

Prof. Dr.-Ing.habil.  
**Dirk Joachim Lehmann**  
Data Science in IoT  
Fakultät für Informatik  
[di.lehmann@ostfalia.de](mailto:di.lehmann@ostfalia.de)  
[www.dirk-lehmann.de](http://www.dirk-lehmann.de)

## **Bild-basierte Digitale Sonifikation**

### **Hintergrund**

In der Bild-basierten digitalen Sonifikation werden Objekte von Interesse bild-basiert detektiert, konkrete Parameter zum Objekt, wie Geschwindigkeit, Richtung, und Distanz aus den Bild-basierten Eigenschaften geschätzt und auf ein charakteristisches Tonmuster abgebildet. So können Kraftfahrzeug-Fahrer/sehbehinderter Personen u.Ä. in die Lage versetzt werden konkrete Situationen in ihrer Umwelt intuitiv bzw. „hörend“ wahrzunehmen (heranfahrendes Kind auf Fahrrad beim Linksabbiegen, Wildwechsel, etc.).

Der Idee nach: „Wir können etwas übersehen, aber wir können nicht weghören“.

Dadurch ist es möglich die Umgebung hörend und unterschiedliche Objekte unterscheidbar /diskriminierbar wahrzunehmen. Insbesondere werden die Art des gehörten Objektes, ihre ungefähre Distanz und ihre Richtung sowie Bewegungsrichtung (steht das Objekt still / bewegt sich das Objekt auf „mich“ zu, etc.) hörbar unterscheidbar, wodurch eine Interaktion mit der Umgebung ermöglicht wird, basierend rein auf akustische Informationen.

---

### **Abgrenzung gegenüber analoge Sonifikation**

Sonifikation selbst ist bereits seit längerer Zeit bekannt und im praktischen Einsatz. Beispielsweise im Echolot in Unterseebooten, oder als Einparksensor in ihrem Fahrzeug (meist als Ultraschallsensor). Diese Form entspricht einer analogen Sonifikation: hierbei werden über Signal-Laufzeitverfahren Distanzen zu Objekten ermitteln, welche ein ausgesandtes Signal reflektieren. Die Laufzeiten werden auf einen Ton unterschiedlicher Höhen (Frequenzen) oder Lautstärken (Amplituden) abgebildet. Eine Unterscheidung der Objekte, welche das Signal reflektiert haben, ist kaum möglich. Aus diesem Grund reagiert ihr Einparksensor mit dem gleichen Warnton, wenn sie zu nah an eine Pflanze oder zu nah an zu hohes Graß heranfahren oder aber sie zu nah an eine Wand heranfahren. Wünschenswert wäre eigentlich lediglich ein Warnton bei dem Fall mit der Wand, weil die Pflanze für Ihr Fahrzeug kaum eine Gefährdung darstellt.

---

### **Vorteile der bild-basierten digitalen Sonifikation**

In der bild-basierten digitalen Sonifikation ist diese Unterscheidung unterschiedlicher Objekte möglich. Mit Hilfe von bild-basierten Detektionsmethoden aus dem Bereich der Computer Vision (SSD, Yolo, maskRCNN, Yoloact++, etc.) werden zunächst relevante Objekte von Interesse in Echtzeit erkannt (z.B.

eine Wand, oder ein heranfahrendes Kind, etc.). Uninteressante Objekte – wie hohes Graß beim einparken – werden nicht betrachtet. Diesbezüglich wird als techn. Setup lediglich eine Monokamera (Webcam etc.) benötigt und ein handelsüblicher Arbeitsrechner. Nun werden interessante Parameter für dieses detektierte Objekt von Interesse über Methoden der Bildverarbeitung geschätzt, wie z.B. die Entfernung/Distanz und die Richtung des Objektes. Die folgende Abbildung stellt den Prozess dar:



- Erkenne relevante Objekte und bilde dieses auf charakteristisches Tonmuster für den Fahrer ab

### Technische Realisierung der Bild-basierten digitalen Sonifikation

Ziel dieses Projektes ist es ein generalisiertes Sonifikationssystem in Python zu entwickeln und praktisch zu evaluieren. Die technische Realisierung der Bild-basierten digitalen Sonifikation ist schematisch in *Abb.: Technische Realisierung Sonifikation* dargestellt und wird im Folgenden detailliert beschrieben.

Zur technischen Realisierung der Sonifikation...

- 0) ... wird ein Beschreibungsformat benötigt um ein Sonifikations-Modell beschreiben und ablegen zu können. Wir bezeichnen dieses als **Sonifikation-Model Datenformat (SMD)**.
- 1) ... wird ein **SMD-Erstellungswerkzeug** benötigt, um SMD-Dateien durch einen Nutzer erstellen zu können. Die geschieht in einer initialen Sonifikation-Modell-Erstellungsphase.
- 2) ... wird ein **SMD-Anwendungssystem** benötigt, um SMD-Daten laden zu können und auf aktuelle Bilddaten aus angeschlossenen Kamerasystemen zur Ton-Muster-Synthese anwenden zu können.

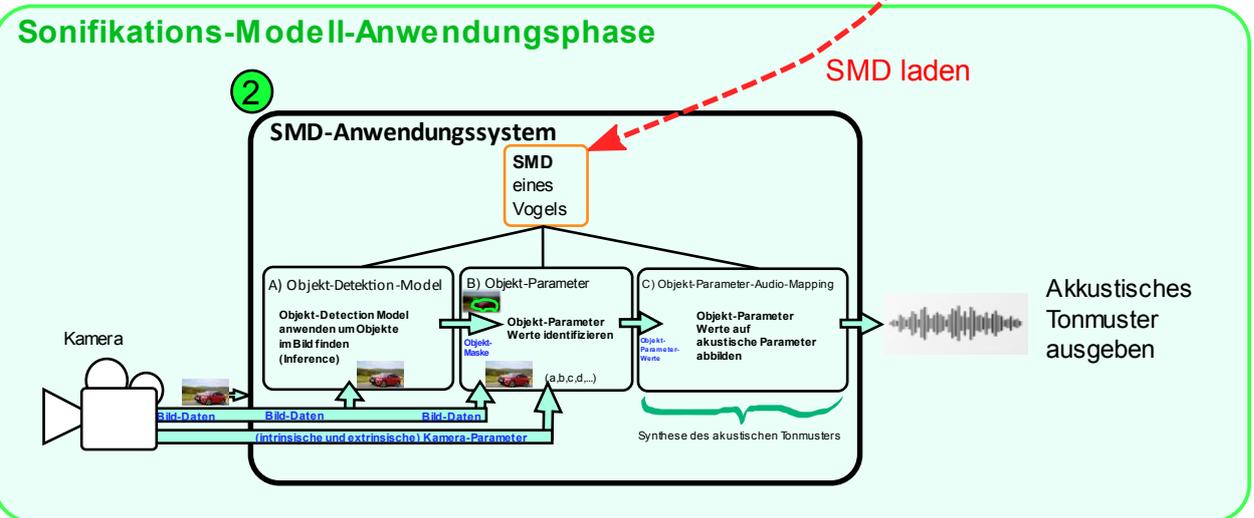
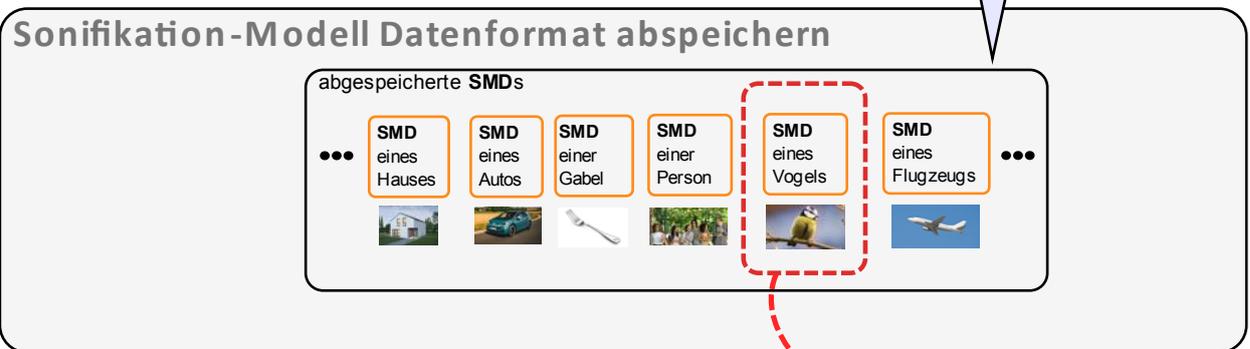


Abb.: Technische Realisierung Sonifikation

---

## Zu 0) Das Sonifikation-Model Datenformat (SMD)

Eine erste Herausforderung ist es ein geeignetes Datenformat zu definieren für ein praktikabel einsetzbares Sonifikationsmodell.

Ein **Sonifikation-Modell Datenformat (SMD)** beinhaltet folgende Beschreibungen und Datenobjekte:

A) Ein *Objekt-Detektion-Model* **OD (Objektklasse)**, welches eine **Objektklasse** von Interesse im Bild detektiert, wie ein „Fahrzeug“ oder ein „Haus“ etc.

⇒ In der Praxis, werden wir diesbezüglich idealerweise Neuronale-Netze einsetzen, welche für die Objektdetektion ausgelegt sind, wie *maskR-CNN* [11], *Single-Shot-Multibox-Detektor* (SSD) [12], *You Only Look Once* (Yolo) [12] etc.

B) *Objekt-Parameter*, welche zur späteren Sonifikation-Modell-Anwendungsphase aus dem Bild (aus der Bild-Region in dem das erkannte Objekte gefunden wurde) abgeleitet werden sollen. Objekt-Parameter welche zur Verfügung gestellt werden sollen sind:

- Distanz zum Objekt **d**
- Orientierung zum Objekt **r**
- rel. Bewegungsrichtung des Objektes **vr**
- rel. Geschwindigkeit des Objektes **v**

C) *Objekt-Parameter-Audio-Mapping*: Im *Objekt-Parameter-Audio-Mapping* wird beschrieben, in welcher Art und Weise die *Objekt-Parameter* auf *akustische Parameter* abgebildet werden. Solche akustischen Parameter sind:

- Die **Frequenz / Tonhöhe**
- Die **Amplitude / Lautstärke**
- Den **Ton-Rhythmus**
- Die **Toncharakteristik** (wie ein Instrument [Gitarre, Violine], Perkussion, oder ein Geräusch etc.)

Die ausgewählten *Objekt-Parameter* werden im *Objekt-Parameter-Audio-Mapping* den möglichen *akustischen Parameter* zugeordnet:

*Objekt-Parameter* => akustische Parameter

Ein Beispiel:

- Distanz zum Objekt **d** => **Amplitude** => **Violine**
- Orientierung zum Objekt **r** => **Amplitude** => **Violine**

---

## Zu 1) SMD-Erstellungssystem

Im **SMD-Erstellungssystem** wird ein konkretes Sonifikation-Modell im Sonifikation-Model Datenformat (SMD) erstellt und abgespeichert. Als Beispiel, kann es sich um ein Sonifikation-Modell handeln, um ein Objekt „Auto“ zu erkennen, zudem die Objekt-Parameter *Distanz zum Objekt*, mit dem Ziel dieses auf die *Tonhöhe* eines konkreten Tones abzubilden. Um dies praktisch zu realisieren werden drei System-Komponenten benötigt/müssen entwickelt werden:

- (a) ein **Trainingssystem für Objekterkennung**, um den Bild-basierten Objekterkennung des SMD zu trainieren/modellieren/erstellen.

Eingabe: Bilder und Ihre Labels, sowie weitere Nutzerinformationen, wie die Auswahl des konkreten Objekt-Detektion-Modells

Ausgabe: Einsatzfähig trainiertes Objekt-Detektion-Modells

---

- (b) ein **Objekt-Parameter-Selektion-System**, um festzulegen, welche Parameter (Distanz, Orientierung,...) genutzt werden sollen, um diese in der Sonifikation-Modell-Anwendungsphase auf Töne und ein Akustisches-Muster abzubilden.

Eingabe: Selektieren von Objekt-Parameter von Interesse durch den Nutzer

Ausgabe: Menge von Objekt-Parametern von Interesse

---

- (c) eine **Objekt-Parameter-Audio-Equalizer-Mapping-System**, um festzulegen, in welcher Art und Weise, die selektierten Parameter auf Töne abgebildet werden. Beispielsweise kann der Parameter *Distanz zum Objekt* zugewiesen werden zu der Tonhöhe eines Instrumentes wie eine Violine.

Eingabe: Abbildung von der Menge von Objekt-Parametern von Interesse auf die akustischen Parametern

Ausgabe: Beschreibung der akustischen Parameter

---

## Zu 2) SMD-Anwendungssystem

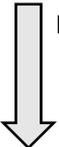
Im **SMD-Anwendungssystem** wird eine konkrete SMD-Datei geladen und die Tonmuster-Synthese realisiert. Der Prozess zur Tonmuster-Synthese muss implementiert werden und ist wie folgt realisiert:

Über eine Monokamera werden Bilder in das Anwendungssystem eingespielt



A) Objekt-Detektion-Modell

⇒ Auf die Bilder wird das Objekt-Detektion-Modell des SMD angewandt: Objekt-Detektion Modell anwenden um Objekte im Bild zu finden.

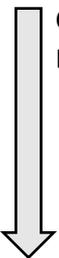


Eingabe: Bilder aus Kamera in Echtzeit, intrinsische und extrinsische Kameraparameter

Ausgabe: Beschreibung der akustischen Parameter

B) Objekt-Parameter

⇒ Anhand der Bilder und der detektierten Objekt-Maske aus dem Schritt davor sind die Objekt-Parameter mit Methoden der Bildverarbeitung/Computer Vision: Objekt-Parameter Werte identifizieren.



Eingabe: Bilder aus Kamera in Echtzeit, intrinsische und extrinsische Kameraparameter, Objekt-Maske aus dem Schritt davor, Menge der Objekt-Parameter die gefunden werden sollen.

Ausgabe: Gefundene Objekt-Parameter Werte

C) Objekt-Parameter-Audio-Modell

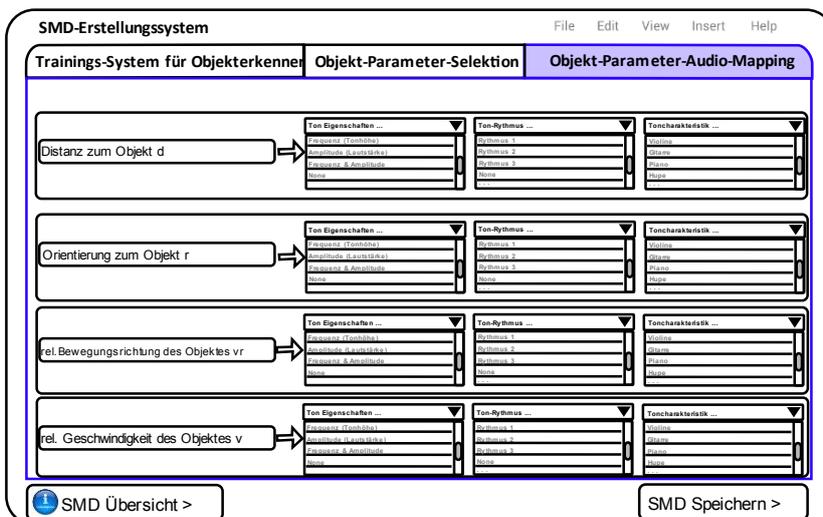
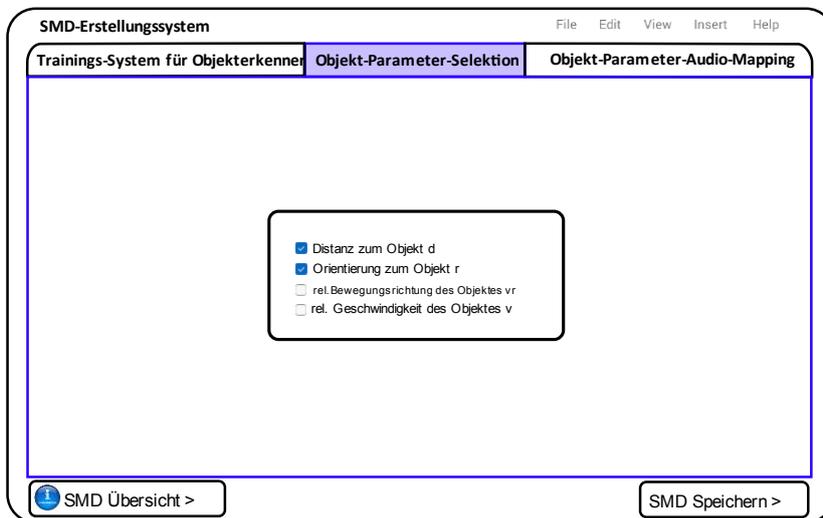
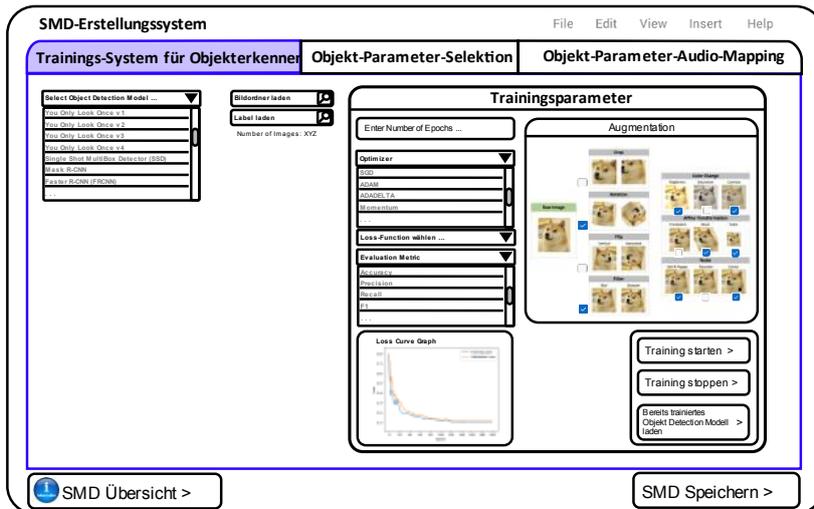
◦ Die gefundenen Objekt-Parameter Werte werden auf die akustischen Parameter abgebildet und daraus ein akustisches Ausgangssignal synthetisiert und ausgegeben

Eingabe: Objekt-Parameter Werte

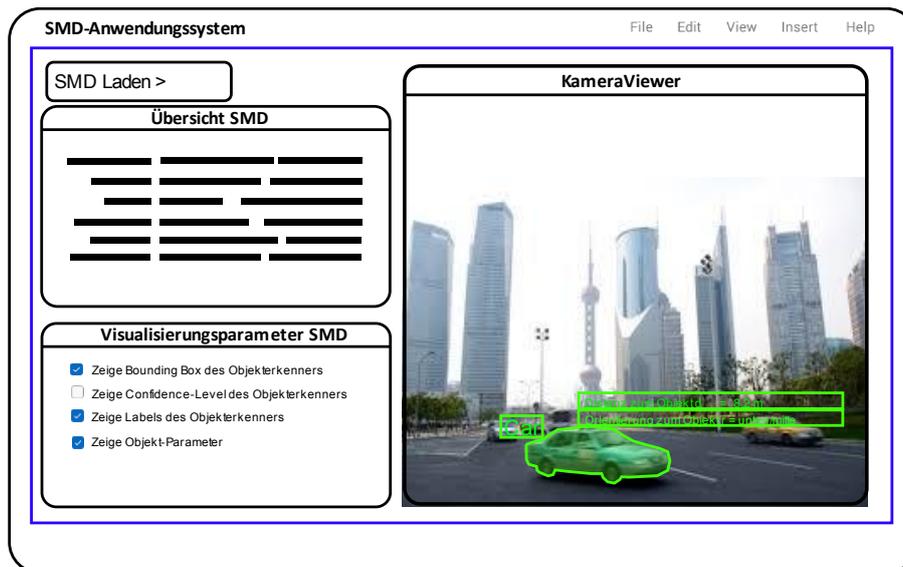
Ausgabe: akustisches Tonmuster

---

Im Folgenden sehen sie mögliche Nutzerinterfaces für das **SMD-Erstellungssystem...**



... und für das **SMD-Anwendungssystem**



---

## Angebot

Im Rahmen ihres *Praxisprojektes*, *Masterseminars*, *Masterprojektes*, ihrer *Bachelorarbeit*, *Masterarbeit* oder ähnlichen Studienleistungen - wie z. B. einem *interdisziplinären Digitalisierungsprojekt* - können sie gerne diese Aufgabe bearbeiten

Melden Sie sich gerne bei mir unter: [di.lehmenn@ostfalia.de](mailto:di.lehmenn@ostfalia.de)

---

## Aufgaben/Arbeitspakete

Das Projekt besteht aus klar voneinander abgrenzbaren Arbeitspaketen (AP):

- **AP1) Sonifikation-Modell Datenformat (SMD)**

Beispiele für in diesem Paket umzusetzende Features:

- SMD-Datenstruktur erstellen
- Funktion „SMD abspeichern“ implementieren
- Funktion „abgespeicherte SMD in SMD-Datenstruktur laden“ implementieren
- ...

- **AP2) SMD-Erstellungssystem-Backend**

Beispiele für in diesem Paket umzusetzende Features:

- SMD-Erstellungssystem-Backend: Trainings-System für Objekterkennung
- SMD-Erstellungssystem-Backend: Trainings-System für Objekterkennung => Objekt-Detektionsmodelle zur Verfügung stellen
- SMD-Erstellungssystem-Backend: Trainings-System für Objekterkennung => Trainingsbilddaten laden
- SMD-Erstellungssystem-Backend: Trainings-System für Objekterkennung => Label für das Training laden
- SMD-Erstellungssystem-Backend: Trainings-System für Objekterkennung => Augmentationsparameter für Training integrieren
- SMD-Erstellungssystem-Backend: Trainings-System für Objekterkennung => Trainingsparameter, wie Wahl Optimizer, Loss-Funktion, Evaluation Metric etc., Wahl des Anteils von Trainingsdaten zu Validation-Daten, Anzahl Epochen, etc. integrieren
- SMD-Erstellungssystem-Backend: Trainings-System für Objekterkennung => "Training starten" implementieren
- SMD-Erstellungssystem-Backend: Trainings-System für Objekterkennung => "Training stoppen" implementieren
- SMD-Erstellungssystem-Backend: Trainings-System für Objekterkennung => "Bereits Trainiertes Objekt Detection Modell laden" implementieren

- SMD-Erstellungssystem-Backend: Objekt-Parameter-Selektion => Auswahl von Objekt-Parametern in SMD-Datenstruktur hinterlegen
- SMD-Erstellungssystem-Backend: Objekt-Parameter-Audio-Mapping System => akustische Parmater: "Toneigenschaften Wahl" implementieren
- SMD-Erstellungssystem-Backend: Objekt-Parameter-Audio-Mapping System => akustische Parmater: "Ton-Rhythmus Wahl" implementieren
- SMD-Erstellungssystem-Backend: Objekt-Parameter-Audio-Mapping System => akustische Parmater: "Toncharakteristik" implementieren
- SMD-Erstellungssystem-Backend: Objekt-Parameter-Audio-Mapping System => "Distanz zum Objekt d" auf "akustische Parmater abbilden" implementieren
- SMD-Erstellungssystem-Backend: Objekt-Parameter-Audio-Mapping System => "Orientierung zum Objekt r" auf "akustische Parmater abbilden" implementieren
- SMD-Erstellungssystem-Backend: Objekt-Parameter-Audio-Mapping System => "rel. Bewegungsrichtung des Objektes vr" auf "akustische Parmater abbilden" implementieren
- SMD-Erstellungssystem-Backend: Objekt-Parameter-Audio-Mapping System => "rel. Geschwindigkeit des Objektes v" auf "akustische Parmater abbilden" implementieren
- ...

### - AP3) SMD-Anwendungssystem-Backend

Beispiele für in diesem Paket umzusetzende Features:

- SMD-Anwendungssystem-Backend: Kamera-Connector implementieren
- SMD-Anwendungssystem-Backend: abgespeicherte SMD-Dateien in interne SMD-Datenstruktur laden
- SMD-Anwendungssystem-Backend: SMD.Objekt-Detection-Model auf Bilddaten anwenden
- SMD-Anwendungssystem-Backend: Umgang mit Mehrfacherkennungen bei der Objekt-Erkennung für die spätere Tonmuster-Synthese
- SMD-Anwendungssystem-Backend: SMD.Objekt-Parameter."Distanz zum Objekt d" aus den Bildern und dem erkannten Objekt berechnen
- SMD-Anwendungssystem-Backend: SMD.Objekt-Parameter."Orientierung zum Objekt r" aus den Bildern und dem erkannten Objekt berechnen
- SMD-Anwendungssystem-Backend: SMD.Objekt-Parameter."rel. Bewegungsrichtung des Objektes vr" aus den Bildern und dem erkannten Objekt berechnen
- SMD-Anwendungssystem-Backend: SMD.Objekt-Parameter."rel. Geschwindigkeit des Objektes v" aus den Bildern und dem erkannten Objekt berechnen
- SMD-Anwendungssystem-Backend: Synthese des akustischen Tonmuster aus den vorher berechneten/festgelegten akustischen Parametern
- SMD-Anwendungssystem-Backend: akustischen Tonmuster dem Nutze ausspielen
- ...

### - AP4) User Graphic Interface/Nutzerinterface für bild-basierte Sonifikation

Beispiele für in diesem Paket umzusetzende Features:

- UI: SMD-Erstellungssystem-Frontend: Trainings-System für Objekterkenner UI implementieren
- UI: SMD-Erstellungssystem-Frontend: Objekt-Parameter-Selektion UI implementieren
- UI: SMD-Erstellungssystem-Frontend: Objekt-Parameter-Audio-Mapping UI implementieren
- UI: SMD-Anwendungssystem-Frontend UI implementieren
- ...

Zu den Arbeitspaketen hinzu kommen notwendige Recherchetätigkeiten, Make-Or-Buy-Entscheidungen, Beachtung von Lizenzfragen, Aspekte der Continuous Integration und des Code-Managements, Dokumentationsaufgaben, Fragen zum Aufsetzen/Deployment und der Migration von Entwicklersystemen (und zum Projektmanagement), wie sie in der Softwareentwicklung üblich sind.

Im Rahmen einer Projektbearbeitung wird nicht erwartet, dass unmittelbar alle APs bearbeitet werden können. Je nach Umfang Ihrer zu erbringenden Studierendenleistung können Teilaspekte einzelner APs bearbeitet werden. Den konkreten, jeweiligen Umfang stimmen wir im Vorfeld gerne gemeinsam ab.

---

## Vorkenntnisse

Es ist hilfreich – aber keine Voraussetzung – wenn Sie Vorkenntnisse/Interesse mitbringen in

- Softwareentwicklung

- Python
  - Grundkonzepte der Bildverarbeitung / Computervision
  - Grundkonzepte des Maschinellen Lernens / des Deep Learnings
  - Data Engineering
- 

## Vorarbeiten/Voraussetzungen

Studentische Arbeiten zum Thema Sonifikation

Objekterkennung in Bildern zur Vertonung von Umgebungssituationen, Hannes Stocker, Bachelor Thesis, Universität Magdeburg, [0]

[http://46.38.235.241/webpage/dirkfiles/betreutearbeiten/Stocker\\_BA\\_2019.pdf](http://46.38.235.241/webpage/dirkfiles/betreutearbeiten/Stocker_BA_2019.pdf)

Kurzeinführung in Python:

[https://www.youtube.com/watch?v=x\\_kYpwi1L1k](https://www.youtube.com/watch?v=x_kYpwi1L1k)

OnboardingProzess

Alle wichtigen Zugänge einrichten um die Arbeit am Projekt aufnehmen zu können:

<http://46.38.235.241/webpage/dirkfiles/misc/onboarding/OnboardingProzess.pdf>

---

## Organisatorisches

Projektmanagement per Scrum in Trello:

<https://trello.com/b/p8jjlseu/bild-basierte-digitale-sonifikation>

Projektcode-Management per GitHub:

<https://github.com/DirkJLehmann/BildbasierteDigitaleSonifikation.git>

Projektdokumentations-Management:

[https://docs.google.com/document/d/1jDu1SgSVyKXYY-O50mJv\\_SVC8fpxIdx5eV6Txw3KYk/edit?usp=sharing](https://docs.google.com/document/d/1jDu1SgSVyKXYY-O50mJv_SVC8fpxIdx5eV6Txw3KYk/edit?usp=sharing)

---

## Referenzen

[0] Feature Selektion und deren Selektionsstrategien für die Objekterkennung in Bildern zur Vertonung von Umgebungssituationen, Hannes Stocker, Bachelor Thesis in Computer Science, Fakultät für Informatik, Otto-von-Guericke-Universität zu Magdeburg, 2019,

[http://46.38.235.241/webpage/dirkfiles/betreutearbeiten/Stocker\\_BA\\_2019.pdf](http://46.38.235.241/webpage/dirkfiles/betreutearbeiten/Stocker_BA_2019.pdf)

[1] MUSART, MUSical Audio transfer function Realtime Toolkit, Abigail J. Joseph, Suresh K. Lodha, Proceedings of the 2002 International Conference on Auditory Display, Kyoto, Japan, July 2-5, 2002

<https://smartech.gatech.edu/bitstream/handle/1853/51366/JosephLodha2002.pdf?sequence=1&isAllowed=y>

[2] Sonification Sandbox, Georgia Tech Sonification Lab, Bruce Walker

[http://sonify.psych.gatech.edu/research/sonification\\_sandbox/](http://sonify.psych.gatech.edu/research/sonification_sandbox/)

[3] SONART : THE SONIFICATION APPLICATION RESEARCH TOOLBOX

[4] The Sonification Handbook by Thomas Hermann, Andy Hunt, John G. Neuhoff (Eds.)

Edited by Thomas Hermann, Andy Hunt, John G. Neuhoff,  
<https://sonification.de/handbook/>

[5] Listen to our Data: Model-Based Sonification for Data Analysis T. Hermann and H. Ritter Department of Computer Science University of Bielefeld

[6] Principles of Sonification: An Introduction to Auditory Display and Sonification

[7] SONIPY: THE DESIGN OF AN EXTENDABLE SOFTWARE FRAMEWORK FOR SONIFICATION RESEARCH AND AUDITORY DISPLAY

[8] Evolving from xSonify: a new digital platform for sonorization

[9] Sonification of network traffic flow for monitoring and situational awareness  
Mohamed Debashi, Paul Vickers,

<https://journals.plos.org/plosone/article/file?id=10.1371/journal.pone.0195948&type=printable>

[10] <https://de.wikipedia.org/wiki/Objekterkennung>

[11] <https://www.youtube.com/watch?v=g7z4mkfRjl4>

[12] <https://www.youtube.com/watch?v=8QL69cAj2kU>

---

Bei Interesse melden Sie sich bitte unter:

Prof. Dr.-Ing.habil.  
Dirk Joachim Lehmann  
Data Science in IoT  
Fakultät für Informatik  
[di.lehmann@ostfalia.de](mailto:di.lehmann@ostfalia.de)