Computer Vision for Music Identification

Philip Kurmann, philip@kman.ch

22 April 2017

Abstract

Das Paper "Computer Vision for Music Identification" von Yan Ke, Derek Hoiem und Rahul Sakthankar beschreibt einen Algorithmus zur Identifikation von Musik Titel anhand einer verrauschten Audioaufnahme. Zur Bestimmung werden Methoden der Bilderkennung wie "Boosted Classifiers" und "Local Descriptor Based Object Recognition" verwendet.

Beim Erkennen von Musik ist man vor allem mit dem Problem konfrontiert, dass die Suchanfrage eine Tonaufnahme sein kann, welche eine schlechte Qualität aufweist oder stark verrauscht ist. Weitere Anforderungen an den Algorithmus: Hohe Erkennungsrate, hohe Präzision, Suche durch kurze Audioaufnahmen sowie schnelles Auffinden der Resultate. Dazu soll das System skalieren, damit auch eine grosse Musikdatenbank durchsucht werden kann.

Dieses Dokument fasst die wichtigsten Punkte des von Ke, Hoiem und Sukhuthankar beschriebenen Algorithmuses zusammen.

Warum das Thema Musikerkennung

Im Rahmen der Module Machine Learning und Deep Learning des CAS Machine Inteligence wurden divers Methoden des Maschinellen Lernens behandelt. Praktisch alle diese Methode bezogen sich jedoch auf das Thema Bilderkennung bzw. Textgenerierung / Textübersetzung.

Das Thema Musikerkennung könnte die Ausdehnung der gelernten Bilderkennungsmethoden auf Musik / Audio näher beleuchten.

Zusammenfassung

Spektrogramme

Um Musik mittels Bilderkennungsverfahren zu erkennen, müssen die Schallwellen in Bilder umgewandelt werden. Hierzu werden die Songs in kurze Schnipsel unterteilt, aus welchen mittels der "Short-Term Fourier Transformation" (SSTM) Spektrogramme erstellt werden. Die Spektrogramme repräsentieren die Stärke von 33 logarithmischen Frequenzbänder zwischen 300 und 2000 Hz.

Ohne weitere Filterung kann bereits in diesen Spektrogrammen eine gewisse Ähnlichkeit von verrauschten Aufnahmen mit den Originalen erkannt werden (siehe Abbildung 1b).

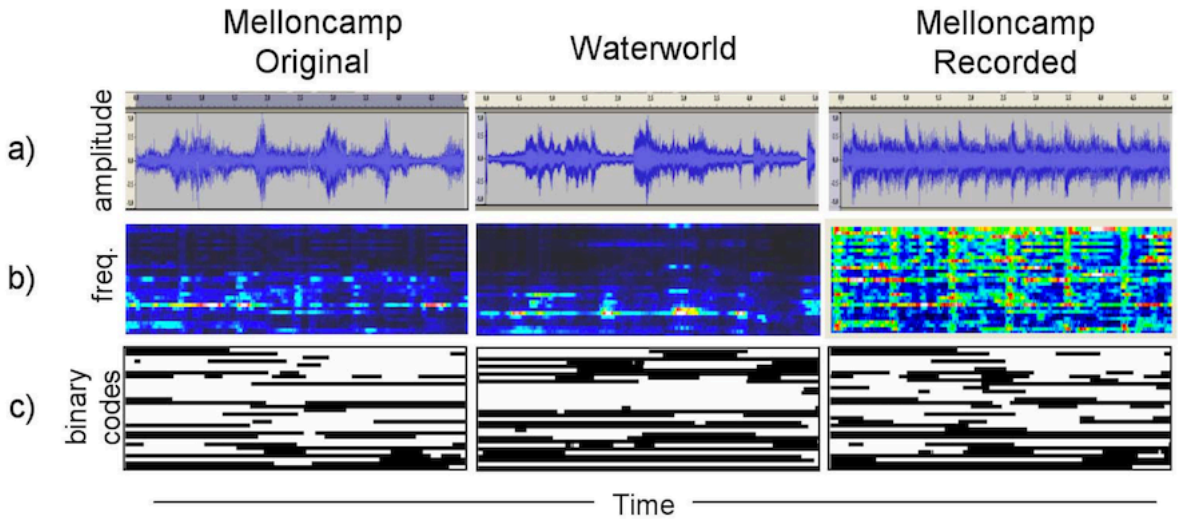

Figure 1: Darstellungen von Audiosignalen (Ke, Hoiem und Sukthankar 2005, p. 2)

Filter

Wenngleich im Spektrogramm bereits Ähnlichkeiten erkennbar sind, ist das einfache Vergleichen von Spektrogrammen nicht geeignet, da es zu ungenau und zu langsam ist. Um diesen Problemen zu begegnen, werden Filter gelernt, welche die zu erwarteten Störungen herausfiltern, jedoch die notwendigen Information erhalten. Mittels Machine Learning werden aus einer relativ grossen Menge an zur Verfügung stehenden Filter die relevanten ausgewählt.

Die Filter des beschriebenen Systems weisen folgende Charakteristiken auf: Basis Frequenz von 1-33, Bandweite von 1-33 und Zeit von 1 Frame (11.6ms) bis 82 Frames (951ms). Anhand dieser Parameter erhält man ca. 25'000 unterschiedliche Filter, wovon 32 Filter ausgewählt werden. Durch wiederholtes Anwenden der Filter auf Fenster von 2048 Samples, welche um jeweils 64 Samples weitergeschoben werden, erhält man eine weitere Abstraktion des Audiosignals, welches robust gegenüber Rauschen ist (siehe Abbildung 1c).

Pairwise Boosting

Die Auswahl der Filter wird mittels dem Pairwise Bossting Verfahren durchgeführt (siehe Jang u. a. 2009). Durch die Selektion der richtigen Filter können zur Anfrage passende Audioschnipsel effektiv ermittelt werden.

Okklusionsmodell

Um eine effektive Erkennung zu gewährleisten muss zusätzlich detektiert werden können, ob ein bestimmtes Audioschnipsel vorwiegend aus Musik bzw. aus Hintergrundgeräuschen besteht. Solche, welche vorwiegend aus Störungen bestehen werden als Okklusion bezeichnet und sollen nicht in die Abfrage mit einfliessen.

Abfrage

Alle Songs in der Datenbank werden in kleine Stücke unterteilt und anhand der oben beschriebenen Kriterien indexiert. Bei der Abfrage wird nun das Abfrage-Sample wiederum in Stücke unterteilt und jedes Schnipsel

mit den indexierten verglichen. Die Datenbankgrösse und die hohe Zahl von Abfragen welche durch die Zerstückelung entsteht, erfordern einen effektiven Algorithmus zur Bestimmung der Ähnlichkeiten.

Als effiziente Methode hat sich das Abspeichern der M beschreibenden Parameter in einer Hash-Tabelle erwiesen. Alle Parameter mit einem Hamming-Abstand von 2 werden als Near-Neighbors bezeichnet. Bei der Abfrage werden in einem ersten Schritt alle Resultate gesucht, welche eine Hamming Distanz von 0 aufweisen. Danach werden M weitere Proben durchgeführt, bei welcher jedes mal ein einzelnes Bit gedreht wird. Die so erhaltenen Resultate weisen eine Hamming-Distanz von 1 auf. Als letzten Schritt werden alle Kombination mit 2 gedrehten Bits gesucht, wodurch die Resultate mit Hamming-Distanz 2 gefunden werden. Obwohl man es nicht erwarten würde, hat sich diese einfache Methode als sehr effizient herausgestellt.

Nachdem alle Near-Neighbor Schnipsel gefunden wurden, muss man den Song identifizieren. Dies wird anhand einer geometrischen Verifikation durchgeführt, welche Ähnlichkeiten mit der Objekterkennung durch Local Features aufweist. Für jeden möglichen Song wird geprüft, ob sich die Deskriptoren stabil über die Zeit verhalten. Mittels RANSAC wird über die Zeitausrichtung der ermittelten Kandidaten iteriert und die so erhaltene Fehleranzahl (eM-Score) wird als Distanzmetrik zur Bestimmung verwendet.

Random Sample Consensus (RANSAC)

RANSAC ist ein iterativer Algorithmus zur Schätzung eines Modells innerhalb einer Reihe von fehlerbehafteten Messwerten und wurde 1985 von Martin Fischler und Robert Bolles entwickelt. Er weisst eine hohe Robustheit gegenüber Ausreissern auf und findet deshalb Verwendung in diversen Machine Learning Verfahren, bei denen der Least Squares Ansatz schlecht funktioniert.

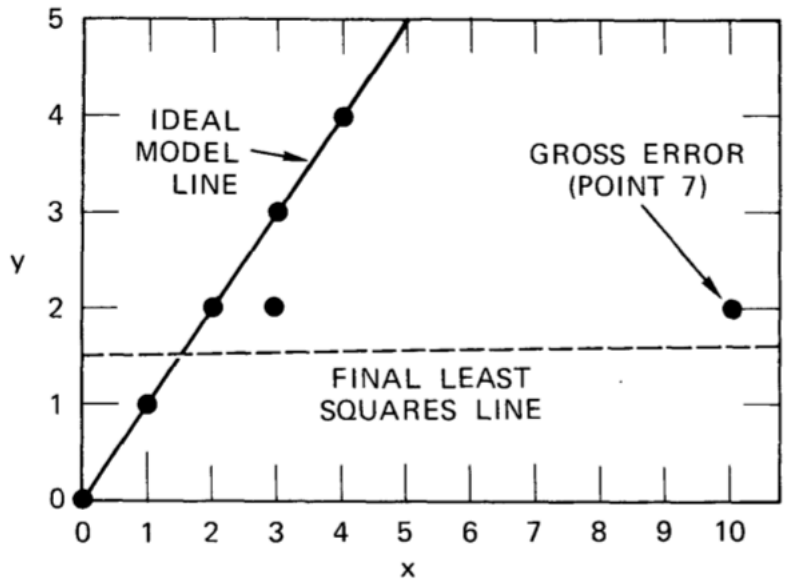

Figure 2: Angleichung einer Geraden an eine Punktwolke mit hohem Ausreisser (Fischler und Bolles 1981, p. 382)

Voraussetzung für RANSAC ist, dass mehr Datenpunkte vorliegen, als für die Bestimmung des Modells benötigt werden. Die Vorgehensweise zur Bestimmung eines guten Modells mittels RANSAC kann folgendermassen zusammengefasst werden:

1. Zufällige Auswahl von Datenpunkte, die zur Bestimmung des Modells notwendig sind. (Für eine Gerade beispielsweise 2 zufällige Datenpunkte).

2. Ermittlung der Modellparameter (Bei einer Geraden beispielsweise a und b der Geradengleichung $y = ax + b$)
3. Ermittlung der Anzahl Punkte, die weniger weit weg vom Modell sind, als ein bestimmter Schwellwert. (eM-Score)
4. Schritte 1-3 n mal wiederholen und das beste Modell bestimmen.

Test Setup und Performance

Der Training-Datensatz bestand aus 78 Songs. Daraus wurden 2 Test-Datensätze mit Störungen erstellt werden. Hierzu wurden die Songs auf schlechten Lautsprecher abgespielt und mittels günstigen Mikrofonen wieder aufgenommen. Mittels diesem Setup wurden 2 Testsets erstellt: Test-Set A, bei welchem die Songs mit tiefer Lautstärke abgespielt wurden und Test-Set B, bei welchem sehr laute Hintergrundgeräusche zu hören waren.

Die Datenbank umfasste 1862 Songs von unterschiedlichen Musikrichtungen. Mit dem Test-Set A konnten 90% der Songs erkannt werden mit einer Genauigkeit von 96%. Beim schwierigeren Test-Set B konnten immerhin 80% der Songs mit einer Genauigkeit von 93% erkannt werden.

Schlussfolgerung

"Computer Vision for Music Identification" erklärt auf anschauliche Weise, wie zuverlässige Systeme zur Identifizierung von Songs gebaut werden können. Zum Verständnis des kompletten Algorithmuses (beispielsweise das RANSAC Verfahren) müssen jedoch weiterführende Quellen konsultiert werden.

Obwohl im Dokument Ke, Hoiem und Sukthankar (2005) Verfahren der Bilderkennung eingesetzt werden, ist es doch ein sehr spezifisches Problem das behandelt wird. Der erste Teil, die Umwandlung der Audiosignale in die Pairwise Boosted Audio Fingerprints (siehe Jang u. a. 2009) ist jedoch eine zuverlässige und robuste Methode, um Audio Signale mittels Machine Learning zu verarbeiten.

Referenzen

Fischler, Martin A. und Robert C. Bolles. 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM* 24, Nr. 6: 381–395.

Jang, Dalwong, Chang D. Yoo, Sunil Lee, Sungwoong Kim und Ton Kalker. 2009. Pairwise Boosted Audio Fingerprint.

Ke, Yan, Derek Hoiem und Rahul Sukthankar. 2005. Computer Vision for Music Identification.