</sup> ebenda

Bach.<sup>11</sup> Zwanzig Jahre später veröffentlichte Disney mit „Fantasia“ einen Film-Klassiker, der als Vorläufer klassischer Musikvideos betrachtet werden kann. Der Film wird durchgehend von einem Orchester begleitet und kommt bis auf einen Kommentar ohne Sprache aus.

Mit dem Aufkommen der elektronischen Musik gewann die Musik-Visualisierung an Bedeutung, nicht zuletzt da technische Neuerungen das automatische Generieren dieser erleichterte. Der britische Amateurfunkpionier Fred Charles Judd war eine wichtige Figur in der Verbreitung der neuen Musikform. Nebst einigen der ersten Synthesizer baute er auch ein Musik-Visualisierungs-Gerät, das er „Chromasonics“ nannte. 1963 demonstrierte er Chromasonics, welches als einer der ersten Hardware-Visualizer betrachtet werden kann, an der Audio Fair in London. Das Gerät zeigte auf einem Fernsehbildschirm auf Basis von Musik von einem Tonträger farbige abstrakte Bilder an. Das Gerät wurde aber nie in Masse angefertigt; Es blieb nur bei diesem Prototyp.<sup>12</sup>



**Abbildung A:** Standbild von Judd's Hardware-Visualizer „Chromasonics“ (© Fred Charles Judd)

<sup>11</sup> URL: [www.centerforvisualmusic.org/Raumlichtkunst.html](http://www.centerforvisualmusic.org/Raumlichtkunst.html) (Stand: 08.08.2014)

<sup>12</sup> URL: [videocircuits.blogspot.ch/2014/05/fc-judd.html](http://videocircuits.blogspot.ch/2014/05/fc-judd.html) (Stand: 08.08.2014)

Zeitgleich mit der Entwicklung der Disco-Musik in den 1970er-Jahren wurden Lichtorgeln populär. Es handelt sich dabei um eine Art Scheinwerfer, die im Rhythmus der Musik verschiedene Farben annehmen können.<sup>13</sup> Heutzutage werden bei den Live-Auftritten von DJs und anderen Musikern aufwändige Lichtshows inszeniert, welche auch als eine Art Musik-Visualisierung betrachtet werden können.

Ebenfalls in den 1970er-Jahren wurde von Atari ein Gerät namens „*Atari Video Music System*“ verkauft, das mit der Stereoanlage und dem Fernseher verbunden wurde. Während man seine Musik hörte, zeigte es dann auf dem Bildschirm verschiedene Animationen dazu an. Es war wohl das einzige je in Serie produzierte Gerät seiner Art, und die Produktion wurde mangels Erfolg bereits nach einem Jahr wieder eingestellt.<sup>14</sup>

Mit dem Aufkommen des Personal Computers wurden die ersten Software-Visualizer umgesetzt. 1985, nur ein Jahr nachdem der erste Apple Macintosh veröffentlicht wurde, also zu einer Zeit in der die Heimcomputer noch an ihrem Anfang standen, wurde im „*Sinclair User Magazine*“ folgendes Produkt beworben:

„*SOUND TO LIGHT GENERATOR - a completely superb psychedelic light show. Simply play the music into the computer and watch the screen. Select from over 35 screens via the keyboard. Super fast, super sensitive, super value at £10.*“<sup>15</sup>

Dies ist nur ein Beispiel der vielen Software-Visualizer, die zu der Zeit schon existierten. PCs mit ausreichender Rechenleistung zum Ausführen der Musik-Analyse konnten Visualisierungen in Echtzeit generieren. Heutzutage ist nahezu jedes Musik-Wiedergabeprogramm wie *iTunes*, *Windows Media Player* oder *Winamp* mit einer Art Visualizer ausgestattet.<sup>16</sup>

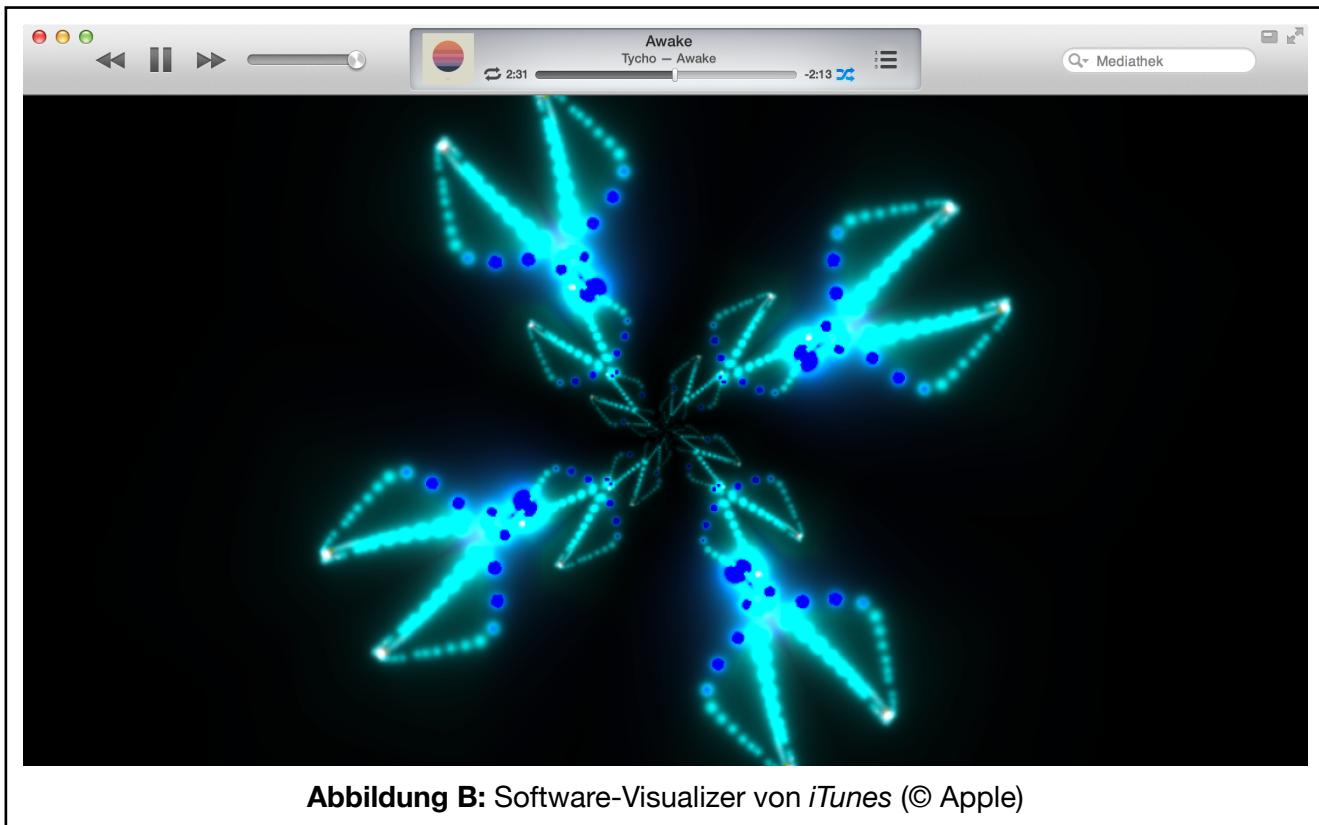
---

<sup>13</sup> URL: [de.wikipedia.org/wiki/Audiovisualisierung#Audiovisualisierung\\_als\\_Unterhaltungsmedium](http://de.wikipedia.org/wiki/Audiovisualisierung#Audiovisualisierung_als_Unterhaltungsmedium) (Stand: 12.08.2014)

<sup>14</sup> ebenda

<sup>15</sup> URL: [www.worldofspectrum.org/showmag.cgi?mag=SinclairUser/Issue064/Pages/SinclairUser06400097.jpg](http://www.worldofspectrum.org/showmag.cgi?mag=SinclairUser/Issue064/Pages/SinclairUser06400097.jpg) (Stand: 08.08.2014)

<sup>16</sup> URL: [de.wikipedia.org/wiki/Audiovisualisierung#Audiovisualisierung\\_als\\_Unterhaltungsmedium](http://de.wikipedia.org/wiki/Audiovisualisierung#Audiovisualisierung_als_Unterhaltungsmedium) (Stand: 12.08.2014)



**Abbildung B:** Software-Visualizer von *iTunes* (© Apple)

## 2.1 VJing

Um die 1980er-Jahre entwickelte sich mit der DJ-Szene auch die Szene der *Visual Jockeys* (VJs). Das *VJing* ist eine etwas andere Form der Musik-Visualisierung; VJs mixen Filmsequenzen, Bilder und Grafiken aus verschiedenen Quellen, um während des Auftritts eines Musikers live Darstellungen zu kreieren. VJs werden meist von Nachtclubs angeheuert, um den Auftritt der Musiker um ein visuelles Element zu erweitern.<sup>17</sup>

Michael Spahr ist ein lokaler Berner VJ, der um die Jahrtausendwende aktiv war. Auch heute noch produziert er Videos, er tritt aber nicht mehr live auf. Im Interview erfuhr ich von ihm einiges über seine Kunstform. Spahr, der unter dem Pseudonym „*VJ Rhaps*“ auftritt, erfuhr das erste Mal von der Szene, als er in Amsterdam lebte. Dort experimentierte er im freien Fernsehen mit Videokunst und eignete sich seine Fähigkeiten mittels Learning by Doing an. Zurück in der Schweiz begann er dann, auch live als VJ aufzutreten.

Die *Visuals* (Grafiken und Videos) der VJs seien viel mehr eine Erweiterung der Musik als nur deren Darstellung gewesen, sagte er; Sie schüfen einen zusätzlichen Raum, in dem die

<sup>17</sup> URL: [en.wikipedia.org/wiki/VJing](http://en.wikipedia.org/wiki/VJing) (Stand: 08.08.2014)

Zuschauer etwas neues erfahren und erleben können. Spahr sagte sogar, er habe sich bei seinen Visuals kaum von der Musik beeinflussen lassen; Die Stadt und die Natur seien seine Inspiration gewesen. Damit stehen die Animationen des VJs im Gegensatz zu jenen, die von Visualisierungsprogrammen generiert werden: Beim Visualisierungsprogramm werden rein die musikalischen Eigenschaften dargestellt - VJs hingegen erweitern durch ihre Arbeit die Musik um zusätzliche visuelle Elemente.

Das Generieren der Live-Visuals erforderte zu Spahr's Zeiten ein aufwändiges Setup. Die VJs seien die ersten am Set gewesen und die letzten, die gegangen seien, erzählte er. Nicht jeder Club sei mit einem Beamer ausgestattet gewesen, und für viele Dinge die heute mit einer Computer-Software gemacht werden können, hätten spezielle Geräte genutzt werden müssen. Eines dieser Geräte war das Video-Mischpult; Diese spezielle Hardware sah gleich aus wie ein DJ-Mischpult, erlaubte aber das Kombinieren und Überlagern von Videoaufnahmen anstatt Musik. Heute wird dieser *Compositing* genannte Vorgang auch mittels Computerprogrammen durchgeführt.

Spahr erzählte, dass VJs meist im Schatten der DJs stünden. „Am Schluss wollen die Leute tanzen und da ist der DJ im Vordergrund.“<sup>18</sup> Trotzdem scheint es die Menschen zu begeistern, wenn das, was sie hören, durch etwas visuelles begleitet wird - ansonsten hätte sich die VJ-Szene wohl kaum je gebildet.

---

<sup>18</sup> Interview mit Michael Spahr, 23.06.2014 (s. Anhang)

### 3 Inspiration

Bei meinen Recherchen zum Thema fand ich auch einige kontemporäre Musik-Visualisierungen, die mich beim Gestalten der Meinen inspiriert und motiviert haben. Im Folgenden drei auffällige Werke, die zeigen, wie unterschiedlich Musik-Visualisierungen aussehen können.

#### 3.1 Michal Levy: Giant Steps



Michal Levy hat Synästhesie;<sup>19</sup> Wenn sie Musik hört, sieht sie Farben und Formen. Mit dem Kurzfilm „Giant Steps“ wollte sie zeigen, was sie wahrnimmt, wenn sie ein Musikstück hört.<sup>20</sup> Diese Visualisierung ist im Vergleich zu anderen Animationen sehr bewegt und erzählt bereits eine einfache Geschichte; Im Video wird im Zusammenspiel mit der Musik aus vielen farbigen Rechtecken zuerst ein Haus, dann eine ganze Stadt aufgebaut. Die Animation beginnt sehr einfach, mit einem einzelnen Quadrat, doch entwickelt sich dann schnell weiter zu einem chaotischen Gebilde. Diese Entwicklung geschieht im Einklang mit der Musik, denn auch das Jazz-Stück beginnt simpel und wird

<sup>19</sup> Synästhesie: Neurologisches Phänomen bei dem die Stimulation eines Sinnes (hier: Gehör) andere Wahrnehmungen hervorrufen kann

<sup>20</sup> URL: [www.michalevy.com/giant-steps](http://www.michalevy.com/giant-steps) (Stand: 12.08.2014)

dann immer lauter und komplexer. Levy's Werk ist eines der inspirierendsten, das ich während meiner Recherchen gefunden habe.

### 3.2 Esteban Diácono: *Ljósið*



**Abbildung D:** Standbild aus Esteban Diácono's „*Ljósið*“ (© Esteban Diácono)

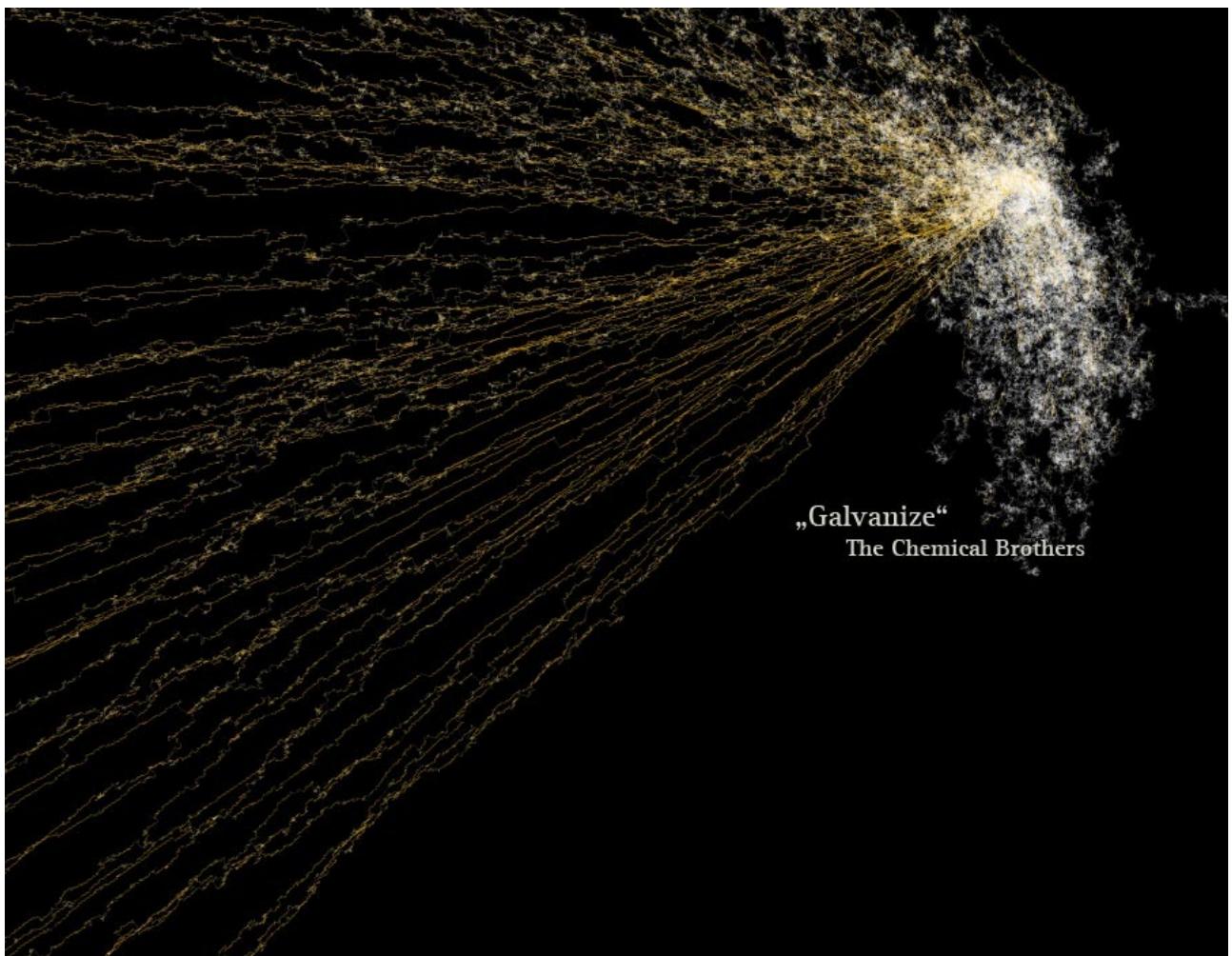
Diese Animation von Esteban Diácono zum klassischen Stück „*Ljósið*“ zeigt eine eher traditionelle Form der Musik-Visualisierung. Die Lautstärke der einzelnen Instrumente wird vom Rauch in verschiedenen Farben in einer Wellenform dargestellt. Ist ein Instrument lauter, bewegt sich der Rauch nach oben, wird es leiser, senkt sich der Rauch.<sup>21</sup> Dies ist ein einfaches Konzept, aber Diácono hat es perfekt umgesetzt; Die im Grunde simple Animation wirkt durch die gelungene Farbkomposition und die harmonischen Bewegungen des Rauchs fast schon hypnotisierend. Laut dem Künstler zeigen die blauen Farbtöne die Melancholie, und die wärmeren Farbtöne den hoffnungsvollen Teil des Liedes.<sup>22</sup> Das Video und das Musikstück passen hervorragend zusammen, und deshalb ist „*Ljósið*“ ein hervorragendes Beispiel für eine Musik-Visualisierung. Das ist wohl auch der Grund dafür, weshalb das Video online schon über eine Million mal angeschaut wurde.<sup>23</sup>

<sup>21</sup> URL: [portfolio.estebandiacono.tv/Olafur-Arnalds-Ljosid](http://portfolio.estebandiacono.tv/Olafur-Arnalds-Ljosid) (Stand: 12.08.2014)

<sup>22</sup> ebenda

<sup>23</sup> ebenda

### 3.3 Matthias Dittrich: Narratives 2.0



**Abbildung E:** Ein Bild aus der Serie „Narratives 2.0“ von Matthias Dittrich zum Stück „Galvanize“ von den Chemical Brothers (© Matthias Dittrich)

Musik-Visualisierungen müssen nicht zwingend Animationen sein, wie Matthias Dittrich mit seiner Serie „*Narratives 2.0*“ zeigt. Er hat eine Reihe von statischen Bildern veröffentlicht, die auf Basis von Musik entstanden sind. Die Zeitkomponente der Musikstücke wird dabei auf interessante Weise in das Bild eingebunden: Der Beginn des Musikstücks liegt im Zentrum des Bildes, die verschiedenen Frequenzen „explodieren“ dann über Zeit fächerartig aus diesem Zentrum.<sup>24</sup> Die Umsetzung des Konzepts ist zwar etwas minimalistisch; Doch die Idee, Musik, in welcher die Zeit ein essentieller Bestandteil ist, statisch darzustellen ist vielversprechend und kann zu interessanten Ergebnissen führen.

<sup>24</sup> URL: [www.matthiasdittrich.com](http://www.matthiasdittrich.com) (Stand: 12.08.2014)

## 4 Erarbeitungsprozess

Im Folgenden werden chronologisch die Arbeiten beschreiben, die zum Endprodukt geführt haben. Dies ist nur eine Übersicht des Vorgehens. Die genauen technischen Einzelheiten werden in späteren Abschnitten erklärt.

### 4.1 Übersicht über das Thema

Nach der Themenfindung begann ich mich im Internet über mein Thema zu informieren. Dabei stiess ich unter anderem auf die Werke von Michal Levy, Esteban Diácono und Matthias Dittrich. So erfuhr ich, was bereits gemacht wurde, und welche Möglichkeiten es gibt. (→ 3. *Inspiration*)

Für den geschichtlichen Teil dieser Arbeit, und um mehr über die Szene zu erfahren, interviewte ich Michael Spahr, einen lokalen *Visual Jockey* (VJ). Ich war erstaunt, als ich nach weiteren Recherchen im Internet erfuhr, wie lange das Konzept der Musik-Visualisierung schon besteht. (→ 2. *Geschichte der Musik-Visualisierung*)

Visualisierungen können entweder durch Computerprogramme oder mittels dedizierter Hardware generiert werden.<sup>25</sup> Ich musste mich also zu Beginn der Arbeit entscheiden, ob ich ein Visualisierungsprogramm für den Computer programmieren oder einen spezielles Gerät bauen wollte. Hardware-Visualisierungsgeräte haben heute so gut wie keine Vorteile gegenüber Computerprogrammen mehr. Das Bauen dieser ist zudem mit Kosten verbunden, während das Programmieren kostenlos ist. Ich entschied mich deshalb dafür, ein Programm umzusetzen.

### 4.2 Schaffen der technischen Grundlagen

Bald machte ich mich an die Umsetzung eines ersten Prototyps in der Programmiersprache Python. Ich entschied mich für diese Programmiersprache, da ich sie am besten kannte und so am schnellsten zu ersten Ergebnissen kommen konnte. Ich plante den groben Ablauf des Programmes, und skizzierte, wie es funktionieren sollte. (→ 5. *Funktionsweise des Visualisierungsprogramms*)

---

<sup>25</sup> URL: [www.instructables.com/id/Video-Beats-Music-Visualizer/](http://www.instructables.com/id/Video-Beats-Music-Visualizer/) (Stand: 12.04.2014)

Ich entschied, dass das Programm eine bewegte Animation zur Musik generieren sollte. Es wäre auch möglich gewesen, ein Standbild zu generieren, welches die durchschnittlichen Lautstärken und Tonhöhen während der Gesamtdauer des Stückes darstellt. Matthias Dittrich's „*Narratives 2.0*“ ist ein Beispiel dafür.<sup>26</sup> Es gibt sogar eine Firma, die für ihre Kunden statische Bilder mit der Wellenform ihrer Lieblingsklänge herstellt.<sup>27</sup> Meiner Meinung nach ist bei einer solchen Grafik aber die Verbindung zum ursprünglichen Musikstück zu wenig ersichtlich.

Für das Erstellen des ersten Prototyps musste ich einige wichtige Entscheidungen treffen, die dann auch in die finale Version meines Visualisierungsprogrammes gelangten. Beim ersten Schritt im Programmfluss, also dem Auslesen der Audio-Datei, musste ich mich für ein Dateiformat entscheiden. Ich wählte nach einigen Recherchen *RIFF WAVE*. Ich informierte mich über dieses Format und programmierte einen Mechanismus zum Auslesen der Dateien. (→ 5.1. *Auslesen der Audio-Datei*)

Die nächste Aufgabe war das Umsetzen einer Methode zur Analyse der Musik. Mein Ziel war es, aus der zuvor ausgelesenen Audio-Datei Informationen über die Tonhöhen und deren Lautstärken im Lied extrahieren zu können. Ich erfuhr, dass diese musikalischen Eigenschaften in Verbindung zu den physikalischen Eigenschaften Frequenz und Amplitude stehen.<sup>28</sup> Ich las, dass zum Erlangen dieser Informationen aus einer Schallwelle eine sogenannte *Fourier-Transformation* angewendet wird.<sup>29</sup> Diese Transformation ist ein wichtiger Bestandteil der Signalverarbeitung.<sup>30</sup> Ich musste diese also auch verwenden. Zum besseren Verständnis setzte ich selbst einen Algorithmus für eine diskrete Fourier-Transformation um. Schlussendlich nutzte ich dann aber den optimierten, schnelleren Algorithmus von *SciPy*, einer Programmbibliothek<sup>31</sup> verschiedener mathematischer

---

<sup>26</sup> URL: [www.matthiasdittrich.com](http://www.matthiasdittrich.com) (Stand: 12.08.2014)

<sup>27</sup> Born of Sound: [www.bornofsound.com](http://www.bornofsound.com)

<sup>28</sup> URL: [de.wikipedia.org/wiki/Frequenz#Frequenz\\_im\\_Alltag](http://de.wikipedia.org/wiki/Frequenz#Frequenz_im_Alltag) (Stand: 12.08.2014)

<sup>29</sup> URL: [www.stackoverflow.com/questions/153712/creating-music-visualizer](http://www.stackoverflow.com/questions/153712/creating-music-visualizer) (Stand: 12.08.2014)

<sup>30</sup> Signalverarbeitung: Bereich der Ingenieurwissenschaft, die sich mit der Analyse und Bearbeitung von analogen und digitalen Signalen beschäftigt

<sup>31</sup> Programmbibliothek: Eine Sammlung von Funktionen, die in ein Programm eingebunden und verwendet werden können

Werkzeuge für Python. Beobachtungen der Psychoakustik<sup>32</sup> erlaubten mir, von Frequenz und Amplitude ausgehend auf Tonhöhe und Lautstärke zu schliessen. (→ 5.2. Analyse der Schallwellen)

Für den Prototyp wurden etliche Zeilen Code geschrieben, getestet, wieder verworfen und überarbeitet. Aus dem Prototyp ging ein Grundgerüst für das Öffnen von Audio-Dateien, das Analysieren dieser und das Zeichnen von Grafiken hervor. Aufbauend auf dieses konnte ich dann die einzelnen Visualisierungen umsetzen.

### 4.3 Skizzieren und Umsetzen einer Visualisierung

Schon während der ersten Recherchen begann ich einige gestalterische Ideen für meine Visualisierungen zu skizzieren. Ich liess mich dabei von den Werken anderer inspirieren, versuchte aber auch möglichst viel eigene Ideen einzubringen. Die Einschränkungen die die verwendete Analysemethode mit sich brachte, machten das Gestalten einer einzigartigen Animation schwieriger. Mein Ziel war es trotzdem, mich nicht zu stark an den technischen Möglichkeiten zu orientieren, sondern meiner Kreativität freien Lauf zu lassen. (→ 5.4. Einschränkungen)

Während dem Skizzieren hörte ich ein Musikstück und versuchte mir vorzustellen, wie ich einzelne Komponenten davon grafisch darstellen könnte. Anschliessend versuchte ich, den Elementen auf meiner Skizze die Eigenschaften der Musik zuzuordnen, auf die sie reagieren sollten.

Das Umsetzen der Skizzen für das Visualisierungsprogramm erforderte dann noch einmal einen weiteren grossen Programmieraufwand. Ich experimentierte lange mit verschiedenen Parametern der Audio-Analyse, um die Visualisierungen so nahe wie möglich an meine Vorstellungen zu bringen. Die technischen Einschränkungen meiner Analysemethode erschwerten das aber erheblich.

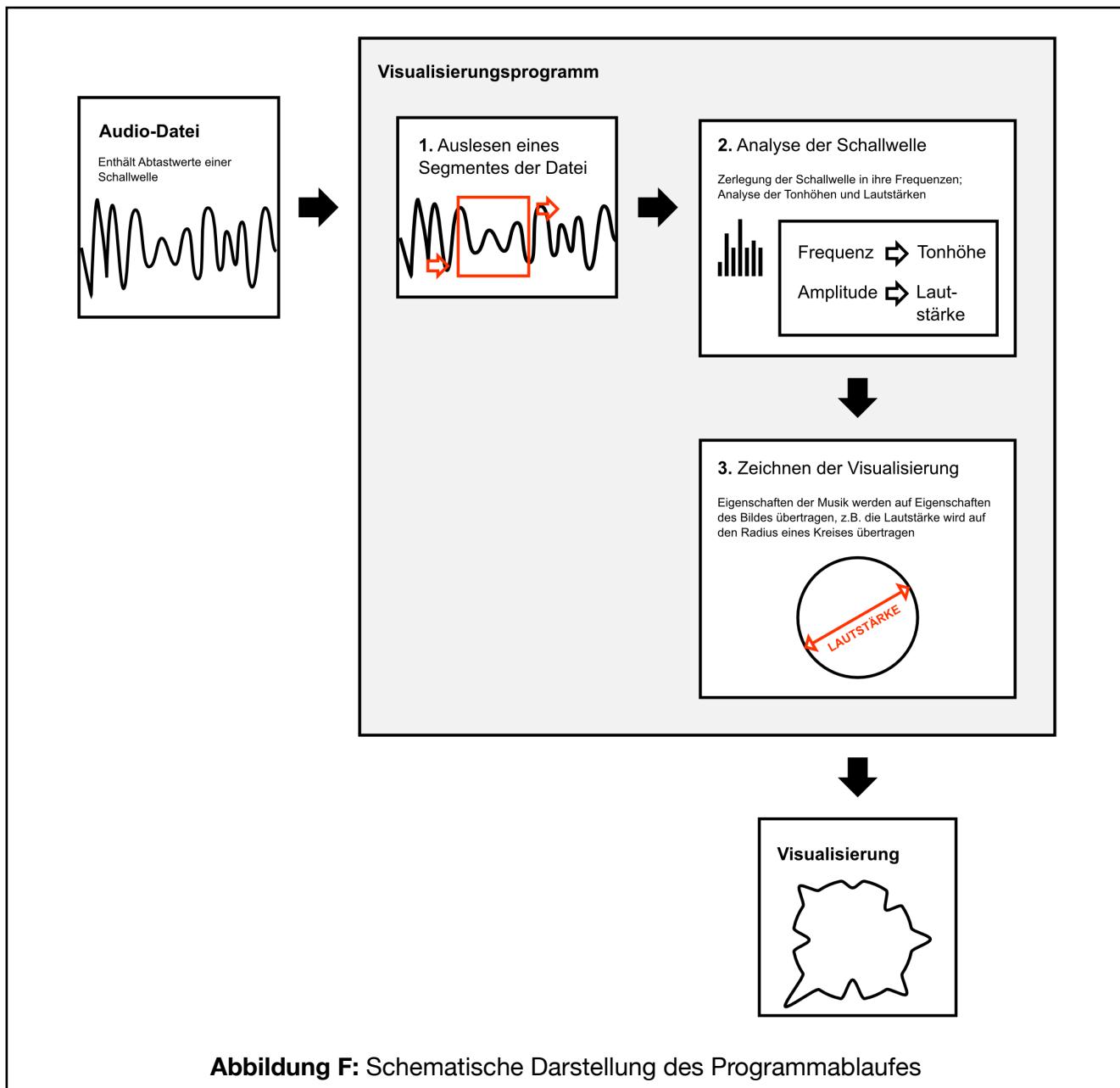
Schlussendlich vereinte ich das Grundgerüst und die Visualisierungen zu einem Programm, welches in der Lage ist, zu einer beliebigen Audio-Datei eine der zwei Animationen anzuzeigen, die ich programmiert habe.

---

<sup>32</sup> Psychoakustik: Forschungsgebiet, das sich mit den Zusammenhängen des menschlichen Empfindens und der Physik von Schall auseinandersetzt

## 5 Funktionsweise des Visualisierungsprogramms

Mein Programm erstellt Visualisierungen zu Musik. Dazu erhält es als Eingabe eine Audio-Datei, analysiert diese und generiert ein Video mit einer Animation zur Musik als Ausgabe. Dieses Video kann entweder in Echtzeit, während die Musik dazu spielt auf dem Bildschirm angezeigt werden, oder als Videodatei gespeichert werden.



Videos sind eine Aneinanderreihung mehrerer Standbilder. Üblicherweise enthalten sie 25 oder 30 Standbilder pro Sekunde.<sup>33</sup> Man spricht dann von einer Bildwiederholfrequenz

<sup>33</sup> TV-Standard in Europa: 25 Bilder pro Sekunde (*PAL*); [en.wikipedia.org/wiki/PAL](https://en.wikipedia.org/wiki/PAL)

(*Frame rate*) von 25 oder 30 fps<sup>34</sup>. Diese Bildwiederholfrequenzen garantieren ein flüssiges Video. Das menschliche Auge erkennt dann nicht mehr die einzelnen Standbilder, sondern Bewegungen. Eine zu tiefe Bildwiederholfrequenz wird als „stocken“ des Videos wahrgenommen.<sup>35</sup>

Das Visualisierungsprogramm muss also eine gewisse Anzahl Bilder pro Sekunde generieren, um eine flüssige Animation zu garantieren. Dazu durchläuft es für jedes Standbild folgende drei Schritte:

1. **Auslesen eines Segmentes der Audio-Datei:** Ein Teil der Schallwelle an der aktuellen zeitlichen Position im Musikstück wird ausgelesen, um sie analysieren zu können.
2. **Analyse der Schallwellen dieses Segments:** Es werden Eigenschaften der Musik wie Tonhöhe und Lautstärke aus der Schallwelle extrahiert.
3. **Zeichnen der Visualisierung:** Die Eigenschaften der Musik (bspw. Lautstärke) werden auf Eigenschaften des Bildes (bspw. Radius eines Kreises) übertragen.

Mit jeder Wiederholung dieser Schritte wird ein späteres Segment der Audio-Datei ausgelesen; Das Programm bewegt sich vom Beginn des Musikstückes zu dessen Ende. Die Standbilder ergeben dann aneinandergereiht und synchronisiert mit der Musik eine passende Visualisierung. Diese wird schliesslich auf dem Bildschirm angezeigt oder in einer Video-Datei gespeichert.

---

<sup>34</sup> fps: Frames per Second, Standbilder pro Sekunde

<sup>35</sup> URL: [de.wikipedia.org/wiki/Bewegte\\_Bilder](http://de.wikipedia.org/wiki/Bewegte_Bilder) (Stand: 08.08.2014)

## 5.1 Auslesen der Audio-Datei

### 5.1.1 Wahl des Dateiformates

Alles was wir hören sind Schallwellen.<sup>36</sup> Jedes Instrument und jeglicher Gesang erzeugen Schallwellen. Spielen mehrere Musiker ein Stück, so gelangt dies als eine einzelne kombinierte Schallwelle zu unserem Gehör.

Am Computer gibt es zwei grundsätzliche Möglichkeiten, Musik zu speichern. Entweder beschreibt das Dateiformat eine aufgenommene Schallwelle, oder das „Rezept“ zum Erstellen dieser Schallwelle, sprich die Noten jedes einzelnen Instruments.<sup>37</sup>

Die erste Möglichkeit ist weit verbreitet. Es gibt etliche Formate zur Speicherung von Schallwellen, unter anderem MP3<sup>38</sup>, AAC<sup>39</sup> oder RIFF WAVE<sup>40</sup>. Die meisten Heimanwender verwenden eines dieser Formate für ihre persönliche Musik-Sammlung auf dem MP3-Player oder dem PC. Die Formate sind weit verbreitet, da sie dafür genutzt werden können, Musik, die in einem Studio mit Mikrofonen aufgenommen wurde, zu speichern.

Für die zweite Möglichkeit gibt es nur ein allgemein akzeptiertes Format - das Format des *Musical Instrument Digital Interface*, kurz MIDI. MIDI-Dateien kann man sich wie ein Notenblatt vorstellen. Die Datei selbst enthält noch keine Schallwellen - diese muss der Computer durch spielen der Noten generieren.<sup>41</sup> Ein Beispiel für eine weit verbreitete Anwendung dieses Formats ist das Senden von Noten, die auf einem Keyboard „eingegeben“ werden, zu einem Synthesizer, der diese Noten dann spielt. Zur Veröffentlichung oder Speicherung von kompletten Musikstücken wird MIDI so gut wie nie genutzt, schon nur deshalb weil eine Speicherung von Gesang nicht möglich ist. (Die Noten des Gesangs könnten zwar gespeichert werden, aber nicht die Stimme selbst.) Es besteht aber die Möglichkeit, unter Voraussetzung eines guten musikalischen Gehörs, die

---

<sup>36</sup> URL: [www.podcomplex.com/guide/physics.html](http://www.podcomplex.com/guide/physics.html) (Stand: 12.08.2014)

<sup>37</sup> URL: [www.danmusic.com/mid/audio.html](http://www.danmusic.com/mid/audio.html) (Stand: 12.08.2014)

<sup>38</sup> MP3: MPEG Audio Layer III

<sup>39</sup> AAC: Apple Audio Codec

<sup>40</sup> RIFF WAVE: Waveform Audio File Format, oft nach der Dateiendung (.wav) nur „WAV“ genannt

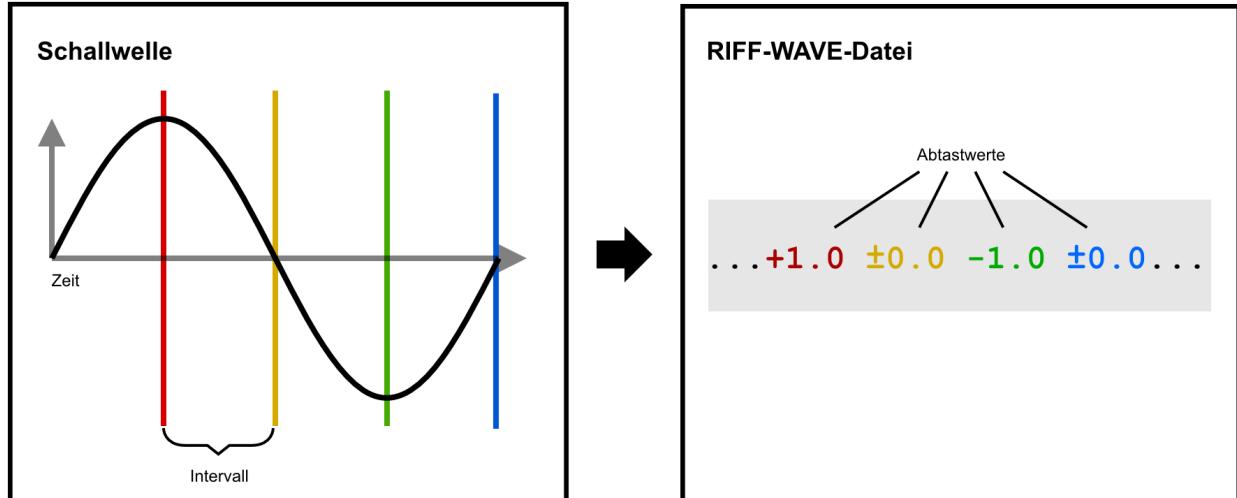
<sup>41</sup> ebenda

Noten eines Musikstückes herauszuhören, aufzuschreiben, und dann in einer MIDI-Datei zu speichern.

Die Schallwelle, die in MP3s und ähnlichen Dateien gespeichert ist, muss zuerst analysiert werden, um die musikalischen Eigenschaften wie Tonhöhe, Lautstärke und Klangfarbe zu extrahieren.<sup>42</sup> Bei MIDI-Dateien sind diese musikalischen Eigenschaften bereits in der Datei gespeichert; Eine Analyse ist nicht nötig. Da aber kaum Lieder im MIDI-Format zur Verfügung stehen, habe ich mich für das RIFF-WAVE-Format entschieden. RIFF-WAVE-Dateien beschreiben eine Schallwelle und sind einfacher auszulesen als vergleichbare Formate wie zum Beispiel MP3.

### 5.1.2 Das RIFF-WAVE-Dateiformat

RIFF-WAVE-Dateien enthalten die diskrete Beschreibung einer Schallwelle. Die Daten können unkomprimiert gespeichert werden; Das erleichtert das Auslesen, da keine Dekomprimierung vorgenommen werden muss.<sup>43</sup> Eine maximale Qualität ist garantiert, da keine Kompressionsartefakte auftreten. Ein Nachteil davon sind die vergleichsweise grossen Dateien.



**Abbildung G:** Aufnahme eines Tones in eine RIFF-WAVE-Datei; Die farbigen Balken sind Messungspunkte. Diese gemessenen Werte werden in der Datei gespeichert (farbig markiert).

<sup>42</sup> URL: [www.lightlink.com/tjweber/StripWav/WAVE.html](http://www.lightlink.com/tjweber/StripWav/WAVE.html) (Stand: 12.08.2014)

<sup>43</sup> ebenda

Bei der Erzeugung von RIFF-WAVE-Dateien wird der Ausschlag einer Schallwelle in fixen Intervallen gemessen. Nach jeder Messung wird der gemessene Wert (*Abtastwert*) am Ende der Datei angefügt. Dieser Prozess nennt man *Abtastung*. Die Datei enthält danach eine Liste von Punkten auf der Schallwelle, mit regelmässigen zeitlichen Abständen.<sup>44</sup> Der erste Wert in der Datei, also die erste Messung, entspricht der Position der Welle zu Zeit null. Der nächste Wert in der Datei entspricht der Position der Welle nach der Dauer eines Intervalls. Werte weiter hinten in der Datei entsprechen Werten zu einer späteren Zeit. Der Zusammenhang zwischen Zeit und Position in der Datei kann mit folgender Formel beschrieben werden:

$$t = \frac{1}{F_s} * p$$

**Formel 1:** Berechnung der Zeit-Position (t: Zeit,  $F_s$ : Abtastrate, p: Position in der Datei in Bytes)

Die *Abtastrate* beschreibt, wie viele Messungen pro Sekunde gemacht werden. Je grösser die Abtastrate, desto kleiner sind die zeitlichen Abstände zwischen den Messungen. Bei einer höheren Abtastrate nimmt die Qualität zu, da die Welle mit mehr Punkten genauer beschrieben werden kann, aber auch die Dateigrösse, da für dieselbe Zeitspanne mehr Messungen gespeichert werden müssen.

Es ist also notwendig, die Abtastrate zu kennen, um die Schallwelle korrekt auslesen und über den richtigen Zeitraum darstellen zu können. Deshalb wird diese Information noch vor den eigentlichen Messungen der Punkte auf der Welle am Anfang der Datei im sogenannten *Header* angegeben. Der Header enthält auch weitere wichtige Informationen, die zum Auslesen der Datei notwendig sind.

Die folgende Tabelle enthält das Format einer typischen RIFF-WAVE-Datei. Die *Position* beschreibt, an welcher Stelle in der Datei die Information gespeichert ist; die *Länge*, wie viel Platz zur Speicherung dieser Information benötigt wird.

Ein *Bit* ist die kleinste Speichereinheit des Computers und kann zwei Werte speichern. Ein *Byte* besteht aus 8 Bits und kann Werte von 0 - 255 speichern. Bei der Speicherung in RIFF-WAVE-Dateien kann die Anzahl Bits zur Speicherung der Abtastwerte ausgewählt

---

<sup>44</sup> ebenda

werden. Eine höhere Anzahl Bits ermöglicht eine feinere Unterscheidung zwischen ähnlichen Werten, da ein grösserer Wertebereich gespeichert werden kann.

Position (Bytes)	Länge (Bytes)	Beschreibung
0	4	<b>Header-Beginn</b> Die Buchstaben <b>fmt</b> (mit abschliessendem Leerzeichen, deshalb die Länge von 4 Bytes) markieren den Beginn des Headers
4	4	Länge des Headers in Bytes
8	2	Kompression (im Visualisierungsprogramm wird keine verwendet)
10	2	Anzahl der Kanäle (1 = Mono, 2 = Stereo, etc.)
12	4	Abtastrate
16	4	Bytes pro Sekunde: Wie viele Bytes zur Speicherung aller Abtastwerte einer Sekunde der Schallwelle benötigt werden
20	2	Anzahl Bytes für die Speicherung eines Abtastwerte für jeden Kanal zusammengezählt
22	2	Anzahl Bits für die Speicherung eines einzelnen Abtastwertes
24	4	<b>Daten-Beginn</b> Die Buchstaben <b>data</b> markieren den Beginn der Abtastwerte
28	4	Länge des Datenblocks
32	siehe Byte Nr. 28	Abtastwerte; Hier werden die eigentlichen Abtastwerte aneinandergereiht gespeichert

**Tabelle 1:** Format einer typischen RIFF-WAVE-Datei<sup>45</sup>

Mein Visualisierungsprogramm implementiert einen Mechanismus, um RIFF-WAVE-Dateien auszulesen. Es unterstützt RIFF-WAVE-Dateien, die die Abtastwerte mit 8 bits ohne Vorzeichen oder 16 bits mit Vorzeichen speichern (siehe Byte Nr. 22 im Header).

---

<sup>45</sup> URL: [de.wikipedia.org/wiki/RIFF\\_WAVE#Beispiel\\_eines\\_allgemein\\_lesbaren\\_WAVE-PCM-Formats](https://de.wikipedia.org/wiki/RIFF_WAVE#Beispiel_eines_allgemein_lesbaren_WAVE-PCM-Formats) (Stand: 10.08.2014)

## 5.2 Analyse der Schallwelle

Nach dem Auslesen der Audio-Datei befinden sich Informationen über die Positionen der Welle zu verschiedenen Zeitpunkten im Speicher des Computers. Die Schallwelle liegt in der *Zeitdomäne* vor. Da die Klänge aller Instrumente in einer einzelnen Schallwelle kombiniert vorliegen, kann diese sehr komplexe Formen annehmen. Die für die Musik relevanten physikalischen Eigenschaften müssen nun aus dieser Welle extrahiert werden. Dabei kommt die sogenannte *Fourier-Transformation* zur Anwendung.<sup>46</sup>

### 5.2.1 Fourier-Transformation

Nach Joseph Fourier, einem französischen Mathematiker und Physiker, kann jede beliebige Welle als eine Summe von Sinus- und Cosinuswellen ausgedrückt werden. Selbst die unförmigsten, kompliziertesten Wellen können mittels Addition von Sinus- und Cosinuswellen aufgebaut werden.<sup>47</sup>

Sinus- und Cosinuswellen werden zusammenfassend auch als *Sinusoide* bezeichnet. Ein Sinusoid wird durch drei Eigenschaften definiert:<sup>48</sup>

1. *Frequenz*: Anzahl Wellenberge und -täler in einem gewissen Zeitabschnitt; Die Einheit *Hertz* (Hz) wird für Frequenzen verwendet. Ein Hertz entspricht einem Wellenberg und -tal pro Sekunde.
2. *Amplitude*: Maximale Auslenkung der Welle; „Höhe des Wellenbergs“
3. *Phase*: Verschiebung der Welle auf der X-Achse

$$y = a * \sin(f * x + \phi)$$

**Formel 2:** Eine Sinuswelle (a: Amplitude, f: Frequenz,  $\phi$ : Phase)

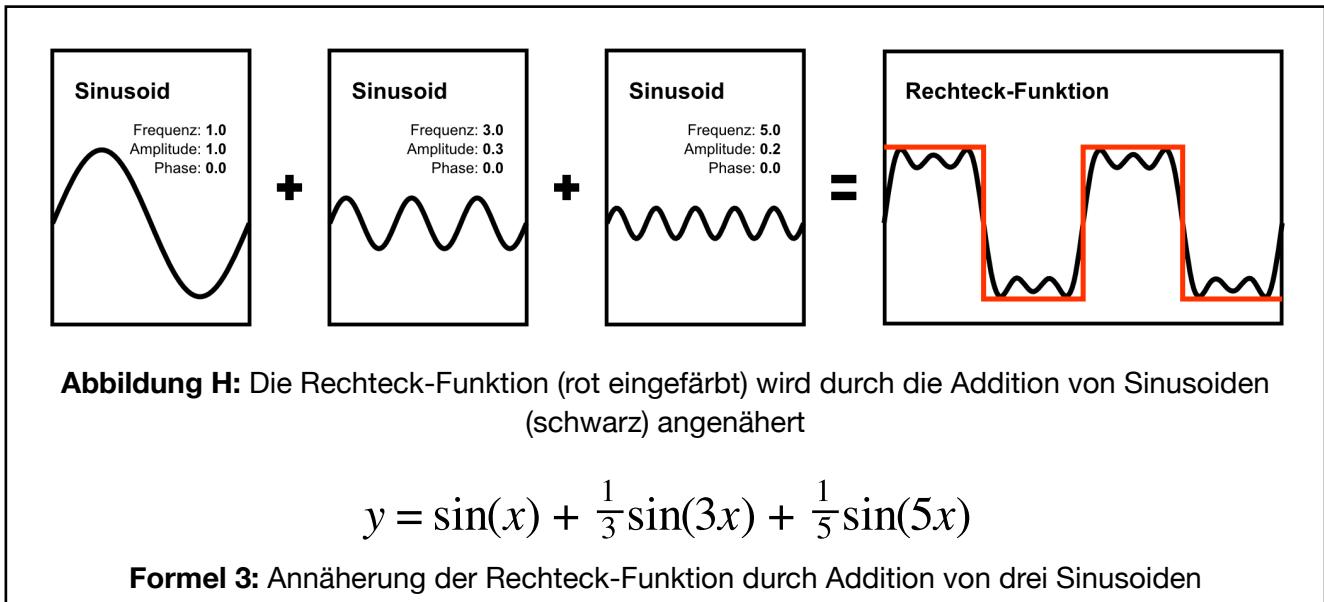
Durch die Addition mehrerer Sinusoide mit den richtigen Eigenschaften kann man jede beliebige Wellenform erhalten. Auch die im vorherigen Schritt ausgelesene Schallwelle kann als Summe von Sinusoiden ausgedrückt werden.

<sup>46</sup> URL: [en.wikipedia.org/wiki/Fourier\\_analysis#Applications\\_in\\_signal\\_processing](https://en.wikipedia.org/wiki/Fourier_analysis#Applications_in_signal_processing) (Stand: 12.08.2014)

<sup>47</sup> URL: [www.betterexplained.com/articles/an-interactive-guide-to-the-fourier-transform/](http://www.betterexplained.com/articles/an-interactive-guide-to-the-fourier-transform/) (Stand: 12.08.2014)

<sup>48</sup> URL: [de.wikipedia.org/wiki/Sinusoid](https://de.wikipedia.org/wiki/Sinusoid) (Stand: 12.08.2014)

Im Folgenden Beispiel wird eine Rechteck-Funktion<sup>49</sup> durch Addition von drei Sinusoiden angenähert. Um eine perfekte Rechteck-Funktion zu erhalten müsste aber eine unendliche Anzahl von Sinusoiden addiert werden.

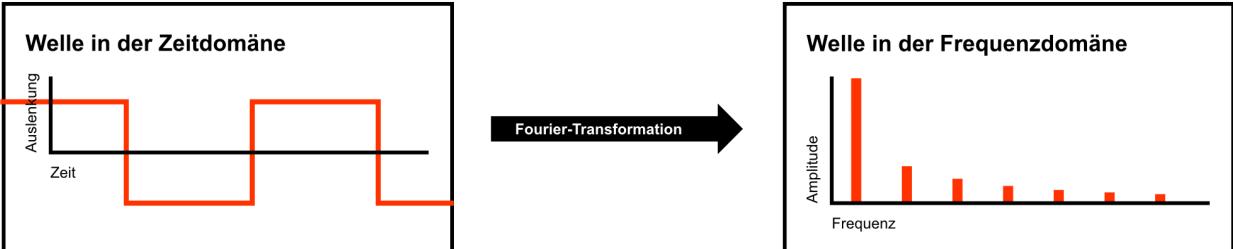


Das Berechnen der Eigenschaften der Sinusoide (Amplitude, Frequenz und Phase), aus denen die ursprüngliche Welle zusammengesetzt werden kann, wird als Fourier-Transformation bezeichnet. Die *Fourier-Transformierte* einer Welle kann man veranschaulichend als Rezept für die ursprüngliche Welle ansehen; Die Zutaten des Rezeptes sind Sinusoide verschiedener Frequenz, Amplitude und Phase. Durch Zusammenfügen aller Zutaten, also durch das Addieren aller Sinusoide, erhält man wieder die ursprüngliche Welle.<sup>50</sup>

Die Fourier-Transformation transformiert die ursprüngliche Welle von der *Zeitdomäne* in die *Frequenzdomäne*. In der Frequenzdomäne kann die Schallwelle deutlich einfacher auf Eigenschaften der Musik analysiert werden, da die Frequenzen und Amplituden der Sinusoide in Zusammenhang mit den Tonhöhen und deren Lautstärken im Lied stehen. Die Phasen sind für die musikalische Analyse irrelevant.

<sup>49</sup> Rechteck-Funktion: Schwingung die in einer bestimmten Frequenz sprunghaft zwischen einem Minimal- und Maximalwert hin- und her wechselt

<sup>50</sup> URL: [www.betterexplained.com/articles/an-interactive-guide-to-the-fourier-transform/](http://www.betterexplained.com/articles/an-interactive-guide-to-the-fourier-transform/) (Stand: 12.08.2014)



**Abbildung I:** Die Fourier-Transformierte (rechts) gibt Auskunft über die Frequenzen und Amplituden der Sinusoide, aus denen die ursprüngliche Welle (links) besteht.

Zur Berechnung der Fourier-Transformation mit diskreten Zahlenwerten, wie diese bei der ausgelesenen Schallwelle vorliegen, wird die *Korrelation*, ein Konzept aus der Signalverarbeitung angewandt.<sup>51</sup> Der *Korrelationswert* beschreibt, wie ähnlich sich zwei Wellen sind. Bei einem hohen Korrelationswert sind sich die zwei Wellen sehr ähnlich. Ist der Korrelationswert nahe null, sind die Wellen sehr unterschiedlich.<sup>52</sup>

Zur Berechnung der Korrelation werden die beiden Wellen Wert für Wert miteinander multipliziert. Die Produkte werden addiert.<sup>53</sup>

$$\sum_{i=0}^N x(i) \cdot y(i)$$

**Formel 4:** Berechnung der Korrelation zweier Funktionen für die Punkte von 0 bis N

Um die Fourier-Transformation zu berechnen wird die Korrelation von Sinusoiden (in diesem Fall Cosinuswellen) mit Frequenzen von null bis zur höchstmöglichen Frequenz und der ursprünglichen Welle berechnet. In anderen Worten: Ein Sinusoid jeder Frequenz wird mit der ursprünglichen Welle „verglichen“. Der Korrelationswert zwischen einem Sinusoid und der ursprünglichen Welle beschreibt dessen Anteil an der ursprünglichen Welle, also dessen Amplitude. Mittels Korrelation werden also die Amplituden aller Komponenten-Sinusoide berechnet.<sup>54</sup>

<sup>51</sup> URL: [www.practicalcryptography.com/miscellaneous/machine-learning/intuitive-guide-discrete-fourier-transform/](http://www.practicalcryptography.com/miscellaneous/machine-learning/intuitive-guide-discrete-fourier-transform/) (Stand: 12.08.2014)

<sup>52</sup> ebenda

<sup>53</sup> ebenda

<sup>54</sup> ebenda

Die Fourier-transformierte enthält auch Informationen über die Phase der Komponenten-Sinuswellen. Das Berechnen der Phasen funktioniert ähnlich; Es wird ebenfalls die Korrelation zwischen der ursprünglichen Welle und verschiedenen Sinusoiden berechnet. Die Sinusoide sind bei der Berechnung der Phasen aber Sinuswellen, nicht Cosinuswellen wie bei der Amplitudenberechnung.<sup>55</sup> Die Phasenberechnung ist für die Analyse der musikalischen Eigenschaften nicht relevant.

$$X(k) = \sum_{n=0}^{N-1} x(n) * \cos\left(\frac{2\pi kn}{N}\right) - i \sum_{n=0}^{N-1} x(n) * \sin\left(\frac{2\pi kn}{N}\right)$$

**Formel 5:** Diskrete Fourier Transformation (Zur Veranschaulichung umgeformt)  
k: Frequenz; x(n): zu transformierende Schallwelle; N: Anzahl Messungen der Schallwelle

Formel 5 ist eine zur Veranschaulichung umgeformte diskrete Fourier-Transformation. Es werden die Amplitude und Phase des Komponenten-Sinusoids mit der Frequenz  $k$  berechnet. Das Resultat dieser Berechnung ist eine komplexe Zahl; Der reale Teil enthält die Amplitude, der imaginäre Teil die Phase. Wie unschwer zu erkennen ist enthält diese Formel zwei Korrelationsberechnungen (vgl. Formel 4).<sup>56</sup>

Um die gesamte Schallwelle zu transformieren wird die diskrete Fourier-Transformation mit obiger Formel für jede Frequenz ( $k$ ) berechnet.

Ich habe diesen Algorithmus zur Berechnung der Fourier-Transformation zum besseren Verständnis in der Programmiersprache Python selbst umgesetzt. Das Visualisierungsprogramm nutzt aber die Fourier-Transformation von *SciPy*, einer Programmbibliothek wissenschaftlicher Funktionen für die Programmiersprache Python.

### 5.2.2 Fensterfunktion

Da im Visualisierungsprogramm die Audio-Datei nur segmentweise ausgelesen wird, kann es zu Artefakten kommen. Das passiert, wenn die Länge des Segments, das Ausgelesen wird, nicht einem ganzzahligen Vielfachen der Periode der Schallwelle („Länge der Welle“) entspricht. In diesem Fall wird die Welle „angeschnitten“. Das führt

---

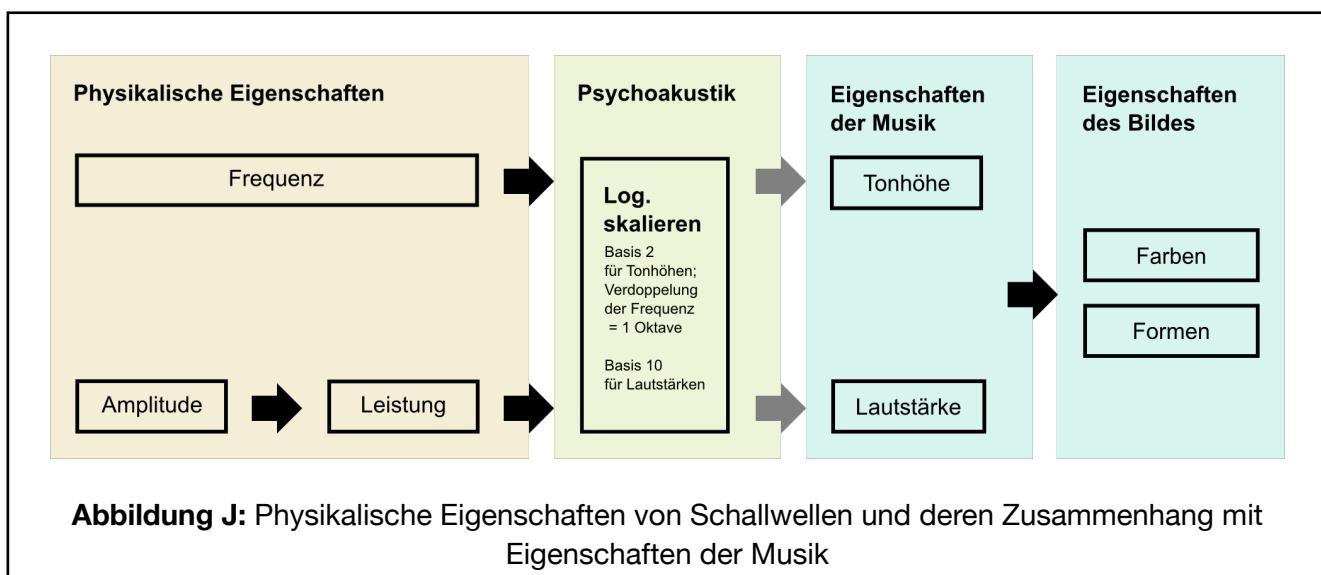
<sup>55</sup> ebenda

<sup>56</sup> ebenda

zu einem sogenannten *Leck-Effekt*, der die Ergebnisse der Fourier-Transformation verfälscht.<sup>57</sup>

Um dem entgegenzuwirken kann eine Fensterfunktion angewandt werden. Fensterfunktionen gewichten die Abtastwerte am Anfang und Ende des Segments geringer als diejenigen in der Mitte des Segments. Damit wird der Leck-Effekt verminder. In der Signalverarbeitung finden unter anderem die Fensterfunktionen von Hamming, Von Hann, Blackman und Bartlett Anwendung. Mein Visualisierungsprogramm nutzt eine Hamming-Fensterfunktion.<sup>58</sup>

### 5.2.3 Eigenschaften von Musik



Die Frequenzen und Amplituden der einzelnen Sinusoide, aus denen die Schallwelle besteht, sind für die Analyse der musikalischen Eigenschaften wichtig; Deren Frequenzen stehen in Verbindung mit der Tonhöhe und die Amplituden stehen in einem Zusammenhang mit der Lautstärke.<sup>59</sup> Die Fourier-Transformierte der Schallwelle kann also verwendet werden, um auf die Lautstärke verschiedener Tonhöhen zu schliessen.

Die Psychoakustik erforscht und beschreibt die Zusammenhänge zwischen den physikalischen Eigenschaften von Schallwellen und dem menschlichem Empfinden von Schall. Aber das menschliche Gehör ist individuell; Eine komplett korrekte Übertragung

<sup>57</sup> URL: [de.wikipedia.org/wiki/Leck-Effekt](http://de.wikipedia.org/wiki/Leck-Effekt) (Stand: 12.08.2014)

<sup>58</sup> URL: [de.wikipedia.org/wiki/Fensterfunktion](http://de.wikipedia.org/wiki/Fensterfunktion) (Stand: 12.08.2014)

<sup>59</sup> URL: [www.podcomplex.com/guide/physics.html](http://www.podcomplex.com/guide/physics.html) (Stand: 12.08.2014)

der physikalischen Eigenschaften von Schallwellen auf die empfundenen Eigenschaften der Musik ist nicht möglich, eine Annäherung aber schon.<sup>60</sup>

Eine psychoakustische Beobachtung ist, dass eine Verdoppelung der Frequenz einer Steigerung der Tonhöhe um eine Oktave entspricht.<sup>61</sup> Um also dem menschlichen Empfinden von Tonhöhen möglichst nahe zu kommen, skaliert das Visualisierungsprogramm die Frequenzen logarithmisch mit einer Basis von 2. Ohne diese Skalierung wären Tonhöhenunterschiede bei tiefen Tönen zu wenig sichtbar und bei hohen Tönen übervertreten.

Die Übertragung der Amplitude auf die Lautstärke gestaltet sich als schwieriger. Zunächst muss basierend auf der Amplitude der Welle deren physikalische Leistung berechnet werden. Unser Lautstärkeempfinden hängt aber nicht nur von der Leistung ab. Die Höhe des Tones und wie lange dieser andauert beeinflusst auch, als wie laut wir diesen empfinden.<sup>62</sup> Eine hundertprozentig korrekte Übertragung der Leistung auf die empfundene Lautstärke ist nicht möglich. Um bei meiner Visualisierung zu schöneren Ergebnissen zu kommen habe ich zwei Massnahmen getroffen, um das menschliche Hören nachzuempfinden. Zuerst wird die Lautstärke logarithmisch mit Basis 10 skaliert, da dies in Experimenten zu den schönsten Ergebnissen geführt hat. Dann wird sogenanntes *A-weighting* angewandt. Beim A-weighting wird die Lautstärke je nach Tonhöhe angepasst. Da tiefe und sehr hohe Töne vom menschlichen Gehör als leiser empfunden werden, als sie wirklich sind, werden diese abgeschwächt. Lautstärken von Tönen im mittleren Frequenzbereich werden leicht verstärkt.<sup>63</sup>

Eine weitere musikalische Eigenschaft ist die *Klangfarbe*. Wird mit einem Instrument ein Ton angeschlagen, so erklingen gleichzeitig auch einige weitere Töne, die *Obertöne*. Die Anzahl Obertöne, und die Frequenzen und Amplituden dieser beschreibt die Klangfarbe. Die Klangfarbe ist charakteristisch für jedes Instrument. Sie ist das, was uns erlaubt, verschiedene Instrumente, die denselben Ton spielen, zu unterscheiden. Eine Erkennung

---

<sup>60</sup> URL: [de.wikipedia.org/wiki/Musikwahrnehmung#Wahrnehmung\\_von\\_Schallsignalen](https://de.wikipedia.org/wiki/Musikwahrnehmung#Wahrnehmung_von_Schallsignalen) (Stand: 12.08.2014)

<sup>61</sup> URL: [de.wikipedia.org/wiki/Oktave](https://de.wikipedia.org/wiki/Oktave) (Stand: 12.08.2014)

<sup>62</sup> URL: [de.wikipedia.org/wiki/Lautstärke](https://de.wikipedia.org/wiki/Lautstärke) (Stand: 12.08.2014)

<sup>63</sup> URL: [en.wikipedia.org/wiki/A-weighting](https://en.wikipedia.org/wiki/A-weighting) (Stand: 12.08.2014)

der Klangfarbe wird aber für die Musik-Visualisierung meist nicht benötigt und ist unüblich. Es wäre eine grosse Datenbank der Klangfarben aller Instrumente und ein Algorithmus zur Klangfarben-Erkennung notwendig.

## 5.3 Zeichnen der Visualisierung

Nach dem Erlangen der Informationen über die Lautstärken aller Tonhöhen im aktuellen Segment des Musikstücks können diese auf Eigenschaften des Bildes wie Farben und Formen übertragen werden. Zum Zeichnen der Visualisierung nutzt mein Programm eine *Grafikbibliothek* namens *Cairo* und deren Python-Anbindung *PyCairo*.

### 5.3.1 Wahl einer Grafikbibliothek

Grafikbibliotheken vereinfachen das Zeichnen von Grafiken für den Programmierer. Sie stellen beispielsweise Funktionen zum Zeichnen von Kreisen, Linien, Rechtecken und Polygonen auf den Pixeln des Computerbildschirms (*Rasterung*) zur Verfügung.<sup>64</sup> Nebst Cairo gibt es eine Fülle anderer Grafikbibliotheken mit verschiedenen Vor- und Nachteilen die sie für bestimmte Einsatzzwecke geeigneter machen als andere. Die Wahl der richtigen Grafikbibliothek ist wichtig um Zeit zu sparen. Ich konnte die Auswahl meiner Grafikbibliothek recht schnell auf zwei mögliche Kandidaten einschränken, die für meine Zwecke geeignet schienen. Zum Einen Cairo, eine eher neue Grafikbibliothek, und zum anderen die alte und bewährte *OpenGL* (Open Graphics Library, Offene Grafikbibliothek). Die meisten anderen Grafikbibliotheken waren für die Zwecke dieses Projekts entweder zu unausgereift, boten nicht die nötigen Funktionen, oder es war nicht möglich, diese mit der Programmiersprache Python zu verwenden. Zwei Argumente sprachen für die Verwendung von Cairo anstatt OpenGL:

1. Cairo ist vergleichsweise einfach zu verwenden. Das Erlernen der Gebrauchsweise erfordert wenig Zeit. OpenGL hat nach populärer Meinung eine der steilsten Lernkurven aller Grafikbibliotheken.
2. Cairo bietet mehrere sogenannte *Backends*; So kann das Gezeichnete nicht nur auf dem Bildschirm angezeigt werden, sondern auch als Bilddatei (Raster- oder Vektorgrafik)

---

<sup>64</sup> URL: [de.wikipedia.org/wiki/Grafikbibliothek](http://de.wikipedia.org/wiki/Grafikbibliothek) (Stand: 12.08.2014)

exportiert werden. Das Exportieren von Rastergrafiken ist bei OpenGL zwar auch möglich, aber erschien mir umständlicher. Interessanterweise existiert für Cairo sogar ein OpenGL-Backend. Wird dieses Backend verwendet, werden die Zeichen-Routinen von Cairo in „OpenGL-Sprache“ übersetzt und OpenGL übernimmt dann die Darstellung des Gezeichneten. Es handelt sich dabei also um eine Abstraktionsebene, die über OpenGL gelegt wird, und deren Verwendung vereinfacht (aber auch einschränkt, da Cairo nicht alle Funktionen von OpenGL aufweist). Dieses Backend ist aber noch unausgereift und funktionierte auf meinem PC nicht.

Durch die Entscheidung, Cairo zu verwenden beschränkte ich mich auf 2D-Grafiken, da Cairo 3D-Grafiken im Gegensatz zu OpenGL nicht unterstützt. OpenGL hätte ausserdem wohl eine bessere Zeichen-Performance ermöglicht, also mehr Bilder pro Sekunde (fps) generieren können. Die Performance von Cairo war aber für meine Visualisierung ausreichend, um in Echtzeit ein flüssiges Video zu generieren.

### 5.3.2 Zeichnen mit Cairo

Um die Visualisierung mit Cairo zu zeichnen, wird zuerst eine „Leinwand“ mit der gewünschten Grösse und Hintergrundfarbe eingerichtet. Jeder weitere Zeichenvorgang besteht dann aus folgenden drei Schritten:<sup>65</sup>

1. **Definieren einer Form:** Zuerst wird die Form, die gezeichnet werden soll, definiert. Cairo bietet dazu verschiedene Funktionen, unter anderem solche zum Zeichen von Rechtecken, Kreisbögen (oder ganzen Kreisen), und Polygonen, deren Eckpunkte frei definiert werden können.
2. **Auswahl einer Quelle:** Die Quelle kann einfarbig, ein Farbverlauf oder auch eine Bilddatei sein.
3. **Füllen oder Nachziehen der Form:** Die zuvor definierte Form wird mit der Quelle ausgefüllt oder ein Rahmen um die Form wird mit der Quelle eingefärbt. Zur Veranschaulichung kann man sich vorstellen, das ein Stempel mit der Form in eine Farbe (die Quelle) getunkt wird und dann auf die Leinwand gestempelt wird.

---

<sup>65</sup> URL: [www.cairographics.org/tutorial/](http://www.cairographics.org/tutorial/) (Stand: 12.08.2014)

Ist eine Form erstmals definiert, besteht die Möglichkeit, diese zu transformieren, also zu vergrössern, verkleinern, rotieren oder anderweitig zu verzerrn. Formen und Quellen können mehrmals verwendet und beliebig kombiniert werden.

Diese Methode, Grafiken zu zeichnen, ist relativ unkompliziert, aber trotzdem sehr flexibel und bietet sehr viele Möglichkeiten.

## 5.4 Einschränkungen

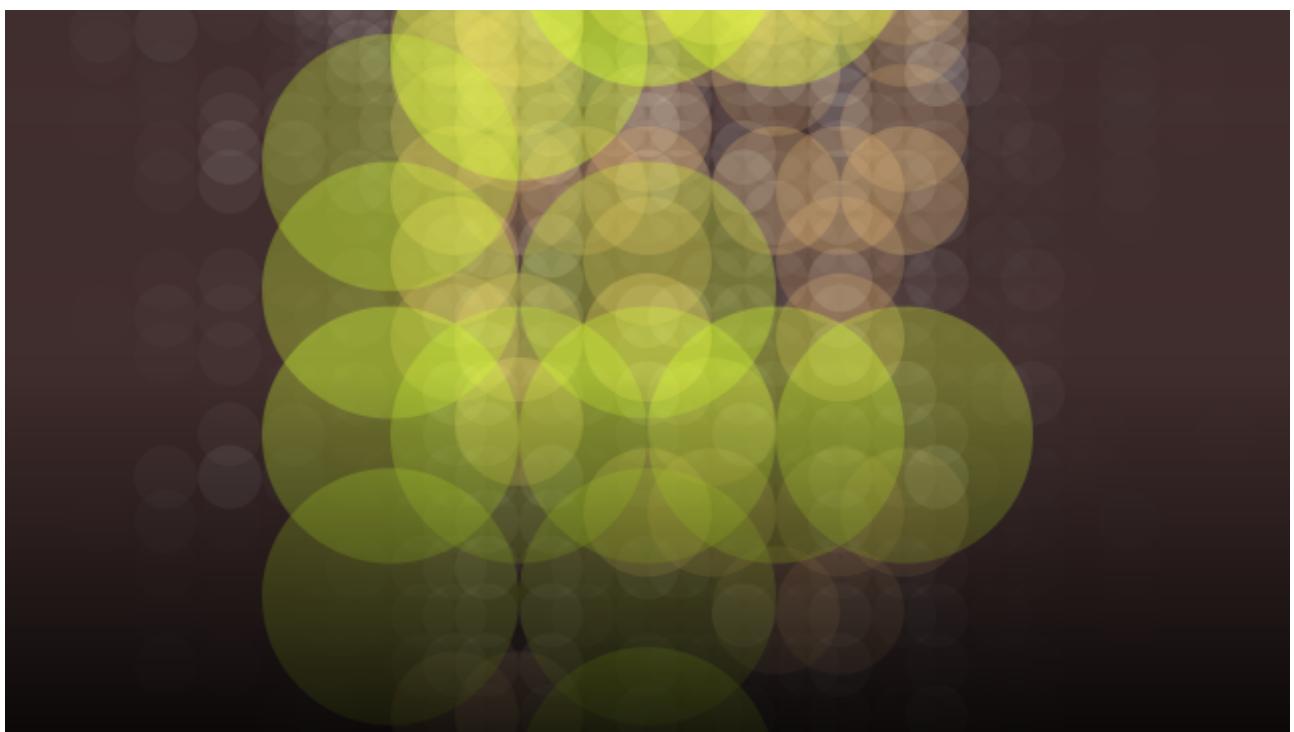
Die Architektur des Programmes bringt einige Einschränkungen mit sich.

Nur durch Analyse der Frequenz ist eine Erkennung der einzelnen Instrumente und der genauen Noten, die diese spielen, sehr umständlich. Das liegt daran, dass bei der Aufnahme von Musik die Töne jedes Instruments am Ende in einer einzigen kombinierten Schallwelle vorliegen. Zwei Instrumente, die dieselbe Note spielen erscheinen dann auf derselben Frequenz. Durch das Abgleichen der Klangfarben wäre eine Erkennung der individuellen Instrumente und Noten zwar theoretisch möglich, aber das ist mit einem grossen Aufwand verbunden und diese Erkennung ist nicht immer zuverlässig. Mein Visualisierungsprogramm kann lediglich die verschiedenen Frequenzen erkennen und ist deshalb nicht in der Lage individuelle Instrumente, die dieselbe Tonhöhe (Frequenz) spielen, zu unterscheiden. Meist reicht aber die Information über die Frequenzen, um eine brauchbare Animation zu generieren.

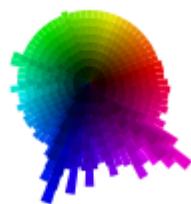
Ein weiteres Problem stellt die Perkussion dar: Der Ton eines lauten Schlagzeugs führt immer zu einem Ausschlag auf allen Frequenzen. Während eines Schlagzeug-Schlags ist der melodische Teil des Musikstückes also schlechter erkennbar. Oft ist die Darstellung des Rhythmus in der Visualisierung aber erwünscht, und so kann dieser anscheinende Nachteil sogar zu Gunsten einer schönen Animation genutzt werden.

Zusätzlich stellt auch die Rechenleistung des Computers eine Beschränkung dar: Das Zeichnen der Visualisierungen darf nicht zu lange Zeit beanspruchen, damit eine hohe Bildwiederholfrequenz und damit ein ruckelfreies Video garantiert werden kann.

## 6 Ergebnisse



**Abbildung K:** Laute Töne lösen bei der „Regen-Visualisierung“ die Darstellung dieser Kreise aus. Die Position auf der X-Achse entspricht der Frequenz der Töne.



**Abbildung L:** Die „Kreis-Visualisierung“: Der Winkel der Spitzen steht in Verbindung zur Tonhöhe, der Radius in Verbindung zu deren Lautstärke.

Das entstandene Visualisierungsprogramm ist in der Lage, zwei verschiedene Animationen zu beliebigen Audio-Dateien anzuzeigen. Knapp 1'250 Zeilen Python-Code beschreiben die Logik des Programms.

Die Umsetzung der skizzierten Visualisierungen gestaltete sich aufgrund der Einschränkungen des Programms als schwierig. Die Animationen erfüllen nicht ganz meine Erwartungen. Trotzdem zeigen sie, dass die Darstellung von Musik möglich ist und diese kreativ vielseitig ausgeführt werden kann. Auch einfache Darstellungen wie die dieser Arbeit erweitern im Zusammenspiel mit der Musik das Hörerlebnis.

Heute hört man Musik oft während man anderweitig abgelenkt ist; Im Gegensatz dazu kann man sich beim Schauen meiner Visualisierungen voll und ganz auf die Musik konzentrieren und erlebt diese dadurch intensiver.

Auf der beiliegenden CD befinden sich acht Videos mit Beispielen der zwei Visualisierungen sowie der Programmcode.

## 7 Quellen

Alle Abbildungen sind Eigenkreationen, wo nicht anders vermerkt.

Informationen über das VJing entstammen einem telefonischen Interview mit Michael Spahr aus Bern vom 23.06.2014.

### 7.1 Literatur

Matthew Neil Bain „Real Time Music Visualization: A Study in the Visual Extension of Music“ (Ohio State University 2008)

Jérôme Sueur „A very short introduction to sound analysis for those who like elephant trumpet calls or other wildlife sound“ (Muséum national d’Histoire naturelle 2014)

Sinclair User Magazine Ausgabe 7, 1987 - Werbung für „Sound to Light Generator“  
[www.worldofspectrum.org/showmag.cgi?  
mag=SinclairUser/Issue064/Pages/  
SinclairUser06400097.jpg](http://www.worldofspectrum.org/showmag.cgi?mag=SinclairUser/Issue064/Pages/SinclairUser06400097.jpg)

### 7.2 Internet

Whipple Library: Isaac Newton: [www.hps.cam.ac.uk/library/universalharmony/  
newton.html](http://www.hps.cam.ac.uk/library/universalharmony/newton.html) (Stand: 07.08.2014)

What is Music Visualization: [www.wisegeek.com/what-is-music-visualization.htm](http://www.wisegeek.com/what-is-music-visualization.htm)  
(Stand: 12.08.2014)

Audiovisualisierung als Unterhaltungsmedium: [de.wikipedia.org/wiki/  
Audiovisualisierung#Audiovisualisierung\\_als\\_Unterhaltungsmedium](http://de.wikipedia.org/wiki/Audiovisualisierung#Audiovisualisierung_als_Unterhaltungsmedium) (Stand: 12.08.2014)

Synästhesie: [de.wikipedia.org/wiki/Synästhesie#Nonverbal\\_akustisch](http://de.wikipedia.org/wiki/Synästhesie#Nonverbal_akustisch)  
(Stand: 08.08.2014)

CVM - Raumlichtkunst: [www.centerforvisualmusic.org/Raumlichtkunst.html](http://www.centerforvisualmusic.org/Raumlichtkunst.html)  
(Stand: 08.08.2014)

*F.C. Judd:* [videocircuits.blogspot.ch/2014/05/fc-judd.html](http://videocircuits.blogspot.ch/2014/05/fc-judd.html) (Stand: 08.08.2014)

*VJing:* [en.wikipedia.org/wiki/VJing](http://en.wikipedia.org/wiki/VJing) (Stand: 08.08.2014)

*Michal Levy - Giant Steps:* [www.michalevy.com/giant-steps](http://www.michalevy.com/giant-steps) (Stand: 12.08.2014)

*Esteban Díacono - Ljósið:* [portfolio.estebandiacono.tv/Olafur-Arnalds-Ljosid/](http://portfolio.estebandiacono.tv/Olafur-Arnalds-Ljosid/) (Stand: 12.08.2014)

*Matthias Dittrich - Narratives 2.0:* [www.matthiasdittrich.com](http://www.matthiasdittrich.com) (Stand: 12.08.2014)

*Video Beats - Music Visualizer:* [www.instructables.com/id/Video-Beats-Music-Visualizer/](http://www.instructables.com/id/Video-Beats-Music-Visualizer/) (Stand: 12.04.2014)

*Physics of Sound:* [www.podcomplex.com/guide/physics.html](http://www.podcomplex.com/guide/physics.html) (Stand: 12.08.2014)

*MIDI vs. Digital Audio:* [www.danmusic.com/midaudio.html](http://www.danmusic.com/midaudio.html) (Stand: 12.08.2014)

*The RIFF WAVE File Format:* [www.lightlink.com/tjweber/StripWav/WAVE.html](http://www.lightlink.com/tjweber/StripWav/WAVE.html) (Stand: 12.08.2014)

*Beispiel eines allgemein lesbaren WAVE-PCM-Formats:* [de.wikipedia.org/wiki/RIFF\\_WAVE#Beispiel\\_eines\\_allgemein\\_lesbaren\\_WAVE-PCM-Formats](http://de.wikipedia.org/wiki/RIFF_WAVE#Beispiel_eines_allgemein_lesbaren_WAVE-PCM-Formats) (10.08.2014)

*Fourier Analysis - Applications in signal processing:* [en.wikipedia.org/wiki/Fourier\\_analysis#Applications\\_in\\_signal\\_processing](http://en.wikipedia.org/wiki/Fourier_analysis#Applications_in_signal_processing) (Stand: 12.08.2014)

*An Interactive Guide to the Fourier Transform:* [www.betterexplained.com/articles/an-interactive-guide-to-the-fourier-transform/](http://www.betterexplained.com/articles/an-interactive-guide-to-the-fourier-transform/) (Stand: 12.08.2014)

*Sinusoid:* [de.wikipedia.org/wiki/Sinusoid](http://de.wikipedia.org/wiki/Sinusoid) (Stand: 12.08.2014)

*An Intuitive Discrete Fourier Transform Tutorial:* [www.practicalcryptography.com/miscellaneous/machine-learning/intuitive-guide-discrete-fourier-transform/](http://www.practicalcryptography.com/miscellaneous/machine-learning/intuitive-guide-discrete-fourier-transform/) (Stand: 12.08.2014)

*Leck-Effekt:* [de.wikipedia.org/wiki/Leck-Effekt](https://de.wikipedia.org/wiki/Leck-Effekt) (Stand: 12.08.2014)

*Fensterfunktion:* [de.wikipedia.org/wiki/Fensterfunktion](https://de.wikipedia.org/wiki/Fensterfunktion) (Stand: 12.08.2014)

*Wahrnehmung von Schallsignalen:* [de.wikipedia.org/wiki/Musikwahrnehmung#Wahrnehmung\\_von\\_Schallsignalen](https://de.wikipedia.org/wiki/Musikwahrnehmung#Wahrnehmung_von_Schallsignalen)

(Stand: 12.08.2014)

*Oktave:* [de.wikipedia.org/wiki/Oktave](https://de.wikipedia.org/wiki/Oktave) (Stand: 12.08.2014)

*Lautstärke:* [de.wikipedia.org/wiki/Lautstrke](https://de.wikipedia.org/wiki/Lautstrke) (Stand: 12.08.2014)

*A-Weighting:* [en.wikipedia.org/wiki/A-weighting](https://en.wikipedia.org/wiki/A-weighting) (Stand: 12.08.2014)

*Grafikbibliothek:* [de.wikipedia.org/wiki/Grafikbibliothek](https://de.wikipedia.org/wiki/Grafikbibliothek) (Stand: 12.08.2014)

*Cairo Graphics - Tutorial:* [www.cairographics.org/tutorial/](http://www.cairographics.org/tutorial/) (Stand: 12.08.2014)

## 8 Eidesstattliche Erklärung

Hiermit bestätige ich, André Rösti, die vorliegende Arbeit selbstständig und unter Angabe aller benötigten Quellen verfasst zu haben.

---

Uetendorf, 13.08.2014

## Anhang A: Interview mit Michael Spahr

### **Was war Dein erster Kontakt mit VJing? Wie bist Du ans VJing herangekommen?**

Das erste Mal erlebte ich dieses Phänomen in Clubs bei Konzerten. Vor etwa 15 Jahren lebte ich in Amsterdam. Ich war viel unterwegs und sah das das erste Mal bei Bands, vor allem auch bei den „Ninja Tune Label Nächten“. Coldcut waren die Pioniere des VJings. Das war das erste Mal das ich das live erlebt habe.

Gleichzeitig hatte es in Amsterdam offenes Fernsehen, wo man selbst experimentelle Videos machen konnte. Ich selbst habe dort mit Kollegen zusammen Fernsehen gemacht. Das war auch schon im Geist des VJings.

In der Schweiz erlebte ich das das erste Mal mit meinen Kollegen, die hier dann auch VJing machten.

### **Trittst du auch live auf?**

Heute trete ich nicht mehr immer live auf, nur zum Teil. Ich bin im Moment Redaktionsleiter der Inforeaktion von Radio Rabe, und das ist im Moment meine Hauptbeschäftigung. Ich bin im Moment eher mit Radio dran, obwohl ich vor allem mit Fernsehen gearbeitet habe. Ich habe inzwischen Familie und trete von dem her nicht mehr wirklich live auf.

Ich bin aber jahrelang live aufgetreten. Vor 15 Jahren, so anfangs 2000er-Jahre bin ich zeilenweise jede Woche irgendwo aufgetreten. Am Anfang vor allem auf Parties zusammen mit DJs, meistens auch mit anderen VJs. Wir hatten ein VJ-Kollektiv. Meistens zu zweit oder zu dritt machten wir nächtelang Visuals. Dann vor 10 Jahren kam ich etwas weg von den Parties und Clubnächten und hin zu Videos zu Musik, Theater, und auch anderen Künsten.

Das heisst ich bin jahrelang live aufgetreten, inzwischen weniger. Heute produziere ich mehr Sachen. Auch wenn ich live bin. Clubnächte sind hauptsächlich Improvisation; Man hat riesen Archive von Bilder und Filmen und man wusste nie am Anfang vom Abend, was passieren wird - bis 4 Uhr, 6 Uhr oder manchmal auch 9 Uhr am Morgen.

Heute mache ich auch viel konzentriertere Sachen, wo ich Filme sehr gut vorbereite. Aber häufig spiele ich immer noch Sachen live, wo ich das Tempo ändern kann, oder innerhalb eines Videos kann ich mit Loops arbeiten. Also mache ich immer noch einen Teil live.

**Zu der Zeit als du häufiger live aufgetreten warst; Ich nehme an da war zuerst der DJ da, und dann kamst du. Wie haben die Leute da reagiert? Bist du im Zentrum gestanden, oder warst du eher im Hintergrund um für das Ambiente zu sorgen? Wie muss man sich das vorstellen?**

Es gibt zwei Seiten davon:

Auf Visuals bezogen - Es ist etwas anders bei Konzerten oder Theater-Sachen, wo du eher im Vordergrund bist, da du die Sachen selbst entwickelst oder zusammen entwickelst. Das ist etwas anders bei DJs: Du weisst welcher Stil der DJ hat, und du weisst nicht welche Platten er auflegt. Genauso weisst der DJ nicht welche Visuals du spielst. Im Zentrum war also immer der DJ. In den Meisten Fällen - Es gibt Ausnahmen - wurdest du von einem Club angefragt, der zuerst einen DJ buchte und dann - wenn er genug Geld hatte - auch noch einen VJ buchte. Da machte ich mir keine Illusionen: Die Leute kommen nicht wegen den Visuals, sie sagen vielleicht: „Oh, da hat es noch Visuals“. Aber am Schluss wollen die Leute tanzen und da ist der DJ im Vordergrund.

Auf der Bühne ist es hingegen wieder lustig: Wir waren uns häufig viel mehr am bewegen als der DJ. Wir hatten meistens viele Geräte auf der Bühne, und die Leute sind dauernd auf die Bühne gekommen und sind auf die Bühne gekommen und haben mit uns gesprochen: „Könnt ihr noch etwas mehr House auflegen?“ oder „Könnt ihr noch etwas mehr Drum ,n Bass auflegen?“ Weil sie meinten wir waren die DJs, weil wir uns auf der Bühne so viel bewegt haben.

Was wir später machten war als Reaktion darauf das wir immer das 5. Rad am Wagen waren - das war etwa vor 10 Jahren - das ich mit meinen VJ-Kollegen uns eine eigene Party-Reihe aufzogen im Progger. Das hiess „Tanznacht“. Das war als es gerade eine neue Turnhalle gab in Bern und man auch etwas Narrenfreiheit hatte, weil die Turnhalle war

alt. Die Leute gingen dorthin, weil man in einer alten Turnhalle Party feiern kann. Da haben wir als VJ die DJs eingeladen. Damit haben wir den Spiess einmal umgekehrt. Und wir waren an den Gewinnen beteiligt, und die DJs hatten eine fixe Gage.

**Nachdem das du in Amsterdam zum ersten mal in Kontakt mit VJing kamst und dich dann entschieden hast, das selbst auch zu machen - Wie hast du das gelernt? Gibt es Schulen, oder war das Learning by Doing?**

Das war Learning by Doing.

Zu meinem VJ-Schaffen gehörte immer das ich selbst Sachen produziert habe. Viele VJs benutzen Samples und nehmen Sachen von anderen. Ich habe immer auch selbst gefilmt oder eigene Fotos gemacht die ich animiert habe. Da habe ich das Handwerk des Filmes schon gelernt, mit Kursen. Ich arbeitete dann auch beim Fernsehen mit Film und lernte da das Handwerk. Ich habe aber nicht eine Filmschule gemacht; Ich bin ursprünglich Historiker. Aber ich bin durch die Politik in die Medienwelt gekommen und habe so dann wiederum das Handwerk des Filmens gelernt. Ein gewisses Denken hat das bestimmt beeinflusst. Aber das VJing selbst konnte man nicht lernen. Es gab noch keine Kunsthochschulen. Transmediale Kunst und neue Medienkunst steckte noch in den Kinderschuhen. Man musste es selbst lernen.

Ein Kollege von mir aber, der dazumals die Software entwickelte die wir benutzen, der tat das im Rahmen einer Schule, eine eigene Software zu entwickeln. Er machte eine Schule zu neuen Medienproduktion und lernte dort ein gewisses Handwerk an Softwareentwicklung und benutzte das, um eine VJ-Software zu entwickeln.

**Zu deinen Visuals: Was ist deine Inspiration? Gibt es bestimmte Quellen oder lässt du dich von der Musik beeinflussen?**

[Von der Musik] liess ich mich eigentlich gar nicht beeinflussen.

Bei den Parties liess ich mich weniger von der Musik beeinflussen; bei bestimmten Musikprojekten spielte die Musik sehr stark eine Rolle. Es gab häufig auch Projekte das ich mit einem Komponist zusammen eine ganze Performance gemacht haben. Das heisst wir

haben Musik komponiert, und dazu die Bilder gemacht, und umgekehrt. Das war ein Zusammenspiel. Bei den DJs war ich eher frei mit meinem Kollektiv. Wir hatten einen Ehrenkodex, das wir keine bekannte Filme sampeln würden. Wir wollten kein Material von bekannten Filmen klauen, die dann jeder kennt. Ich hatte auch andere Kollegen, die dann zum Beispiel einen Laswerschwertkampf aus Star Wars benutz weil das cool aussieht. Das taten wir nicht. Wir haben eher unbekanntere Filmszene „kreativ geklaut“. Wir liessen uns inspirieren von Bildern die sich bewegen und dann auch selbst produziert. Von der Inspiration würde ich sagen: Ich bin nicht ein extrem grafischer Mensch. Es gibt viele Visuals die sehr grafisch sind, mit Flash-Vektor-Animationen und so, abstrakte Formen, ich war eher inspiriert von den Filmen an sich. Ich bin ein extremer Film-Freak, oder auch der Natur - dem Gegensatz. Du fährst durch einen Wald und lässt die Kamera laufen. Du siehst dann die Bäume, wie sie an dir vorbeiflitzen, und diese Bäume werden dann zwar auch so wie zu abstrakten Formen, die Art und Weise wie wir sie spielten wirkten sie sehr abstrakt und surrealisch, aber es sind eigentlich immer natürliche, organische Quellen.

Wobei mich die Stadt mehr inspiriert als die Natur. Alles in der Stadt. Normalerweise wenn die Leute etwas hässlich finden ist das wunderbar schön. Auch Orte, die nicht touristisch sind. Mich interessiert mehr das dort, wo die Leute durchlaufen. Fahrzeuge, Baustelle. Das Alltäglichste - zum Beispiel ein Menschenstrom in der U-Bahn. Das ist eigentlich alltäglicher als der Eiffelturm.

**Wenn du Musik hörst, hat das für dich ein Aussehen? Stellst du dir bereits im Kopf Visuals dazu vor, oder hörst du einfach die Musik ohne gross etwas zu sehen?**

Das ist schwierig zu sagen. Zu der Zeit als ich mehr zu Party-Musik Live-Visuals gemacht habe dachte ich noch eher: „Ah, das ist jetzt ein Song den ich verwenden könnte.“

Aber sonst denke ich das ich denke gleich wie die meisten Menschen, die Bilder im Kopf haben wenn sie Musik hören.

**Würdest du sagen das deine Videos eher die Musik darstellen oder diese erweitern?**

Ich würde eher sagen eine Erweiterung, vokalem der live-Aspekt. Du schaffst dort wie einen zusätzlichen Raum, wo die Leute etwas erfahren und erleben können. Es ist wie eine Erweiterung.

**Als du live aufgetreten bist, wie sah da dein Arbeitsablauf aus? Als du eine Idee hattest, wie hast du die umgesetzt?**

Zuerst: Wir haben immer Witze gemacht - wir waren die ersten die da waren, und die letzten die gingen. Das ist jetzt heute anders, aber vor 10-15 Jahren war es noch so, das nicht jeder Club einen Beamer hatte. Das heisst wir mussten alles hinbringen, das ganze Material. Auch als wir mit den DJs unterwegs waren und mit den Gagen verhandelt haben sagten wir, „Wir müssen etwas zur Seite legen für's Material.“ Die DJs sagten dann, „Wir auch, wir müssen noch Platten kaufen.“ Das war zu einer Zeit als man noch mehr Platten kaufte. Und dann sagten wir: „Nein, das müssen wir auch, wir müssen auch Videos beschaffen. Das ist etwas anderes. Aber ihr müsst nicht ein [PA / KA / ??] mitschleppen. Ihr müsst nicht ein Lautsprecher mitschleppen. Ihr müsst nicht ein Mischpult mitschleppen.“ Heute ist das zum Glück etwas anders: meistens hat es einen Beamer und ein langes Kabel. Aber manchmal fängt es dort schon an, das es dann zum Beispiel das falsche lange Kabel hat. Du musst technisch also alles sehr gut verstehen, du musst deine Geräte kennen. Du baust zuerst mal auf. Das heisst du bist der erste da. Und du musst dann alles wieder abbauen, das heisst du bist auch der letzte da. Vielleicht musst du noch eine Leinwand die du irgendwo aufgehängt hast wieder herunter holen, weil du sonst irgendwelche Leute erschlagen hättest. Das heisst du musst zuerst einmal warten bis der Club wieder leer ist.

Das ist der eine Aspekt. Dann hast du auf der Bühne - heute mache ich häufig alles auf dem Laptop, das heisst du hast nur noch einen Laptop - aber sonst hättest du häufig auch noch einen Videomischer.

## Videomischer, ist das ein Hardware-Gerät?

Genau. Ich hatte häufig auch eine Kombination in den letzten Jahren, da ich häufig auch noch mit Ton gearbeitet habe: Ein Mischpult auf dem du nicht nur Video zusammenmixen kannst, sondern auch noch Töne. Inzwischen gibt es aber mit High Definition diese HD-Mixer, und die sind extrem teuer, da dieser Markt noch eine Nische ist. Ich habe zum Beispiel kein HD-Mixer. Wenn ich mixen will, dann fahre ich immer noch auf dem alten Fernsehformat. Aber heute mische ich nicht mehr viel, und ich habe eigentlich nur noch eine Software. Ich brauche eine DJ-Software, mit der man auch Videos mischen kann.

Dann hast du live eventuell noch Kameras. Eine weile hatten wir auch DJ DVD-Player, wo man auf DVDs scratchen konnte. Das gab es, und wir waren wohl die ersten und einzigen in der Schweiz die das hatten, denn wir hatten einen Vertrag mit Pioneer, die die entwickelten, und wir konnten die in der Schweiz promoten. Das kam aber zu einer Zeit, als die Leute eher auf Software umgestiegen sind, und so konnten sich diese nicht mehr richtig etablieren. Sie waren auch viel zu teuer. Aber das ist eine andere Geschichte.

Dann gibt es das Inhaltliche: Wir haben oft als zwei Personen Dinge übereinandergemixt. Oder auch in der eigenen Software Videos übereinandegemixt. Du musst dir dann überlegen, z.B. - wir haben die Videos immer sehr kontrastreich gemacht, damit man alles das Weiss ist herausnehmen kann. Das ist so die Bluescreen-Technik, die Keying-Technik. Mit Bluescreen kannst du nicht unbedingt arbeiten, da du im Film nicht das blau weghaben möchtest. Wir haben also eher mit Schwarz und Weiss gearbeitet. also beispielsweise ein Film schwarzweiss oder mit starken Kontrasten, und dann einen farbig, und diese dann gemischt. Also wir haben Bilder improvisiert.

Wie ich sonst arbeite: Mein eigener Stil ist das ich Fotos nehme - so kreiere ich mein Material. Ich mache Fotos, überall wo ich hingehe. Ich nehme Objekte, Häuser, Statue, Strassenschiler, und dann schneide ich diese aus in einem Fotobearbeitungsprogramm. Dann animiere ich diese mit einem Trickfilmprogramm. Ich habe ein uraltes 2D-Trickfilmprogramm. Man kann ein Männchen zeichnen und dann mit Knochen das Männchen zum tanzen bringen. Ich bringe damit irgendwelche Objekte die eigentlich statisch sind zum bewegen. Ich benutze bis zu 5 verschiedenen Softwares um zum

Resultat zu gelangen, das ich möchte. Heute arbeite ich also eher mit solchen Collagen - mit Video-Collagen. Heute arbeite ich mehr mit Collagen die ich im Vorfeld produziere, da ich finde, diese sind schöner und ich kann ihnen eher meinen eigenen Stil aufdrücken. Früher hat man natürlich nächtelang einfach Collagen improvisiert. Das Problem damit - ich habe das auch erlebt - ich bin in Japan oder Berlin in einen Club hineingegangen und ich hätte nicht sagen können das das nicht ich und Miene Crew war die diese Visuals gemacht haben, weil das überall sehr ähnlich aussah auf der Welt, weil alle mit einer ähnlichen Technik gearbeitet haben.

**In deinen neueren Videos, möchtest du eine politische Aussage machen?**

Ja das ist zum Teil ganz bewusst politisch, oder das ich versuche Themen zu verarbeiten. Das ist der Unterschied zu früher. Heute versuche ich bewusst, Themen aufzuarbeiten.

Bei der Partyszene, manchmal dachte nwir „Oh wir sind so poetisch“ - aber manchmal haben das nur wir bemerkt, oder es war unsere interpretation. Aber ich bin eigentlich bewusst sehr politisch, gesellschaftliche Themen. Ich schaffe beim Radio bei einer polkt und gesellschaftssendung, wo ich täglich mit solchen Themen konfrontiert bin, wo ich mich auch als Historiker mit solchen Themen auseinandersetze. Dan verarbeite ich das sehr häufig in meiner Kunst. Ich finde meine Kunst soll auch politisch sein.

**Für meine eigene Mathearbeit mache ich selbst eine Musik-Visualisierung. Hättest du irgendwelche Tipps für mich, worauf ich achten sollte?**

Mein Tipp ist: Es ist schön wenn du ein eigenen Stil hast. Wenn du etwas machst, das sich unterscheidet. Manchmal braucht das aber 10 Jahre, das war bei mir zum Beispiel so. Ich habe lange gesucht bis ich das irgendwo gefunden habe.

Achte auf deinen Stil. Damit du dich unterscheidest, auf der ganzen Welt. Es gibt viele Videokünstler. Auf der anderen Seite würde ich sagen - das Ding ist - mit Videos - du kannst probieren: Du kannst ein Video machen und Musik darunterlegen und auf irgendeine Art passt es immer. Du hast also extrem viel Freiheit. Das kann dann sehr beliebig werden. Also solltest du darauf achten: Du kannst mit Zufällen arbeiten, experimentiere mit verschiedenen Bildern.

Probiere deine Vision irgendwie mit Bilder umzusetzen. Du musst ja keinen Video-Clip haben. Ein Video-Clip ist viel aufwändiger und du musst dich viel mehr an Regeln hast. Bei Visuals - geniesse die Freiheit. Du kannst Dinge machen die ein Filmemacher nicht darstellt. Du kannst zum Beispiel ein Achsensprung machen - als Filmer verwirrst du die Leute, aber als Visagist kannst du damit einen Schock auslösen.

Schau gut umher, lasse dich inspirieren.

Die Dinge die ich jetzt mache - ich habe einen extrem kleinen Output. Ich mache 2 - 3 Werke pro Jahr. Ich bin jetzt auch nicht darauf angewiesen das ich diese verkaufen kann. Lass dir lieber Zeit.