



Algorithmus Schmiede

Wir schreiben Programme, die komplexe Probleme lösen.



Data Science



Numerik



Physik

Vorstellung der Algorithmus Schmiede
bei Siemens Mobility

Die Algorithmus Schmiede

... schreibt Programme, die komplexe Probleme lösen.

Mehr Infos zu uns
hier klicken /
scannen



Unsere Entwickler sind promovierte Naturwissenschaftler.
Wir programmieren in **Python** und **C++**.

Sie profitieren von:

- Algorithmen mit höchster Zuverlässigkeit
- Tiefes Verständnis für physikalische Zusammenhänge
- Wissenschaftliche Arbeitsweise

Projektbeispiele Customer Services






- 📄 **Condition Based Maintenance:**
Wartung je nach Nutzungshistorie des Fahrzeugs
- 📄 **Optische Inspektion:** Erkennung von Schäden, Abnutzung oder Korrosion auf Videos / Fotos durch Algorithmen
- 📄 **Fehlersuche bei komplexen, nicht reproduzierbaren Problemen auf Basis von Fahrzeugdaten, z.B. Klimaanlage fällt unregelmäßig aus**
- 📄 **Virtuelle Sensoren:** Bewertung des Zustands der Weiche beim Überfahren
- 📄 **Voraberkennung von Störfällen**

Projektbeispiele Rolling Stock



Vorstufen zum autonomen Fahren:

-  **Fahrerassistenzsystem zur Kollisionsfrühwarnung**
-  **Verbesserte Erkennung der Schienenführung anhand von Sensordaten**
-  **Aufbau von Umgebungsmodellen durch Kombination von Sensoren**

Optimale Steuerung von Komponenten mithilfe von KI

-  **Traktion beim Beschleunigen**
-  **Bremsvorgänge (schnell, sanft, leise)**

Projektbeispiele Rolling Stock



- 📄 **Automatisiertes Softwaretesting**
für verschiedene Geräteversionen bei verschiedenen Kunden
- 📄 **Reduzierung von Fahrtlärm**
- 📄 **Optimale Anordnung von Fahrzeugkomponenten**
(Gewicht, Kabellänge, Kosten, ...)

KI Workshops



- 📄 **Marktüberblick zu KI: Use Cases, Standardlösungen und was technisch möglich ist**
- 📄 **KI Projekten identifizieren: Rentable Projekte im Unternehmen finden und in die Unternehmensstrategie einbetten**
- 📄 **KI nutzen mit Python: Mitarbeiter befähigen KI Anwendungen zu entwickeln**

Unsere Kunden

 Referenzprojekte



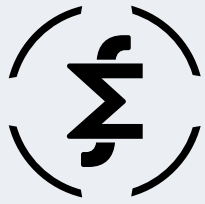
 **HEIDELBERG**

 **Dentsply
Sirona**



EUROPÄISCHE ZENTRALBANK

T Systems



Algorithmus Schmiede

Data Science | Numerik | Physik



Verpassen Sie nichts:



- Folgen Sie der [@Algorithmus Schmiede](#) auf LinkedIn
- Abonnieren Sie unseren [Newsletter](#)

Gerne berate ich Sie unverbindlich zu Ihrer Projektidee.



Dr. Markus Dutschke

Geschäftsführer, Algorithmus Entwickler



+49 178 148 3264



impact@algorithmus-schmiede.de



www.algorithmus-schmiede.de

Vielen Dank für Ihre Aufmerksamkeit

Es folgen Details zu den einzelnen Projektbeispielen

Condition Based Maintenance

- Aufbau von Digitalem Zwilling für Fahrzeug / Gerät:
Berechnung von Verschleiß in Abhängigkeit von Alter und Nutzung
- Synchronisation der Betriebsdaten mit Digitalem Zwilling des Fahrzeugs
- Verschleiß-Prognosen für alle Komponenten im Digitalen Zwilling
- Bestimmung des passenden Wartungstermins für einen Austausch
- Einbezug von Toleranzen (Fertigung, Messtoleranzen, künftige Fahrzeugnutzung) zum Ausschluss von Ausfällen nach gegebenem Konfidenzintervall



**Optimaler nächster Wartungszeitpunkt mit Liste von
Wartungsaktivitäten individuell für jedes Fahrzeug**

Optische Inspektion

- Sammlung und Klassifikation von Defekten mit Kamera-Bilddaten
- Aufbau von physikalischem Modell zur Bewertung verschiedener Defekte in realer Umgebung (mit Spiegelungen, Schmutz, ...)
z.B. Kein kontinuierlicher Farbverlauf -> Kratzer / Beule
- Schneller Algorithmus zur Live-Erkennung von Defekten
- Validierung der Ergebnisse und Bestimmung von erreichter Genauigkeit



**Schäden an Fahrzeugen oder Materialien
können im Vorbeifahren erkannt werden**

Komplexe Fehlersuche



Beispiel: Klimaanlage fällt alle 2-4 Monate aus

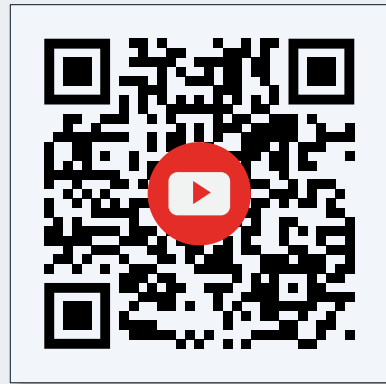
- Automatisiertes Zusammenführen von Datenquellen (PDF, Excel, Datenbank, SAP)
- Analyse von Fahrzeugdaten nach Unregelmäßigkeiten zum Problemzeitpunkt
- Analyse von Fahrzeugdaten nach zeitlicher Entwicklung
- Einbezug von weiteren Datenquellen (z.B. Wetter, Schichtpläne)
- Einbezug von Nutzungshistorie (z.B. Wetter letzte 7 Tage, gefahrene Kilometer diesen Monat)
- Vergleich zwischen verschiedenen Fahrzeugen

**Fehlerursache gefunden,
Softwareseitige Erkennung bevorstehender Ausfälle**

Virtuelle Sensoren

z.B. Bewertung des Zustands von Weichen anhand der Vibrationen beim Überfahren

- Einbezug physikalischer / logischer Rahmenbedingungen
z.B. Kurvenradien von Schienen, GPS Position springt nicht
- Physikalisches Modell zur Ableitung der zu ermittelnden Messgröße
- Benchmark des virtuellen Sensors



**Referenz-Projekt: Bestimmung der Farbe einer
Oberfläche aus Laserdistanzmessung**

Voraberkennung von Störfällen

- Automatisiertes Monitoring aller Fahrzeugdaten
- Zeitreihenanalyse zur Erkennung von Fehlentwicklungen
- Aufbau von Prognosemodellen
(z.B. Bremsleistung in Abhängigkeit zur Bremskraft)
und Warnung bei Abweichung zwischen Prognose und Messwerten
- Modell zur Bewertung der Dringlichkeit bei erkannter Abweichung



Reduzierung von Ausfallzeiten

Kollisionsfrühwarnung

- Identifikation aller bewegter Objekte
- Vorhersage der Bewegungsbahn
- Bei Personen: Erkennung der Wahrnehmung des Fahrzeugs
- Berechnung von Kollisionspunkt und Schwellenwerten für Eskalationsstufen (z.B. Signal, Warnung, Bremsung)



**Notwendige Voraussetzung für autonomes Fahren,
Vermeidung von Ausfallzeiten durch Verkehrsunfälle**

Erkennung der Schienenführung

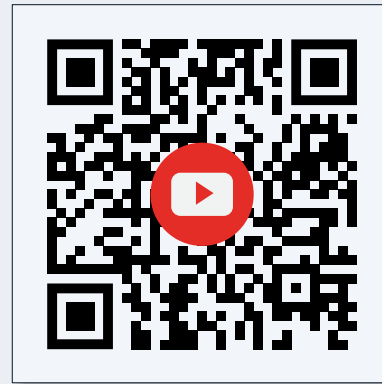


- Aufbau Basismodell via Computer Vision
- Verbesserung bekannter Herausforderungen (z.B. gerade Objekte in Verlängerung zur Schiene)
- Aufbau Bewertungssystem der Schienenerkennung beim Kunden: Erkannte Strecke vs. gefahrener Weg (Kann für jeden Kunden erneut verifiziert werden.)
- Verbesserte Erkennung der Schienenführung anhand von Sensordaten

**Notwendige Voraussetzung für autonomes Fahren,
Verbesserung der Kollisionsfrühwarnung**

Aufbau von Umgebungsmodellen

- Aufbau von Detektionsalgorithmen mit Konfidenzangabe für jeden Sensor
- Zusammenführung der erkannten Objekte/Events mit Konfidenzwerten und statistische Bewertung der „wahrscheinlichsten“ Realität
- Aufbau von Testing-Framework (Zufällige Verteilung von Objekten, Ableitung Sensorsignale, Rekonstruktion Ausgangslage, Abgleich)



**Notwendige Voraussetzung für autonomes Fahren,
Verbesserung der Kollisionsfrühwarnung**

Traktionsoptimierung durch KI

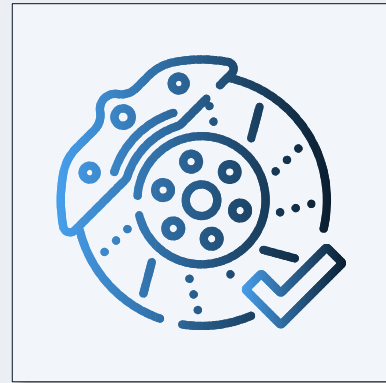
- Datensammlung an Testständen und im Betrieb
- Machine Learning Modell zur Berechnung der Traktion
- Einbezug verschiedener Steuerungsgrößen (z.B. Drehmoment, Sandung)
- Einbezug von externen Faktoren (Gleisbeschaffenheit, Witterung, Gewichtsverteilung, Steigung)



**Optimale Anfahrstrategie für
aktuelles Wetter, Steigung und Gewicht**

Optimale Bremswirkung durch KI

- Messung von Bremsprozessen an Versuchsstand
- Berechnung der Bremswirkung in Abhängigkeit von Steuergrößen (Bremsysteme, Sand) und Umweltfaktoren (Wetter, Schmutz, Steigung)
- Ausgleich fehlender Sensorik (z.B. Lauberkennung) durch Live-Analyse des bisherigen Bremsprozesses
- Bestimmung von Zielfunktionen für verschiedene Bremsmanöver (Notbremsung, Sanftes Bremsen, Leises Bremsen)



**Bremsen wird zuverlässiger und sanfter,
verbesserte Prognose von Bremswegen**

Automatisiertes Softwaretesting

Automatisiertes Softwaretesting
für verschiedene Fahrzeugversionen bei verschiedenen Kunden

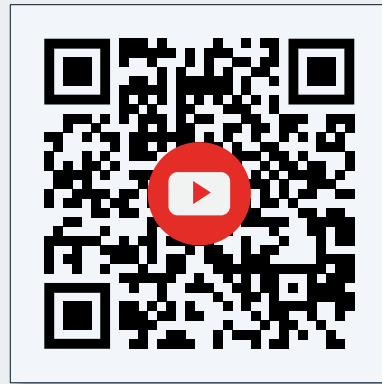
- Ein Digitaler Zwilling für jede Fahrzeugversionen
- Trainieren von Prognosen zum Fahrzeugverhalten während Betrieb
- Zurückspielen aller Digitalen Zwillinge in die CI/CD Pipelines
- Durchführung von automatisierten Softwaretests an allen Fahrzeugversionen gleichzeitig

**Reduzierung von Regressionsfehlern
und weniger Testaufwand**



Reduzierung von Fahrtlärm

- Definition von maximalen Lautstärkepegeln im Streckennetz
- Messung von Schallpegeln und Frequenzen durch das Fahrzeug
- Identifikation von Lärmursachen bei der Fahrt (z.B. Schwelle in Schiene, Quietschen bei langsamer Bremsung, ...)
- Entwicklung von Steuerungsalgorithmen zur Lärmreduzierung (Bremsgeschwindigkeit, Fahrtgeschwindigkeit)



**Modus für Flüster-Fahrt,
Reduzierung von Lärmschutzmaßnahmen**

Optimale Anordnung von Komponenten



- Mathematische Modellierung der einzelnen Komponenten über Eigenschaften (z.B. Gewicht, Kosten, Volumen, Position) und Abhängigkeiten (z.B. Kabelverbindungen, Rohrverbindungen)
- Aufbau von mathematischer Bewertungsfunktion in Hinblick auf Gewichtsverteilung, Gesamtgewicht, Kosten, Montageaufwand, ...
- Numerische Optimierung der Komponenten zur Erreichung des optimalen Werts der Bewertungsfunktion

**Optimierung nach Gewichtung in Ausschreibungen möglich
z.B. Optimale Anordnung für 70% Kosten, 30% Technik**

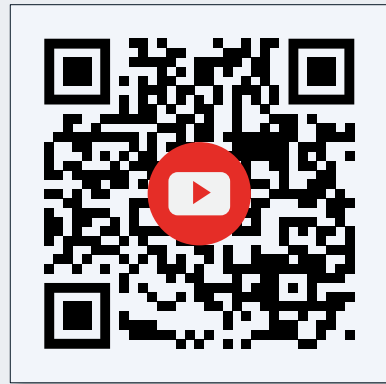
Workshop: Marktüberblick zu KI



Kursinhalte:

- Einführung in die KI und Use Cases
- Sprachverarbeitung:
Chatbots, LLMs, Transkription, Textähnlichkeitssuche (RAG), Übersetzung
- Bildverarbeitung:
Bilder generieren, Bilder bearbeiten, Bildrechte
- Datenanalyse und Vorhersage:
Prognose von zukünftigem Verhalten, Explainable AI, Gruppierung von Daten
- End-to-End Lösungen:
Werbevideos generieren, KI Avatare, Bild zu Video, Microsoft Copilot
- Praxisempfehlungen zu Auswahl und Einsatz von KI-Tools
- Ausblick über weitere Entwicklungen

Workshop: KI Projekten identifizieren



Ziel des Workshops ist, dass Ihre Unternehmenseinheit Maßnahmen und Projekte aus dem Bereich KI zusammenträgt, in die Unternehmensstrategie einordnet und die überzeugende Materialien für eine Budgetentscheidung erstellt.

Inhalte:

- Einführung in KI Anwendungsfälle
- Kreativ unterstützend moderierte Ideensammlung in der Gruppe
- Stakeholderanalyse
- Einordnung in Unternehmensstrategie
- Return on Investment Berechnung für KI-Innovationen
- Die optimale Präsentation beim Entscheider vorbereiten

Workshop: KI nutzen mit Python

Voraussetzungen:

- Grundkenntnisse in der Programmiersprache Python
- Möglichkeit die Kursinhalte im Nachgang zu vertiefen

Inhalte

- Struktur eines Data Science Projektes
- Grafische Darstellung von Daten
- Leistungsstarke Vorhersagemodelle
- Bewertung der Vorhersagequalität
- Textverstehen mit Natural Language Processing

