|  |  |  |
| --- | --- | --- |
| **MA 3062** | | |
| Untersuchung und prototypische Umsetzung eines Lifelong Deep Neural Network Algorithmus | | |
| **Simon Kamm** | | |
|  | | |
| **Beschreibung des Prototyps** | | |
|  | Prüfer: | Prof. Dr.-Ing. Michael Weyrich |
|  | Betreuer: | Benjamin Maschler, M.Sc. |
| Start: 29.04.2019 | | Abgabe: 29.10.2019 |
|  | |  |

**Dokument Versionsverwaltung**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Version | Autor | QS | Datum | Status | Änderungen |
| 0.1 | Kamm |  | 03.08.19 | in Bearb. | Erstellung |
|  |  |  |  |  |  |

*Allgemeine Hinweise zur Benutzung von IAS-VM Dokumentvorlagen*

*Diese Hinweise und Erläuterungen in den einzelnen Kapiteln der Vorlage sind kursiv dargestellt und müssen im Dokument gelöscht werden.*

*Verwendete Symbole:*

*\* generisch (kann mehrfach vorkommen)*

*[...] optional (kann auch weggelassen werden)*

*<...> Platzhalter (Bereich inklusive Klammern muss entsprechend ersetzt werden)*

***Bitte ändern Sie die Formatierung des Deckblatts nicht sondern tragen Sie nur die abgefragten Daten ein!***

*Bei der Erstellung eines Dokumentes hat das Dokument zunächst den Status ‘in Bearb.’ und die Versionsnummer 0.1. Dokumente im Status ‘in Bearb.’ können beliebig geändert, gespeichert und gedruckt werden. Wichtige Änderungen sollten allerdings im Feld „Änderungen“ dokumentiert werden. Die erste fertige Version bekommt die Versionsnummer 1.0. Ab der Version 1.0 wird das Speichern im Zustand ‘vorgelegt’ erlaubt. Dokumente mit niedrigerer Versionsnummer dürfen nicht mit ‘vorgelegt’ gespeichert werden.* ***Diese Informationen müssen manuell eingetragen werden.***

*Dokumente unterliegen ab dem Status ‘vorgelegt’ dem Konfigurationsmanagement und dürfen nicht mehr überschrieben werden. Nach einer Änderung muss das Dokument unter Angabe der durchgeführten Änderungen mit neuer Versionsnummer gespeichert werden. Die Versionsnummern muss dabei um 0.1 hochgezählt werden.*

*Bei der Erstellung eines Dokuments ist folgendes zu beachten:*

* *Wurden anhand des Prototyps die wesentlichen Merkmale der Konzeption umgesetzt?*
* *Kann anhand des Prototyps eine Evaluierung der Konzeption erfolgen?*
* *Ist der Aufwand für die Erstellung des Prototyps berechtigt?*
* *Verhält sich der Prototyp wie es in der Installations- und Benutzungsanleitung beschrieben ist?*
* *Sind Installation und Inbetriebnahme verständlich beschrieben?*
* *Sind alle wichtigen Funktionen ausreichend beschrieben?*

# Inhaltsverzeichnis

0 Inhaltsverzeichnis 2

1 Systemarchitektur 4

2 Beschreibung der Systemkomponenten 4

3 Installations- und Benutzungsanleitung 4

# Systemarchitektur

*Grafische Darstellung der Komponenten und Beziehungen untereinander (z. B. Daten*

*/Kontrollflüsse). Komponenten und Schnittstellen zwischen Komponenten sind zu benennen. (Visio-Diagramm)*

In diesem Abschnitt wird auf die Systemarchitektur des SW-Prototypen für die Realisierung des L DNN Algorithmus eingegangen. Dabei werden die einzelnen Komponenten sowie deren Schnittstellen und die Datenflüsse benannt. Für eine erste Übersicht ist in Abbildung 1 die Systemarchitektur grafisch dargestellt. Die blauen Boxen stellen dabei die einzelnen Komponenten dar, die im Folgenden kurz genannt und in Kapitel 2 detaillierter beschrieben werden. Blaue Pfeile stellen Funktionsaufrufe sowie Datenflüsse dar. Die orangenen Pfeile symbolisieren den logischen Programmablauf, und damit die Abhängigkeiten der einzelnen Komponenten.



Abbildung : Grafische Darstellung der Komponenten und deren Zusammenhang

Mithilfe der gewählten Komponentenstruktur soll es mit beschränktem Aufwand möglich sein, einzelne Bereiche der Systemarchitektur auszutauschen, ohne weitere Komponenten verändern zu müssen. So kann beispielsweise ein anderes Modul A gewählt werden, ohne die Komponenten davor oder danach anzupassen. Auch können in einem möglichen Experiment mit Live-Bildern die Input-Funktion und die Daten-Teilung und Vorbereitungskomponente ausgetauscht werden, wenn diese nicht benötigt werden.

Der generelle Ablauf wird in der Main-Funktion gesteuert. Aus dieser Funktion heraus werden alle Komponenten sowie deren Funktionen aufgerufen. Hier werden zudem auch Parameter zentral gesetzt, die im weiteren Verlauf genutzt werden, z.B. die Anzahl an Trainingsbildern pro Klasse.

In der **Input-Funktion** wird der ausgewählte Datensatz geladen, vorbereitet und separate Datensätze für die Trainings- und Testdaten zurückgegeben an die Main-Funktion. Zusätzlich kann ein zufälliges Trainingsbild vor und nach einer möglichen Bild-Augmentation visualisiert werden um den Einfluss zu sehen.

In der Komponente **Modul A** wird das vortrainierte CNN zur Feature-Extraktion geladen und eine Instanz dieser Klasse zurückgegeben. Als Eingabeparameter wird die Bildgröße angegeben, da davon abhängig die Gewichte des CNNs geladen werden.

In der Komponente **Modul B** wird eine (oder mehrere) Instanz(en) der Klasse Fuzzy-ARTMAP mit den übergegebenen Parametern initialisiert und diese an die Main-Funktion zurückgegeben. Parameter für die Initialisierung sind die Lernrate , der Vigilance-Parameter , der Choice-Parameter , der Parameter für das Match-Tracking sowie die Anzahl an Klassen und der zusätzliche Parameter für die Umsetzung des „Nothing I Know“-Konzepts.

Innerhalb der **Datenteilung- und Vorbereitung** wird der Datensatz für die weiteren Anwendungsfälle in die einzelnen Klassen aufgeteilt und die Daten für die Klassifizierungsaufgabe vorbereitet und die Features aus den Originaldaten extrahiert. Dafür benötigt diese Komponente als Eingangsdaten das Modul A, die Anzahl an Klassen im Datensatz, die Trainings- und Testdaten sowie die per Parameter einstellbare Anzahl an Trainings- und Testbildern pro Klasse. An die Main-Funktion werden dann die Features sowie die Labels der einzelnen Bilder in Listen pro Klasse zurückgegeben, aufgeteilt in eine Trainings- und Testliste.

Im **Kontinuierlichen/Verteilten Lernen** wird abhängig vom gewählten Anwendungsfall kontinuierliches Lernen auf einem Gerät oder auf verteilten Geräten (Instanzen) durchgeführt. Es werden die extrahierten und separierten Trainings- und Testdaten sowie deren Labels mit den initialisierten Instanzen des Moduls B übergeben. In diesem Modul findet das Training und Testen statt. Es werden die finalen Prädiktionen des Netzwerks auf den Testdaten mit den dazugehörigen Labels sowie der erzielten Genauigkeiten über die inkrementellen Aufgaben zurückgegeben.

In der **Berechnung und Visualisierung der Metriken** werden die Ergebnisse visualisiert und gegebenenfalls weitere Metriken berechnet. Implementiert ist die Darstellung einer *Confusion Matrix* pro Instanz von Modul B sowie das Darstellen der Klassifikationsgenauigkeit über die Anzahl an erlernten Klassen.

# Beschreibung der Systemkomponenten

*Grobe Gliederung des Systems in Systemkomponenten. Beschreibung der Aufgabe jeder Komponente.*

# Installations- und Benutzungsanleitung

*Beschreibung wie der Prototyp installiert und benutzt wird. Erläuterung komplexer Operationen und deren Durchführung. Typische bekannte Bedienungsfehler und Lösungsmöglichkeiten.*